

Ajax

Persiapan

- Download:
<https://drive.google.com/file/d/0BxKSp6Axeejq3lOodDVnd2FCX3c/view?usp=sharing>
- Latihan-latihan ini memerlukan koneksi aktif internet
 - Cek kembali koneksi internet anda
- Jalankan server apache dari XAMPP control panel
- Ekstrak file yang telah dibagikan,
- Salin folder week_8 ke htdocs

Ajax

- Asynchronous JavaScript and XML
- Fasilitas yang menghubungkan interaktifitas halaman (javascript) dengan fungsionalitas pada sisi server (data permanen, atau komputasi)
- Salah satu kegunaan Ajax adalah membuat halaman menjadi lebih ringan
 - Menggunakan ajax, membuat fungsi-fungsi aplikasi dapat digunakan tanpa berpindah halaman sama sekali
 - Membuat Single Page Application

Ajax untuk membuat halaman lebih ringan

- Se jauh ini karena menggunakan server localhost, maka semua resource yang kita muat terasa sangat cepat
- Padahal saat aplikasi diupload ke server hosting maka aplikasi bisa menjadi lambat karena banyak hal
 - Servernya hosting sibuk
 - Lewat router yang panjang antriannya
 - Lewat jaringan dengan bandwidth kecil
 - Dll
- Karenanya perlu untuk melakukan testing terhadap aplikasi dalam menghadapi kendala tersebut

Simulasi Lambat Memuat Resource

- Developer harus mengasumsikan bahwa pengguna aplikasinya memiliki kecepatan memuat halaman yang berbeda-beda,
 - (sangat cepat, cepat, lambat, sangat lambat).
- Sehingga kadang harus melakukan testing perilaku halaman apabila ada resource yang lambat dimuat.
- Salah satu website yang bisa digunakan adalah deelay.me

How to use it? Change your resource link to

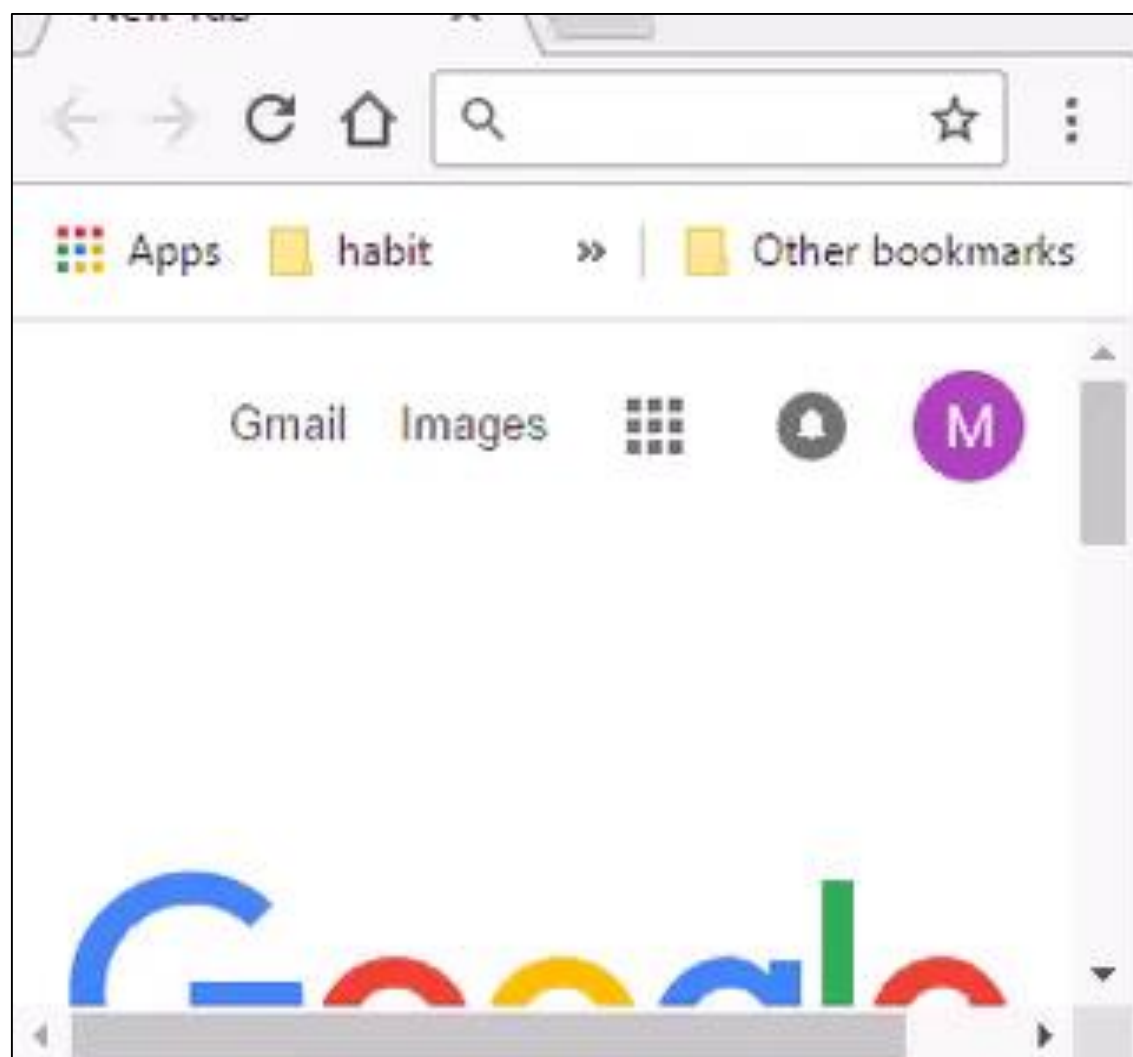
```

```

and watch it sloooooowing down



- Caranya adalah dengan menambahkan URL deelay.me di depan URL asli resourcenya



File Edit Format View Help

Gambar logo.png sebenarnya terletak di
htdocs\week_8\noajax\
sehingga URL untuk mengaksesnya adalah
`http://localhost/week_8/noajax/logo.png`

Karena server lokal, maka gambar
ditampilkan seketika, 0 delay

untuk melakukan simulasi menggunakan
deelay.me cukup tambahkan pada awal url
diatas

`deelay.me/waktudelay`

Misalnya untuk membuat gambar
ditunda/delay 4 detik (4000 milidetik)

Persiapan

- Membutuhkan koneksi aktif internet
- Ekstrak file yang dibagi ke htdocs
- Buka dua jendela, kiri dan kanan



[Lihat Data Mahasiswa](#)
[Lihat Data Mata Kuliah](#)

Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com

localhost/week_8/noajax/mahasiswa.php




[Lihat Data Mahasiswa](#)
[Lihat Data Mata Kuliah](#)

Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com

localhost/week_8/diajax/utama.php

localhost x

localhost/wee ☆




[Lihat Data Mahasiswa](#)
[Lihat Data Mata Kuliah](#)

Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com

localhost x

localhost/wee ☆



Lihat Data Mahasiswa
Lihat Data Mata Kuliah

Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com

Untitled - Notepad

File Edit Format View Help

Kiri Tanpa Ajax (pakai link)
Kanan dengan Ajax (pakai event klik)

Jendela kiri:

1. apabila menu diklik, halaman pindah, URL halaman berubah
2. sehingga semua resource (gambar, video, dll) harus dimuat lagi, walaupun mungkin resourcenya sama saja (mis. sama-sama gambar orang kuning)

Jendela kanan:

1. apabila menu diklik, halaman TIDAK pindah, URL halaman tidak berubah
2. Tidak perlu memuat ulang semua resource.

Ajax membuat halaman lebih interaktif.

Penjelasan Sederhana

No Ajax

- Saat menu diklik (link)
- Maka halaman akan melakukan request ke halaman tersebut
- Hasil request ditampilkan sebagai halaman



[Lihat Data Mahasiswa](#)
[Lihat Data Mata Kuliah](#)

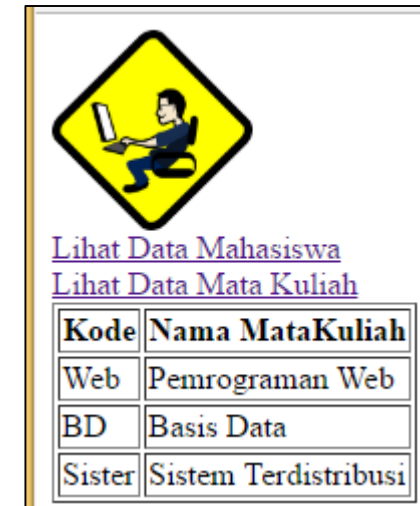
Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com



[Lihat Data Mahasiswa](#)
[Lihat Data Mata Kuliah](#)

Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com

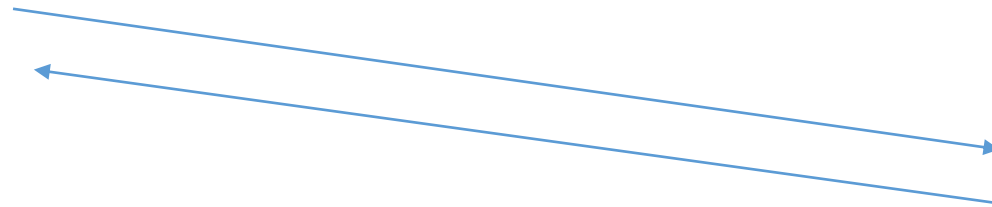
Localhost/week_8/noajax/mahasiswa.php



[Lihat Data Mahasiswa](#)
[Lihat Data Mata Kuliah](#)

Kode	Nama MataKuliah
Web	Pemrograman Web
BD	Basis Data
Sister	Sistem Terdistribusi

Localhost/week_8/noajax/matakuliah.php



- Saat menu diklik,
- maka dibelakang layar browser akan membuat request pada URL
 - misalnya URL localhost/week_8/diajax/matakuliah.php
- Hasil request diproses, misalnya ditampilkan pada bagian tertentu dari halaman

Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com

localhost/week_8/diajax/mahasiswa.php



Lihat Data Mahasiswa
Lihat Data Mata Kuliah

Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com

Kode	Nama MataKuliah
Web	Pemrograman Web
BD	Basis Data
Sister	Sistem Terdistribusi

localhost/week_8/diajax/matakuliah.php

With Ajax

No Ajax

```

<html>
<body>
  <img src=
    "http://deelay.me/3000/http://localhost/week_8/noajax/logo.png"
    width="100" height="100">

  <div>
    <div><a href="http://localhost/week_8/noajax/mahasiswa.php">
      Lihat Data Mahasiswa</a></div>
    <div><a href="http://localhost/week_8/noajax/matakuliah.php">
      Lihat Data Mata Kuliah </a></div>
  </div>

  <!-- tabel dibawah nantinya diganti dengan query data dari
  database (Materi sebelumnya) -->
  <table border=1>
    <tr><th>Nama</th><th>Email</th></tr>
    <tr><td>Adi</td><td>adhee@gmail.com</td></tr>
    <tr><td>Budi</td><td>Budiman@ayoo.com</td></tr>
  </table>

</body>

```

Noajax/Mahasiswa.php



[Lihat Data Mahasiswa](#)
[Lihat Data Mata Kuliah](#)

Nama	Email
Adi	adhee@gmail.com
Budi	Budiman@ayoo.com

Halaman Noajax/Mahasiswa.php

- Berisi gambar yang didelay 3 detik,
- Menu yang berupa link
- Data Dummy (mahasiswa)

```

<html>
<body>
  <img src=
  "http://deelay.me/3000/http://localhost/week_8/noajax/logo.png"
  width="100" height="100">

  <div>
  | <div><a href="http://localhost/week_8/noajax/mahasiswa.php">
  |   Lihat Data Mahasiswa</a></div>
  | <div><a href="http://localhost/week_8/noajax/matakuliah.php">
  |   Lihat Data Mata Kuliah </a></div>
  </div>

  <!-- tabel dibawah nantinya diganti dengan query data dari
  database (Materi sebelumnya) -->
  <table border=1>
  | <tr><th>Kode</th><th>Nama MataKuliah</th></tr>
  | <tr><td>Web</td><td>Pemrograman Web</td></tr>
  | <tr><td>BD</td><td>Basis Data</td></tr>
  | <tr><td>Sister</td><td>Sistem Terdistribusi</td></tr>
  </table>
</body>
</html>

```

Noajax/Matakuliah.php



[Lihat Data Mahasiswa](#)
[Lihat Data Mata Kuliah](#)

Kode	Nama MataKuliah
Web	Pemrograman Web
BD	Basis Data
Sister	Sistem Terdistribusi

Halaman Noajax/Matakuliah.php

- Berisi gambar yang didelay 3 detik,
- Menu yang berupa link
- Data Dummy (Matakuliah)

Aplikasi No Ajax

- Karena menggunakan link maka halaman akan pindah apabila link diklik
- Sehingga **semua** resource halaman harus dimuat

Muat resource

```
<html>
<body>
  <img src=
  "http://deelay.me/3000/http://localhost/week_8/noajax/logo.png"
  width="100" height="100">

  <div>
    <div><a href="http://localhost/week_8/noajax/mahasiswa.php">
    Lihat Data Mahasiswa</a></div>
    <div><a href="http://localhost/week_8/noajax/matakuliah.php">
    Lihat Data Mata Kuliah </a></div>
  </div>

  <!-- tabel dibawah nantinya diganti dengan query data dari
  database (Materi sebelumnya) -->
  <table border=1>
    <tr><th>Nama</th><th>Email</th></tr>
    <tr><td>Adi</td><td>adhee@gmail.com</td></tr>
    <tr><td>Budi</td><td>Budiman@ayoo.com</td></tr>
  </table>
</body>
</html>
```

Noajax/Mahasiswa.php

Muat resource

```
<html>
<body>
  <img src=
  "http://deelay.me/3000/http://localhost/week_8/noajax/logo.png"
  width="100" height="100">

  <div>
    <div><a href="http://localhost/week_8/noajax/mahasiswa.php">
    Lihat Data Mahasiswa</a></div>
    <div><a href="http://localhost/week_8/noajax/matakuliah.php">
    Lihat Data Mata Kuliah </a></div>
  </div>

  <!-- tabel dibawah nantinya diganti dengan query data dari
  database (Materi sebelumnya) -->
  <table border=1>
    <tr><th>Kode</th><th>Nama MataKuliah</th></tr>
    <tr><td>Web</td><td>Pemrograman Web</td></tr>
    <tr><td>BD</td><td>Basis Data</td></tr>
    <tr><td>Sister</td><td>Sistem Terdistribusi</td></tr>
  </table>
</body>
</html>
```

Noajax/Matakuliah.php

DiAjax

Halaman diajax/utama.php

- Berisi gambar yang didelay 3 detik,
- Body akan memanggil fungsi mintaData() saat selesai diLoad
- Menu akan memanggil fungsi mintaData() saat diklik
 - Semua pemanggilan mintaData() mengirimkan argument berupa suatu URL
 - fungsi mintaData() ini tentu saja didefinisikan di file javascript skripku.js
- Terdapat suatu elemen div dengan id="data", untuk apa?

```
<html>
  <head>
    <script language="javascript" src="js/skripku.js" ></script>
    <style> div>div:hover {background-color: cyan;} </style>
  </head>

  <body onload="mintaData('http://localhost/week_8/diajax/mahasiswa.php');">
    

    <div>
      <div onclick="mintaData('http://localhost/week_8/diajax/mahasiswa.php');">Lihat Data Mahasiswa
      </div>
      <div onclick="mintaData('http://localhost/week_8/diajax/matakuliah.php');">Lihat Data Mata Kuliah
      </div>
    </div>

    <div id="data" ></div>
  </body>
</html>
```

diajax/utama.php

- Fungsi `mintaData(urlnya)` sebenarnya hanya membungkus suatu permintaan data (`XMLHttpRequest`).

```
var permintaan = new XMLHttpRequest()
```

- Adalah membuat objek `XMLHttpRequest` kemudian menyimpan objek tersebut ke variable `permintaan`

```
function mintaData(urlnya) {  
    var permintaan = new XMLHttpRequest();  
    permintaan.onreadystatechange = function() {  
        if (this.readyState == 4 && this.status == 200) {  
            document.getElementById("data").innerHTML = this.responseText;  
        }  
    };  
    permintaan.open("GET", urlnya, true);  
    permintaan.send();  
}
```

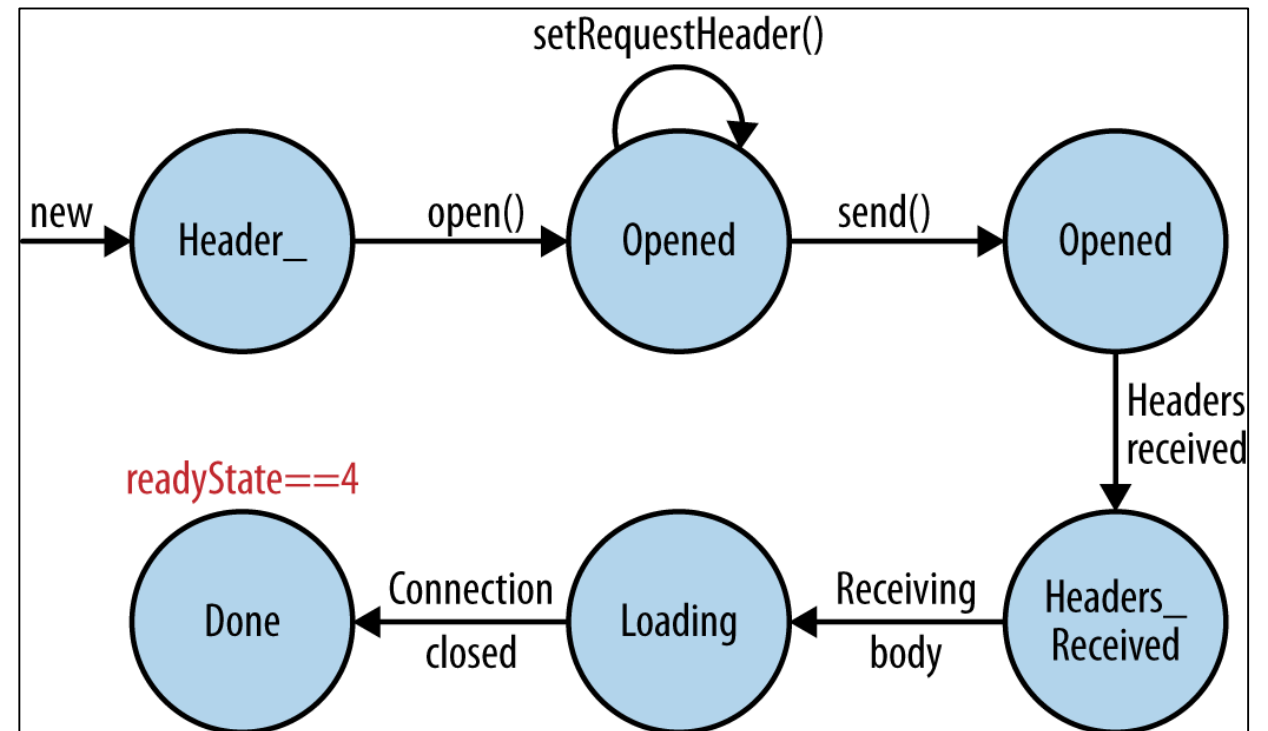
diajax/js/skripku.js

Ready State

- Suatu request yang dikirimkan akan berpindah dari satu state ke state berikutnya

- Ready State

0. Request belum dibuat
1. Koneksi dengan server selesai tercipta
2. Semua HTTP header telah diterima
3. Semua body telah diterima
4. Transfer data telah selesai



- Perpindahan dari satu state ke state lainnya adalah event (sama seperti event klik, mouseover, dll)
 - Sehingga dapat diikat (bind) dengan pemanggilan suatu fungsi

`permintaan.onreadystatechange = function() {`

- Adalah mengikat event perubahan state, pada suatu fungsi anonym (tanpa-nama),
 - Sehingga apabila perubahan state terjadi, maka fungsi ini akan dipanggil

```
function mintaData(urlnya) {
    var permintaan = new XMLHttpRequest();
    permintaan.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            document.getElementById("data").innerHTML = this.responseText;
        }
    };
    permintaan.open("GET", urlnya, true);
    permintaan.send();
}
```

diajax/js/skripku.js

Status Request

- Selain state, suatu request juga memiliki status, status memiliki nomor kode
- Misalnya yang sering dijumpai
 - **500 Internal Server Error**
 - **403 Forbidden**
 - **404 Not Found**
 - **502 Bad Gateway**
- Sebenarnya ada lagi status lain, walaupun sering tidak ditampilkan, yaitu apabila halaman sukses diakses
 - **200 OK**

If (this.readyState ==4 && this.status == 200) {

- Artinya perintah didalam if tersebut hanya akan dijalankan apabila,
 - statenya 4 (transfer data selesai),
 - dan statusnya 200 (request sukses dijalankan)
- **this** berarti mengacu pada objek yang tersimpan pada variable permintaan

```
function mintaData(urlnya) {  
    var permintaan = new XMLHttpRequest();  
    permintaan.onreadystatechange = function() {  
        if (this.readyState == 4 && this.status == 200) {  
            document.getElementById("data").innerHTML = this.responseText;  
        }  
    };  
    permintaan.open("GET", urlnya, true);  
    permintaan.send();  
}
```

diajax/js/skripku.js

- Response dari suatu request akan diterima, diterima melalui atribut **responseText**
- Sehingga sebenarnya untuk menampilkan dapat saja pakai `console.log(this.responseText);` atau `alert(this.responseText);`
- Tetapi yang dilakukan disini adalah dimasukkan pada bagian dalam (innerHTML) elemen yang id="data"

```
function mintaData(urlnya) {  
    var permintaan = new XMLHttpRequest();  
    permintaan.onreadystatechange = function() {  
        if (this.readyState == 4 && this.status == 200) {  
            document.getElementById("data").innerHTML = this.responseText;  
        }  
    };  
    permintaan.open("GET", urlnya, true);  
    permintaan.send();  
}
```

diajax/js/skripku.js

```
<div onclick="mintaData('http  
</div>  
</div>  
<div id="data" ></div>  
</body>  
</html>
```

Cuplikan utama.php


```
permintaan.open("GET", urlnya, true);
```

Adalah menentukan:

- jenis request, "GET" atau "POST",
- URL request
- dan jenis request true (asinkron), false (sinkron)

```
function mintaData(urlnya) {  
    var permintaan = new XMLHttpRequest();  
    permintaan.onreadystatechange = function() {  
        if (this.readyState == 4 && this.status == 200) {  
            document.getElementById("data").innerHTML = this.responseText;  
        }  
    };  
    permintaan.open("GET", urlnya, true);  
    permintaan.send();  
}
```

diajax/js/skripku.js

`permintaan.send();`

Mengirimkan request

```
function mintaData(urlnya) {  
    var permintaan = new XMLHttpRequest();  
    permintaan.onreadystatechange = function() {  
        if (this.readyState == 4 && this.status == 200) {  
            document.getElementById("data").innerHTML = this.responseText;  
        }  
    };  
    permintaan.open("GET", urlnya, true);  
    permintaan.send();  
}
```

`diajax/js/skripku.js`

Halaman diajax/utama.php

- Jadi saat menu diklik, browser dibelakang layar akan melakukan request ke URL, sesuai URL yang dikirim ke fungsi mintaData()
- Dibuat request ke URL tersebut
- Response (balasan) akan ditampilkan pada elemen dengan id="data"

```
<html>
  <head>
    <script language="javascript" src="js/skripku.js" ></script>
    <style> div>div:hover {background-color: cyan;} </style>
  </head>

  <body onload="mintaData('http://localhost/week_8/diajax/mahasiswa.php');">
    

    <div>
      <div onclick="mintaData('http://localhost/week_8/diajax/mahasiswa.php');">Lihat Data Mahasiswa
      </div>
      <div onclick="mintaData('http://localhost/week_8/diajax/matakuliah.php');">Lihat Data Mata Kuliah
      </div>
    </div>

    <div id="data" ></div>
  </body>
</html>
```

1. Menu diklik
2. Request dibuat ke URL
3. Hasilnya ditampilkan

diajax/utama.php

Ajax with JQuery

- Dengan jquery, beberapa baris perintah tersebut dapat lebih disingkat lagi
 - Satu baris saja!
- Jangan lupa disambungkan dengan halamannya,
 - Harus diatas skripku.js

```
/*  
function mintaData(urlnya) {  
    var permintaan = new XMLHttpRequest();  
    permintaan.onreadystatechange = function() {  
        if (this.readyState == 4 && this.status == 200) {  
            document.getElementById("data").innerHTML = this.responseText;  
        }  
    };  
    permintaan.open("GET", urlnya, true);  
    permintaan.send();  
}  
*/  
function mintaData(urlnya) {  
    $("#data").load(urlnya);  
}
```

Ubah pada diajax/js/skripku.js

```
<head>  
    <script language="javascript" src="js/jquery-3.1.1.min.js" ></script>  
    <script language="javascript" src="js/skripku.js" ></script>  
    <style> div>div:hover {background-color: cyan;} </style>  
</head>
```

Tambah pada diajax/utama.php

JQuery bukan hanya untuk mempersingkat perintah

- Sebagian besar browser terbaru (Chrome, IE7+, Firefox, Safari, and Opera) memang sudah menggunakan perintah yang sama untuk membuat objek xmlhttprequest

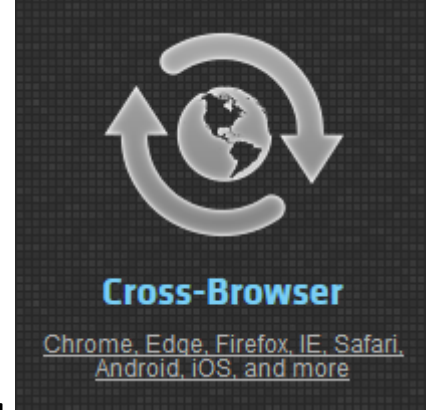
```
var permintaan = new XMLHttpRequest();
```

- Tetapi browser lain (Internet Explorer (IE5 and IE6)), menggunakan sintaks yang berbeda

```
var permintaan = new ActiveXObject("Microsoft.XMLHTTP");
```

- Sehingga program kita tidak akan jalan.

JQuery bukan hanya untuk menyingkat baris perintah

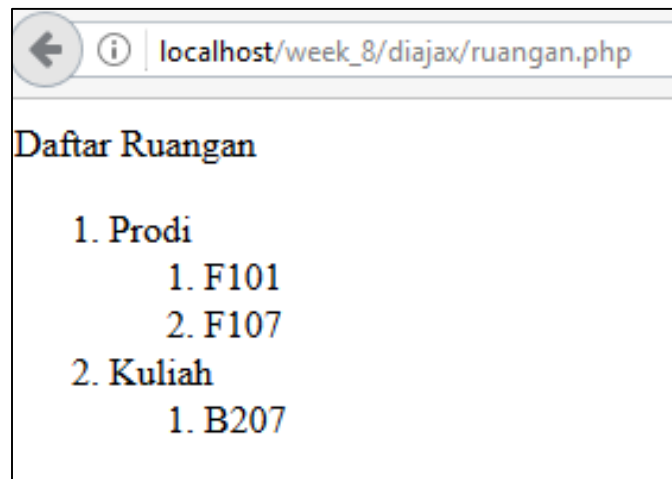


- Library JQuery sudah dilengkapi dengan pengecekan browser yang digunakan,
- Sehingga secara otomatis akan membuat objek Ajax sesuai dengan browser yang dipakai user,
 - Membuat aplikasi ajax kita bisa bisa dijalankan pada bermacam jenis browser yang dipakai client
- Hal inilah yang dimaksud dengan keunggulan Cross-Browser JQuery
 - Membuat developer tidak perlu lagi berpikir tentang apakah codenya dapat dijalankan pada browser client yang bermacam-macam.

Latihan

- Buatlah file baru ruangan.php
- Simpan pada folder diajax
- Cobalah tampilkan pada browser.

http://localhost/week_8/diajax/ruangan.php



Daftar Ruangan

```
<ol>
<li>Prodi
  <ol>
    <li>F101</li>
    <li>F107</li>
  </ol>
</li>
<li>Kuliah
  <ol>
    <li>B207</li>
  </ol>
</li>
</ol>
```

/diajax/ruangan.php

- Pada diajax/utama.php, salin salah satu item menu, paste tepat dibawahnya,
 - Tulisannya jadi “Lihat Data Ruangan”
 - URLnya, pakai URL ruangan.php yang tadi




Muncul menu baru

```
<div>  
  <div onclick="mintaData('http://localhost/week_8/diajax/mahasiswa.php');">Lihat Data Mahasiswa</div>  
  <div onclick="mintaData('http://localhost/week_8/diajax/matakuliah.php');">Lihat Data Mata Kuliah</div>  
  <div onclick="mintaData('http://localhost/week_8/diajax/ruangan.php');">Lihat Data Ruangan</div>  
</div>
```

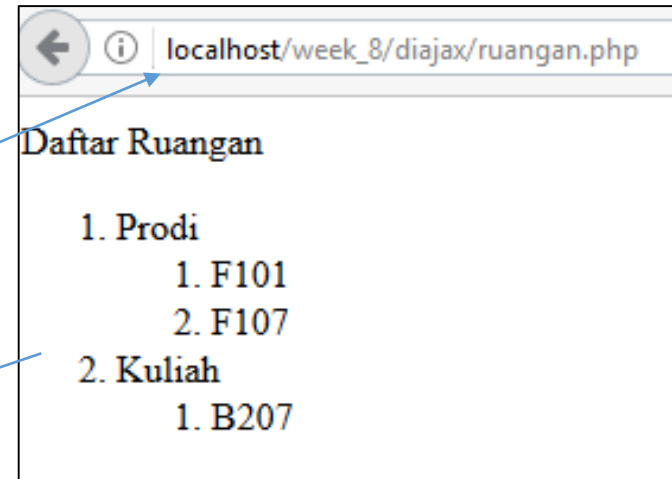
Edit utama.php

Ubah URL, dan tulisannya



Lihat Data Mahasiswa
Lihat Data Mata Kuliah
Lihat Data Ruangan
Daftar Ruangan

1. Prodi
 1. F101
 2. F107
2. Kuliah
 1. B207



localhost/week_8/diajax/ruangan.php

Daftar Ruangan

1. Prodi
 1. F101
 2. F107
2. Kuliah
 1. B207

1. Kalau menu lihat data ruangan diklik
2. Dibelakang layar, browser akan mengirim request ke `http://localhost/week_8/diajax/ruangan.php`
3. Response dari URL tersebut kemudian ditampilkan

Kesimpulan

- Developer harus turut memperhatikan ukuran resource yang digunakan pada halaman
 - Kalau memungkinkan perkecil ukuran resource yang digunakan
 - Atau membuat resource cukup dimuat sekali
- Ajax adalah menggunakan Javascript untuk membuat request HTTP
 - Ajax bekerja dengan membuat request (dibelakang layar) pada suatu URL
 - Sehingga tidak perlu pindah halaman
- Agar aplikasi web dapat dijalankan di berbagai jenis browser
 - Gunakanlah library javascript yang mendukung kompatibilitas cross-browser

<http://imgtfy.com/?q=video+compressor>

<http://imgtfy.com/?q=image+compressor>

Referensi

- [1]. <https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>
- [2]. www.jquery.com
- [3]. http://enterprisewebbook.com/ch2_ajax_json.html