

MODUL WEB PROGRAMMING 1



PENYUSUN

Andrian Eko Widodo, M.Kom

*PROGRAM STUDI SISTEM
INFORMASI STMIK NUSA MANDIRI
JAKARTA*

DAFTAR ISI

JUDUL	i
PENYUSUN	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
MODUL 1 HTML.....	1
MODUL 2 HTML Form.....	4
MODUL 3 CASCADING STYLE SHEETS (CSS).....	17
MODUL 4 Java Scripts I.....	29
MODUL 5 Java Scripts 2 (jQuery).....	36
MODUL 6 Java Scripts (AJAX).....	44
MODUL 7 Pengenalan PHP	53
MODUL 8 MATERI PHP LANJUT PART 2.....	58
MODUL 9 KONEKSI PHP DENGAN MySQL.....	62
MODUL 10 Bahasa PHP dan MySQL	66

MODUL 1

HTML

Tujuan

1. Mahasiswa dapat membuat desain Web menggunakan HTML.
2. Mahasiswa dapat mengerti dasar dari pemrograman Web.

HTML (Hypertext Markup Language) adalah bahasa format dari tag XML(Extended Markup Language) yang digunakan sebagai standar tampilan dari halaman Web. HTML dapat diterima sebagai tampilan halaman Web setelah melalui proses interpretasi dari Web browser. Selain itu HTML memiliki struktur yang fleksibel, tanpa logik serta toleran terhadap kesalahan.

Standar Minimum elemen HTML:

- Document Type Declaration (DTD)
- Head
- Body

```
<DTD>
<html>
<head>
<!--Terdiri dari elemen-elemen yang mendeklarasikan konten sebagai
pendukung
ingkungan seperti title, meta dokumen, CSS, Java Script -->
</head>
<body>
<!--Terdiri dari elemen-elemen yang diinterpretasikan sebagai
tampilan web -->
</body>
</html>
```

1.1. DTD

Sebagai standar versi dokumen W3C yang digunakan sebagai representasi setiap elemen pada dokumen Web,

Contoh:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
```

Standar tipe dokumen tersebut dapat digunakan untuk validasi elemen halaman Web.

1.2. Meta Dokumen

Elemen meta sebagai identitas dari halaman Web yang biasa terdiri dari owner, keywords, la y o u t, ataupun inisialisasi proses seperti r e f r e s h.

Contoh:

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"
/>
<meta name="owners" content="myWeb" />
```

```

· <meta name="keywords" content="Sistem Informasi, Jurnal, Publikasi Ilmiah" />
· <meta name="layout" content="main" />
· <meta HTTP-EQUIV="REFRESH" content="0; url=/myweb/myurl">

```

1.3. Elemen HTML

Jenis-jenis elemen data yang akan ditampilkan pada halaman Web seperti:

Form, Text, Image, Table, Drop-Box, Button dan lainnya.

Contoh:

· Title,

Ditampilkan pada header browser

```
<title>Praktikum Pemrograman Web</title>
```

Text

```

<h1>Selamat Datang Di Website Kami!</h1>
<h2>Web Ini Adalah Tutorial Pemrograman Web</h2>
<h3>Praktikum Pemrograman Web</h3>
<p> <a href="http://uin-malang.ac.id"> Teknik Informatika
UIN Malang </a> </span> link for your success way </p>

```

Image

```



```

Form

```

<form method="POST" action="response2.jsp"
enctype="multipart/form-data">

```

Drop-Box

```

<select name="My Lab">
<option value='CC'>Lab. Praktikum Computer Vision</option>
<option value='SI'> Lab. Sistem Informasi </option>
<option value='CAI'> Lab. Computing & AI </option>
</select>

```

File Browser

```
<input type="file" name="upload" value="" width="200" />
```

Table

```

<table border="1" width="10">
<thead>
<tr>
<th>NIM</th>
<th>Mahasiswa</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>Ani Rahmawati </td>
</tr>
<tr>
<td>124</td>
<td>Budi Susilo</td>
</tr>
</tbody>
</table>

```

List

Bidang Minat:

```
<ol>
<li>Sistem Informasi
<ul>
<li>Data Warehousing </li>
<li>Enterprise Architecture </li>
</ul>
</li>
<li>Multimedia
<ul>
<li>Komputer Grafik </li>
<li>Machinema </li>
</ul>
</li>
</ol>
```

1.4. Instruksi Praktek

- Buat file form1.html
- Kopikan setiap kode script html sebagaimana diatas
- Amati yang terjadi pada browser dengan melakukan refresh pada setiap perubahan kode
- Aktifkan plugin firebug pada browser Mozilla atau chrome, kemudian lakukan inspeksi sambil melakukan perubahan-perubahan kode HTML

1.5. Jawablah pertanyaan berikut:

1. Lakukanlah perubahan pada kode HTML sesuai keinginan anda, Pernahkah terjadi error ketika adakesalahan kode?
2. Cobalah ubah nilai meta refresh `<meta HTTP-EQUIV="REFRESH" content="5; url=http://google.co.id">` apa yang terjadi?, maka jelaskan apa diantara kegunaan meta dokumen?
3. Jelaskan fungsi tag HTML pada Table: thead, tbody, tr, th serta td!

1.6. Tugas I

Dengan menggunakan HTML buatlah desain form HTML, sebagai berikut:

Form HTML

Name:

Gender: ☐ Man ☐ Woman

Address:

Country:

Email:

Photo:

Comment:

Form Registrasi

MODUL 2

HTML Form

Tujuan

3. Mahasiswa mampu membuat form pada dokumen HTML
4. Mahasiswa memahami cara penggunaan dari fasilitas Form pada halaman dokumen HTML
5. Mahasiswa mampu menginputkan field sesuai dengan kegunaan

Overview

Salah satu fasilitas yang disediakan oleh dokumen HTML memungkinkan kita untuk melakukan "interaksi lebih" terhadap halaman dokumen tersebut. Semisal, kemampuan dokumen HTML mampu menyediakan fasilitas yang dapat menerima masukan atau isian data dari user. Data isian user ini nantinya akan dapat diproses lebih lanjut menjadi informasi yang dibutuhkan oleh user maupun oleh pemilik situs tersebut. Teknik pengisian ini dilakukan menggunakan elemen Form.

Form dalam HTML adalah suatu bagian yang berfungsi sebagai input atau masukan dari pengguna yang kemudian akan diproses atau diolah untuk dapat digunakan sesuai dengan kebutuhan. Contohnya proses pengiriman data, browse, hapus, penyuntingan data dan lain sebagainya.

2.1 Menu Form

Form dalam HTML adalah suatu bagian yang berfungsi sebagai input atau masukan dari pengguna yang kemudian akan diproses atau diolah untuk dapat digunakan sesuai dengan kebutuhan. Contohnya proses pengiriman data, browse, hapus, penyuntingan data dan lain sebagainya. Cara penulisan form dalam html adalah.

```
<form action="_url_" method="get|post" enctype="">
    ... elemen - elemen yang ditambahkan dalam form ...
</form>
```

```
<form name="nama_form" method="post/get" action="url">...isi form...</form>
```


Pada penulisan form "nama form" adalah nama form yang sedang digunakan, method adalah metode atau cara yang digunakan untuk menyimpan data ke server sedangkan action adalah alamat atau bisa juga file lain yang digunakan untuk memproses input dari form kedalam server.

2.2 Jenis-jenis media input dalam Form

Pada bagian form tersedia kontrol-kontrol input yang digunakan untuk masing-masing keperluan dalam memasukan data input. Dan kontrol form terdiri dari :

2.2.1 Kontrol Berupa Text

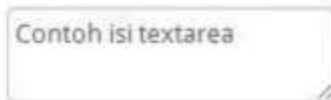
Pengisian informasi dalam bentuk teks pada form HTML dapat dilakukan dengan menggunakan dua buah elemen: `textarea` dan `input`. `textarea` digunakan untuk masukan teks yang terdiri dari beberapa baris, sementara `input` digunakan untuk masukan teks yang hanya satu baris.

Penggunaan elemen `textarea` dapat dilakukan dengan sangat sederhana, hanya langsung memasukkan tag-nya saja:

```
<textarea>
</textarea>
```

selain memberikan tag kosong seperti di atas, kita juga dapat mengisi tag untuk mendapatkan nilai masukan standar:

```
<textarea>
    Contoh isi textarea
</textarea>
```



Gambar 2.1 Contoh Penggunaan Elemen TextArea

Pengaturan panjang dan lebar dari `textarea` dapat dilakukan melalui CSS, dengan menggunakan properti `height` dan `width`.

Walaupun dapat mengisi teks dengan banyak sekaligus, `textarea` tentunya tidak dapat digunakan untuk seluruh kasus pengisian data. Seringkali kita

menginginkan pengguna hanya mengisi data singkat, tanpa isi teks yang banyak. Untuk jenis masukan seperti itu, kita dapat menggunakan elemen input:

```
<input type="text">
```

yang akan menghasilkan elemen masukan seperti berikut:



Gambar 2.2 Contoh Penggunaan Elemen Input

Perhatikan juga bahwa kita menggunakan atribut type pada elemen ini untuk menentukan data-data yang akan diisi. Terdapat banyak jenis tipe data yang dapat kita isikan, dan browser akan menyesuaikan jenis masukan teks yang ada, sesuai dengan atribut type yang kita tentukan. Nilai-nilai yang dapat diisi pada atribut type yaitu:

- | | |
|-------------------|--------------|
| 1. color | 9. time |
| 2. datetime-local | 10. datetime |
| 3. number | 11. month |
| 4. tel | 12. search |
| 5. week | 13. url |
| 6. date | 14. password |
| 7. email | 15. text |
| 8. range | 16. file |

2.2.2 Kontrol Berupa Radio Button

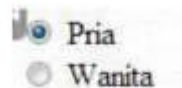
Radio button di gunakan untuk menyatakan pilihan yang bersifat tunggal, umumnya pada web di gunakan untuk kelengkapan pertanyaan seperti pada pendaftaran online, biasanya pertanyaan itu adalah pilihan untuk menyetakan jenis kelamin , agama, ataupun sebagainya. Radio Button ini akan sering kita temui saat kita berselancar di dunia maya yang terkait dengan proses input data seperti pendaftaran dll.

Radio button dibuat dalam HTML dengan menggunakan elemen input, dengan atribut type bernilai : code`radio` :

```
<input type="radio" name="sex"
value="pria">Pria<br><input type="radio" name="sex"
value="wanita">Wanita
```


dalam pembuatan elemen radio button, kita juga wajib menambahkan dua atribut lainnya, yaitu name dan value. Atribut name digunakan untuk memberitahukan browser bahwa radio button dengan atribut name yang sama adalah merupakan kumpulan radio button yang sama, sehingga pengguna tidak boleh memilih dua buah pilihan pada radio button tersebut.

Atribut value digunakan untuk menyimpan nilai yang ingin kita kirimkan ke server.



Gambar 2.3 Contoh Penggunaan Elemen Radio Button

2.2.3 Kontrol Berupa Checkbox

Berbeda dengan radio button yang hanya memungkinkan user memilih satu pilihan, pada input type checked box Anda dapat memilih satu atau beberapa pilihan, atau tidak memilih sama sekali. Pilihan ini biasanya dipakai untuk memasukkan data yang sifatnya opsional. Anda dapat menggunakan "Checkbox" pada atribut. Input checkbox adalah sebuah masukan untuk memilih sesuatu opsi/ pilihan. Jika dipilih maka pengguna akan men-*check* (mencentang) dan jika tidak memilih akan dibiarkan saja.

Elemen ini sama seperti radio button, dibuat dengan elemen input yang nilai atribut type-nya berisikancheckbox.

```
<input type="checkbox" name="day"
value="senin">Senin<br><input type="checkbox" name="day"
value="selasa">Selasa<br><input type="checkbox" name="day"
value="rabu">Rabu
```

Seperti yang dapat dilihat, atribut name dan value juga wajib dimiliki oleh checkbox, dengan alasan yang sama untuk radio button.



Gambar 2.4 Contoh Penggunaan Elemen Checkbox

Meskipun dapat memberikan pilihan nilai kepada pengguna, radio button maupun checkbox sangat tidak ideal untuk digunakan pada pilihan yang memiliki nilai yang sangat banyak. Misalnya, pilihan negara yang ditinggali oleh pengguna dari ratusan

negara yang ada di dunia akan jika ditampilkan pada radio button akan menyebabkan daftar yang dipaparkan sangat banyak dan sulit dibaca oleh pengguna. Dalam kasus seperti ini lebih baik kita menggunakan **dropdown list**.

2.2.4 Dropdown list

Pembuatan elemen dropdown dilakukan dengan menggabungkan dua elemen, yaitu select dan option. Elemeselect membungkus seluruh elemen option yang ada, untuk membentuk sebuah dropdown. Elemen option sendiri merupakan nilai dari dropdown yang diinginkan. Perhatikan kode di bawah:

```
<select name="country">
  <option value="indonesia">Indonesia</option>
  <option value="malaysia">Malaysia</option>
  <option value="filipina">Filipina</option>
  <option value="vietnam">Vietnam</option>
</select>
```

yang akan menghasilkan:



Gambar 2.5 Contoh Penggunaan Elemen Dropdown

dan jika ingin memungkinkan pengguna memilih beberapa pilihan kita dapat menambahkan atribut multiple pada elemen select:

```
<select name="country" multiple>
  <option value="indonesia">Indonesia</option>
  <option value="malaysia">Malaysia</option>
  <option value="filipina">Filipina</option>
  <option value="vietnam">Vietnam</option>
</select>
```



Gambar 2.6 Contoh Penggunaan Elemen Dropdown dengan Pilihan Banyak

2.2.5 Button

Pada saat kita melakukan browsing di dunia maya, dan mengakses sebuah website. Sering kita menemui tombol-tombol fungsi seperti reset, submit, send dan lain sebagainya. Dan sering pula kita sering menemui gambar yang dibuat fungsi seperti tombol.

2.2.5.1 Submit

Tombol ini berfungsi untuk memanggil url yang sudah didefinisikan di atribut *action*. Pembuatan tombol submit juga dilakukan dengan menggunakan elemen input, yang atribut type-nya diisi dengan nilai submit, seperti berikut:

```
<input type="submit" name="submit" value="Masukkan Form">
```

Perhatikan bahwa berbeda dengan elemen-elemen input sebelumnya, nilai dari atribut value diproses menjadi teks dari tombol yang ditampilkan ke pengguna. Hal ini menyebabkan kita tidak lagi perlu menambahkan teks setelah elemen form, seperti pada radio button ataupun checkbox.



Gambar 2.7 Contoh Penggunaan Tombol Submit

2.2.5.2 Reset

Tombol ini berfungsi untuk mengembalikan form ke kondisi awal (mengosongkan nilai semua elemen yang ada pada form). Pembuatan tombol dilakukan sama persis seperti pada tombol submit, dengan perbedaan nilai pada atribut type, yang diisi dengan reset pada tombol penghapusan ini.

```
<input type="reset" name="reset" value="Kosongkan Form">
```


2.2.6 Organisasi Elemen Form

Mengetahui bagaimana membuat elemen-elemen masukan / input pada form masih merupakan langkah awal dalam pembuatan form HTML. Elemen-elemen masukan form jika ditampilkan tanpa informasi tambahan tidaklah berguna, karena pengguna tidak dapat mengetahui data apa yang harus diisikan ke dalam elemen-elemen tersebut.

Untuk memberikan informasi tambahan kepada pengguna, HTML juga menyediakan elemen-elemen yang dapat digunakan sebagai penanda dari elemen masukan form. Adapun elemen-elemen yang dapat digunakan untuk tujuan tersebut yaitu label, fieldset, dan legend.

2.2.6.1 Label

Elemen label memberikan fasilitas untuk menambahkan teks pada elemen masukan form. Elemen ini diisikan dengan deskripsi dari elemen masukan yang ingin ditambahkan, dan harus memiliki atribut `for`. Atribut `for` pada elemen label berisikan nilai yang sama dengan atribut `id` pada elemen masukan form. Pengisian nilai yang sama akan mengikatkan elemen label dengan elemen masukan, sehingga pengguna dapat langsung mengisikan nilai dengan melakukan klik pada teks yang dihasilkan oleh elemen label.

Perhatikan kode berikut di mana isi dari atribut `for` pada label adalah sama dengan isi atribut `id` pada elemen input.

```
<label for="username">Username</label>
<input type="text" name="username" id="username">
```

2.2.6.2 Fieldset

Elemen fieldset merupakan elemen yang digunakan untuk membungkus beberapa elemen masukan form, untuk menandakan bahwa elemen-elemen tersebut merupakan elemen masukan yang berada pada satu grup yang sama, atau saling berhubungan.

Secara standar, elemen fieldset akan memberikan border di sekitar grup elemen-elemen di dalamnya, yang tentunya dapat diubah dengan menggunakan CSS. Berikut adalah contoh penggunaan fieldset :

```

<fieldset>
  <label for="username">Username</label>
  <input type="text" name="username" id="username">
  <label for="password">Password</label>
  <input type="text" name="password" id="password">
</fieldset>

```

yang hasil eksekusinya adalah:



gambar 2.8 Contoh Penggunaan Fieldset

2.2.6.2 Legend

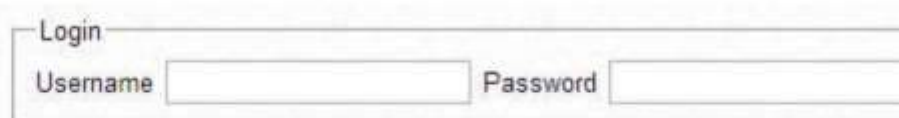
Elemen legend digunakan untuk memberikan judul pada sebuah fieldset. Penggunaan elemen legend sangat sederhana, hanya dengan menambahkan elemen tersebut sebagai *child* pertama dari fieldset, seperti berikut:

```

<fieldset>
  <legend>Login</legend>
  <label for="username">Username</label>
  <input type="text" name="username" id="username">
  <label for="password">Password</label>
  <input type="text" name="password" id="password">
</fieldset>

```

yang akan menghasilkan tampilan seperti berikut:



Gambar 2.9 Contoh Penggunaan Legend

dan tentunya tampilan dari legend dapat diubah dengan menggunakan CSS.

2.2.7 Validasi Masukan pada Form

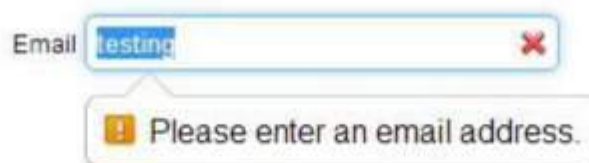
Seringkali dalam pengembangan form pada aplikasi, terdapat elemen-elemen yang tidak wajib diisi oleh pengguna dan elemen-elemen yang wajib diisi oleh pengguna. Aplikasi kemudian akan melakukan pengecekan terhadap nilai masukan dari pengguna, apakah nilai yang wajib ada sudah diisi atau belum. Jika nilai belum diisi maka aplikasi akan menolak form pengguna, dengan pesan kesalahan yang menjelaskan bahwa terdapat isian yang wajib diisi oleh pengguna. Proses pengecekan kewajiban isi atau berbagai batasan lain ini dikenal dengan nama validasi.

Validasi pada elemen-elemen form HTML dilakukan dengan beberapa cara, yaitu:

1. Atribut `required` pada elemen-elemen masukan yang wajib diisi. Jika pengguna tidak mengisi elemen maka browser secara otomatis akan memberikan peringatan kepada pengguna dan membatalkan pengiriman form. Contoh penggunaan atribut ini yaitu:

```
<input type="text" name="username" required>
```

2. Penggunaan atribut `type` yang benar, untuk memastikan browser dapat melakukan pengecekan nilai yang diisi pengguna dengan format sesuai dengan yang dispesifikasikan pada standar HTML. Misalnya, jika menggunakan `type="email"`, maka pengguna tidak dapat mengisi bukanemail sebagai masukan form.



Gambar 2.10 Contoh Pesan Kesalahan Pengisian Email pada HTML

LATIHAN

1. Penggunaan input text dalam sebuah form.

Simpan dengan nama **latihan14.html**

```
1 <html>
2   <head>
3     <title> Menyisipkan Elemen Berupa Text </title>
4   </head>
5   <body>
6     <font type=arial black font size="6" > Data Diri </font>
7     <br> </br>
8     <font ty=arial size ="5" >
9     nama : <input type = "text" name = "nama">
10    <br>
11    <br>
12    nim : <input type = "text" name = "nim" value="" maxlength="9" size="9">
13    </font>
14  </body>
15 </html>
```

Gambar 2.11. Contoh skrip untuk penggunaan input text dalam sebuah form

2. Penggunaan Input Text berupa Password.

Simpan dengan nama **latihan15.html**

```
1 <html>
2   <head>
3     <title> Input Password </title>
4   </head>
5   <body>
6
7     Password : <input type ="password" name="nama" size="11" maxlength="9">
8
9   </body>
10 </html>
```

Gambar 2.12. Contoh skrip untuk penggunaan input text dalam sebuah passwordmenu

3. Penggunaan Input text berupa radio button

Simpan dengan nama **latihan16.html**

```
1 <html>
2 <head>
3   <title> Penggunaan Radio Button </title>
4 </head>
5 <body>
6   Jenis Kelamin :
7   <br>
8   <input type="radio" name="jenis_kelamin" value="L"> Laki-laki
9   <br>
10  <input type="radio" name="jenis_kelamin" value="P"> Perempuan
11  <br>
12
13 </body>
14 </html>
```

Gambar 2.13. Contoh skrip untuk penggunaan input text berupa radio button

4. Penggunaan checkbox dalam HTML.

Simpan dengan nama **latihan17.html**

```
1 <html>
2 <head>
3   <title> Penggunaan Check Box </title>
4 </head>
5 <body>
6   Sebutkan Hobi Anda, boleh lebih dari satu :
7   <br>
8   <input type="checkbox" name="hobi_1" value="Menonton"> Menonton
9   <br>
10  <input type="checkbox" name="hobi_2" value="Game"> Bermain Game
11  <br>
12  <input type="checkbox" name="hobi_3" value="Shopping"> Belanja
13  <br>
14  <input type="checkbox" name="hobi_4" value="Mancing"> Memancing
15  <br>
16  <input type="checkbox" name="hobi_5" value="Olahraga"> Olahraga
17  <br>
18 </body>
19 </html>
```

Gambar 2.14. contoh skrip untuk penggunaan input text berupa checkbox

5. Penggunaan submit, reset, dan image sebagai button. Simpan dengan nama **latihan18.html**

```
1 <html>
2   <head>
3     <title> Konfigurasi Button </title>
4   </head>
5   <body>
6     <form>
7
8       <input type="submit" value="Kirim" name="Submit">
9       <input type="reset" value="Ulangi" name="Reset">
10      <input type="button" value="Batal" name="Cancel">
11      <input type="image" name="Gambar" img src="research.png" width="194" height="90">
12
13    </form>
14  </body>
15 </html>
```

Gambar 2.15. Contoh skrip penggunaan input berupa button

6. Penggunaan Text area dalam Form.
Simpan dengan nama **latihan19.html**

```
1 <html>
2   <head>
3     <title> Membuat Text Area </title>
4   </head>
5   <body>
6     Deskripsi Diri:
7     <br>
8     <textarea name="deskripsi" cols="45" rows="5" align="center"> Tuliskan deskripsi singkat tentang anda.
9   </textarea>
10  </body>
11 </html>
```

Gambar 2.16. Contoh skrip penggunaan input berupa text area

7. Penggunaan Select Dropdown dalam Form. Simpan dengan nama **latihan20.html**

```
1 <html>
2 <head>
3   <title> Menggunakan Menu Dropdown </title>
4 </head>
5 <body>
6   
7 </img>
8   <br>
9   Silahkan pilih orientasi minat :
10  <br>
11  <select name="Orientasi Minat">
12    <option value="dg"> Design Grafis </option>
13    <option value="pj"> Programmer Java </option>
14    <option value="sa"> Sistem Analis </option>
15    <option value="ds"> Dagelan Saja </option>
16  </select>
17  <br>
18  <br>
19  <select name="Orientasi Kemampuan" Multiple>
20    <option value="dg"> Design Grafis </option>
21    <option value="pj"> Programmer Java </option>
22    <option value="sa"> Sistem Analis </option>
23    <option value="aj"> Ahli Jaringan </option>
24    <option value="it"> Trainer IT </option>
25    <option value="ds"> Dagelan Saja </option>
26  </select>
27 </body>
28 </html>
```

Gambar 2.17. Contoh skrip penggunaan Select Dropdown.

MODUL 3

CASCADING STYLE SHEETS (CSS)

Tujuan :

1. Memahami tentang konsep CSS
1. Memahami aturan penulisan pada CSS
2. Memahami selector sebagai pengontrol design
3. Memahami pembuatan background
4. Memformat text pada web
5. Memahami pengaturan bentuk font
6. Membuat pengaturan tabel dan pewarnaannya
7. Membuat hyperlink dan tombol yang menarik

I. KONSEP

CSS Apa itu

CSS :

- Feature untuk membuat dynamic HTML.
- Style sheet mendeskripsikan bagaimana tampilan document HTML di layar (template)
- Membuat special efek (mendefinisikan style untuk <H1> dengan style bold dan italic dan berwarna biru)
- Support ke semua browser.

Aturan penulisan :

- Nilai untuk property tidak boleh dalam tanda petik. contoh : color : green;
- Nama property bersifat case sensitive. color : green;
property :
color value
: green

Cara penggunaan CSS :

- External Style Sheet

Bentuk :

```
<link rel="stylesheet" type="text/css"
      href="css_files.css">
```

dimana :

- <link, merupakan tag pembuka diakhiri dengan tanda ">"
- rel="stylesheet", menerangkan halaman ini akan dikenai efek style sheet
- type="text/css", file yang dipanggil berupa css
- href="css_files.css", alamat dokumen stylesheet yang dipanggil

Contoh Penggunaan :

script efek.css

```
body {
color: green;
background: yellow;
font-family : arial;
}
```

```
1 <HTML>
2 <HEAD>
3 <TITLE>Teknik Informatika</TITLE>
4 <LINK REL="stylesheet"
5 TYPE="text/css" HREF="efek.css">
6 </HEAD>
7 <BODY>
8 <H1>UIN Malang</H1>
9 <P>Teknik Informatika UIN Malang </P>
10 </BODY>
11 </HTML>
```



- Internal Style Sheet

Bentuk umum :

```
<style type="text/css">
...definisi style...
</style>
```

Contoh Penggunaan :

```
1 <HTML>
2 <HEAD>
3 <TITLE>centranet</TITLE>
4 <STYLE type="text/css">
5 body {
6 color: white;
7 background: green;
8 font-family : arial;
9 }
10 </STYLE>
11 </HEAD>
12 <BODY>
13 <H1>Teknik Informatiak</H1>
14 <P>Teknik Informatika UIN Malang berdiri tahun 2004</P>
15 </BODY>
16 </HTML>
```



- Inline Style sheet

Contoh Penggunaan :

```
1 <HTML>
2 <HEAD>
3 <TITLE>centranet</TITLE>
4 </HEAD>
5 <BODY style="color: white;
6 background: green;
7 font-family : arial;" >
8 <H1>TI_UIN MALANG</H1>
9 <P>MATERI 3 Pemrograman Web adalah CSS </P>
10 </BODY>
11 </HTML>
```



II. ATURAN PENULISAN PADA CSS

Syntaxis CSS dibagi dalam 3 bagian :

selector {property : value}

dimana :

selector : tag HTML yang akan didefinisikan (body, H1, Link , dll).

property : atribut yang akan diubah

Cara penulisan :

FONT-FAMILY : sans-serif;

FONT-SIZE : small;

Cara penulisan yang salah :

FONT-FAMILY : "sans-serif";

FONT-SIZE : 'small';

Cara mendeklarasikan kelompok : (tanda koma serta &)

H1, H2 {color : green};

H3, H4 & H5 {color : red};

Cara menuliskan komentar :

- Menggunakan tanda : /* */
- Menggunakan tanda : <!-- -->

Bentuk ukuran :

Ukuran	Keterangan
em	Untuk menentukan ukuran yang pecahan (desimal)
ex	x-height, digunakan untuk menentukan ukuran yang sifatnya vertikal
px	Pixels, menentukan ukuran yang bersifat pixel (layar monitor) seperti ukuran huruf.

III. SELECTOR SEBAGAI PENGONTROL DESIGN

1. Selector untuk merubah body.

<pre><HTML> <HEAD> <TITLE> Selector </TITLE> <style type="text/css"></pre>	<pre>color: red; BACKGROUND-COLOR: green; }</pre>
<pre>body { FONT-FAMILY: Geneva, Arial; FONT-SIZE: 20px;</pre>	<pre></style> </HEAD> <BODY> Halaman efect CSS </BODY> </HTML></pre>

2. Jenis-jenis selector

a. Selector bebas

```
<HTML>
  <HEAD>
    <TITLE> Selector </TITLE>
  </HEAD>
  <style type="text/css">
    gbawah{
    TEXT-DECORATION: underline;
    }
  </style>
</HTML>

<BODY>
<gbawah> Efect Garis Bawah dengan
Selector Bebas </gbawah>
</BODY>
</HTML>
```

b. Selector dengan class

```
<HTML>
  <HEAD>
    <TITLE> Selector </TITLE>
  </HEAD>
  <style type="text/css">
    .right { text-align : right }
  </style>
</HTML>

<BODY>
<h2 class="right">Class Heading 2 </h2>
<p class="right"> Class Paragraf</p>

</BODY>
</HTML>
```

c. Selector Id

```
<HTML>
  <HEAD>
    <TITLE> Selector </TITLE>
  </HEAD>
  <style type="text/css">
    #BODY_115 {
      FONT-SIZE: 20px;
      TEXT-DECORATION: underline;
      COLOR: blue;
      FONT-FAMILY:Comic Sans MS;
    }
  </style>
```

```
</HEAD>
```

```
<BODY id="BODY_115">
Menggunakan ID Selector
</BODY>
</HTML>
```

IV. MEMFORMAT HALAMAN WEB

1. Format dengan margin

```
<HTML>
  <HEAD>
    <TITLE>Pengaturan Margin</TITLE>
    <STYLE ="text/css">
      BODY
      {
        margin-top : 1cm;
        margin-right : 2cm;
        margin-bottom : 1cm;
        margin-left : 2cm;
      }
    </STYLE>
  </HEAD>
  <BODY>
    Pengaturan Margin Halaman
    (1cm,2cm,1cm,2cm)
  </BODY>
</HTML>
```

2. Pemetaan menggunakan padding

Padding hampir sama dengan margin :

- Margin hanya digunakan untuk membuat batasan-batasan sisi halaman.
- Padding dapat membuat batasan-batasan pada komponen web lain seperti tabel, disamping pengaturan batas halaman.

```
<HTML>
  <HEAD>
    <TITLE>Pengaturan Margin</TITLE>
    <STYLE ="text/css">
      BODY {
        padding-top : 10%;
        padding-right : 20%;
        padding-bottom : 40%;
        padding-left: 20%;
      }
    </STYLE>
  </HEAD>
  <BODY>
```



```

</STYLE>
</HEAD>
<BODY>
    Text ini berada di tengah halaman ,
    karna di lakukan pengaturan halaman
    dengan menggunakan padding atas 10%,
    kanan 20%,bawah 40%,dan kiri 20%.
</BODY>
</HTML>

```

V. MEMBUAT BACKGROUND

1. Background warna

```

<HTML>
<HEAD>
<TITLE>Menggunakan Background
Warna</TITLE>
<STYLE type="text/css">
    BODY { background-color : yellow}
</STYLE>
</HEAD>
<BODY>
    Halaman ini di buat Berwarna Kuning
</BODY>
</HTML>

```

2. Background campuran

```

<HTML>
<HEAD>
<TITLE>Menggunakan Background
Warna</TITLE>
<STYLE ="text/css">
    body {background-color : #99CCFF}
    h2 {background : green}
    h3 {background-color : transparent}
    p {background : rgb(240,248,255)}
</STYLE>
</HEAD>
<BODY>
<h2>Header 2,Background Hijao</h2>
<h3>Header 3 , Bakground Transparan</h3>
<p>Background pada paragraph</p>
</BODY>
</HTML>

```

3. Background gambar

Properties	Value	Keterangan
background-image	url	Alamat gambar yang dituju
background-repeat	repeat repeat-x repeat-y no-repeat	Diulang dlm hal Diulang sumbu x Diulang sumbu y Tampil 1 gbr
Background-position	top left top center top right center left center center center right bottom left bottom center bottom right x-% y-% x-pos y-pos	Atas kiri hal Atas tng hal Atas kanan hal Tgh kiri hal Pusat hal Tng kanan hal Bwh kiri hal Bwh tgh hal Bwh kanan hal Pakai nilai %

Contoh 1 :

```

<HTML>
<HEAD>
<TITLE>Menggunakan Background
Gambar</TITLE>
<STYLE ="text/css">
    BODY
    {
        background-image:
        url("drums.jpg");
        background-repeat: repeat-x;
    }
</STYLE>
</HEAD>
<BODY>
    Background Berulang pada Sumbu X
</BODY>
</HTML>

```

Contoh 2 :

```

<HTML>
<HEAD>
<TITLE>Menggunakan Background
Gambar</TITLE>
<STYLE ="text/css">
    BODY
    {
        background-image:url ("motor.jpg");
        background-repeat: no-repeat;
        background-position: center center;
    }

```

```

</STYLE>
</HEAD>
<BODY>
    Background di Pusat Halaman
</BODY>
</HTML>

```

VL FORMAT TEXT PADA WEB

Properties	Value	Keterangan
Pengaturan warna		
color	green, dll	
Pengaturan Spasi (jrk antar karakter)		
letter-spacing	Normal Length	Ukm standar HTML Ukm panjang (cm,px)
Perataan Text		
text-align	left right center justify	
text-decoration	none underline overline line-through blink	Bentuk standar Bergaris bwh Bergaris atas Text dicoret Text berkedip
Pengaturan text indentasi		
text-indent	length %	Dengn cm, px Dengan persentase
Pengubahan Bentuk Karakter		
text-transform	capitalize uppercase lowercase none	

Contoh 1 :

```

<HTML>
<HEAD>
<TITLE>Format Text </TITLE>
<STYLE ="text/css">
    p {color : green; letter-spacing: 0.5cm}
    h4 {letter-spacing: -2px}
</STYLE>
</HEAD>
<BODY>
    <p>Pengaturan Spasi Pada Paragraph
    </p>
    <h4> Header 4</h4>
</BODY>
</HTML>

```

Contoh 2 :

```

<HTML>
<HEAD>
<TITLE>Format Text </TITLE>
<STYLE ="text/css">
    h1 {text-align: center}
    h2 {text-align: left}
    h3 {text-align: right}

```

```

</STYLE>
</HEAD>
<BODY>
    <h1>Header 1,Di tengah</h1>
    <h2>Header 2 , Di kiri</h2>
    <h3>Header 3 ,Di kanan</h3>
</BODY>
</HTML>

```

Contoh 3 :

```

<HTML>
<HEAD>
<TITLE>Format Text </TITLE>
<STYLE ="text/css">
    em {text-decoration : overline}
    h2 {text-decoration: blink}
    h3 {text-decoration: underline}
    a {text-decoration: none}

```

```

</STYLE>
</HEAD>
<BODY>
    <em>Bentuk Overline</em>
    <h2>Header 2, Bentuk Line-
    through</h2>
    <h3>Header 3,Bentuk Underline</h3>
    <p> <a href="http://lecturer.eepis-
    its.edu/~zenhadi">
    Penggunaan Dalam Link,Nilai NONE</a></p>
</BODY>
</HTML>

```

Contoh 4 :

```

<HTML>
<HEAD>
<TITLE>Format Text </TITLE>
<STYLE ="text/css">
    p.besar {text-transform: uppercase}

```

```

        p.kecil {text-transform: lowercase}
</STYLE>
</HEAD>
<BODY>
<p class="besar">
pengubahan kedalam hurup Besar
</p>
<p class="kecil">
PENGUBAHAN KEDALAM HURUF KECIL
</p>
</BODY>
</HTML>

```

VII. PENGATURAN FONT

Properties	Value	Keterangan
Jenis Font		
font-family	arial, dll	
Ukuran Huruf		
Font-size	Small Medium Large Length %	Kecil Menengah Besar Besar font (pt,px) Persentase
Pengaturan gaya pada font		
font-style	normal italic oblique	
Ketebalan huruf		
font-weight	normal bold 100 ~ 900	

Contoh 1 :

```

<HTML>
<HEAD>
<TITLE>Pengaturan Font</TITLE>
<STYLE ="text/css">
    p.italic
    {
    font-size :200 % ;
    font-style: italic;
    }
    p.normal{
    font-family : verdana ;
    font-style: normal;
    }
    p.oblique {font-style: oblique}
</STYLE>
</HEAD>

```

```

<BODY>
<P class="italic">Menggunakan Style
Italic</P>
<P class="normal">Menggunakan Style Normal
</P>
<P class="oblique">Menggunakan Style
Oblique</P>
</BODY>
</HTML>

```

Contoh 2 :

```

<HTML>
<HEAD>
<TITLE>Pengaturan Font</TITLE>
<STYLE ="text/css">
    p.normal
    {
    font-family : verdana ;
    font-weight: normal;
    }
    p.thick
    {
    font-family : verdana ;
    font-weight: bold;
    }
    p.thicker
    {
    font-family : times ;
    font-weight: 900;
    }
</STYLE>
</HEAD>
<BODY>
    <p class="normal">
    This is a paragraph</p>
    <p class="thick">
    This is a paragraph</p>
    <p class="thicker">
    This is a paragraph</p>
</BODY>
</HTML>

```

VIII. PENGATURAN TABEL

Yang perlu diperhatikan adalah pengaturan border, padding dan margin suatu tabel.

1. Pengaturan padding

```
<HTML>
<HEAD>
<TITLE>Pengaturan Padding Table</TITLE>
<style type="text/css">
    td.kiri{
        padding-top: 2cm;
        padding-right: 2cm;
        padding-bottom: 2cm;
        padding-left: 2cm ;
        background-color : #F0F8FF;
    }
</style>
</HEAD>
<BODY>
<TABLE BORDER="1">
<TR>
<TD class="kiri">Padding dengan jarak 2cm
dari kiri,atas,kanan,dan bawah</TD>
<TD>Tanpa jarak </TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

2. Menggunakan file css untuk mempercantik pembuatan tabel.

File `table_specbody.css`

```
/* CSS Document */

TH {
    color : #FFFFFFF;
    background-color : #336699;
    border-width: 1px ;
    border-style:solid;
    border-color :red ;
    font-size: 9pt;
}
TD {
    color : red;
```

```
background-color : #E6E6FA;
border-width: 1px ;
border-style:solid;
border-color :blue ;
font-size: 9pt;
}
```

File html yang akan memanggil file css

```
<HTML>
<HEAD>
<TITLE>Pengaturan Table spec</TITLE>
<LINK REL="STYLESHEET"
TYPE="text/css"
HREF="table_specbody.css">

</HEAD>
<BODY>
<table width="468" border="0"
cellpadding="5" cellspacing="0" >
<tr>
<th width="112" >Nama Computer</th>
<th width="91">Prosesor</th>
<th width="96">Ram</th>
<th width="137" >System Operasi</th>
</tr>
<tr>
<td>Cint-1</td>
<td >IP 4 1,8 Ghz</td>
<td >10 GB</td>
<td >Redhat Linux</td>
</tr>
<tr>
<td >Cint-2</td>
<td >IP 2 Ghz C</td>
<td >6 GB</td>
<td >Mandrake Linux</td>
</tr>
<tr>
<td >Cint-2</td>
<td >XP 2000+ </td>
<td >20 GB</td>
<td>Windows</td>
</tr>
</table>
</BODY>
</HTML>
```

```
        color:#EDEDED;  
        text-decoration:none;  
    }  
    .leftlinks a:link {  
        color:#EDEDED;  
    }  
    .leftlinks a:visited {  
        color:#EDEDED;
```

```
</BODY>  
</HTML>
```

Fungsi tag <div> : hampir sama dengan tag paragraf <p>, berguna untuk mengelompokkan sejumlah baris teks yang memiliki karakteristik yang sama.


```

}
.leftlinks a:hover { display:block; margin:3px 0px;
border-top:1px solid #22476C; border-left:1px solid #22476C; border-
bottom:1px solid #4C86C0; border-right:1px solid #4C86C0; padding: 5px 9px
3px 11px;
text-align:center; background-color:#336699; color:#EED929;
text-decoration:none;
}

```

b. Pembuatan file HTML utk memanggil css

```

<HTML>
<HEAD>
<TITLE>Pengaturan Pada Link Visualisasi tombol</TITLE>

<LINK REL="STYLESHEET" TYPE="text/css"
HREF="link_frtombol.css">

</HEAD>
<BODY>

<div class="leftlinks">
<a href="index.htm">Teras Wamika</a>
<a href="profil.htm">Pssrofil Organisasi</a>
<a href="struktur.htm">Struktur</a>
<a href="pembimbing.htm">Pembimbing
Org</a>
<a href="dpo.htm">DPO Organisasi</a>
<a href="kegiatan.htm">Kegiatan Rutin</a>
<a href="berita.htm">Berita UKM</a>
</div>

```

TUGAS :

Buatlah web pribadi anda, lengkapi dengan pembuatan tombol link menggunakan css.

FASILITAS MENU :

1. HOME : Informasi sekilas tentang diri anda.
2. PENDIDIKAN : Pendidikan yang telah anda tempuh, termasuk pelatihan dan training yang pernah diikuti.
3. JADWAL KULIAH : Gunakan pemakaian tabel dengan css seperti diatas.
4. AKTIVITAS : Pengalaman organisasi yang pernah anda ikuti selama kuliah atau di luar.
5. Tambahkan informasi lain, untuk menambah nilai anda.

MODUL 4

Java Scripts I

4.1 Tujuan

- Mahasiswa dapat mengoperasikan struktur java scripts
- Mahasiswa dapat memakai object dan form pada java scripts

4.2 Materi

- Java Scripts
- Pemrograman
- Form

4.3 Alat dan Bahan

- Sublime Text
- Page, Image

4.4 Prosedur Praktikum

- Peserta membaca dan mempelajari materi praktikum sebelumnya
- Instruktur menerangkan dan menjelaskan teori dan cara kerja
- Peserta mempraktikkan materi percobaan
- Peserta membuat penyelesaian terhadap soal latihan

4.5 Teori

Sekilas tentang JavaScript

Javascript adalah bahasa skrip yang ditempelkan pada kode HTML dan diproses di sisi klien. Dengan adanya bahasa ini, kemampuan dokumen HTML menjadi semakin luas. Sebagai contoh, dengan menggunakan JavaScript dimungkinkan untuk memvalidasi masukan-masukan pada formulir sebelum formulir dikirimkan ke server. Javascript bukanlah bahasa Java dan merupakan dua bahasa yang berbeda. Javascript diinterpretasikan oleh klien (kodenya bisa dilihat pada sisi klien), sedangkan kode java dikompilasi oleh pemrogram dan hasil kompilasinyalah yang dijalankan oleh klien.

Struktur JavaScript

Struktur dari JavaScript adalah sbb :

```
<SCRIPT LANGUAGE = "JavaScript">  
<!--
```

Penulisan kode javascript

```
// -->
```

```
</SCRIPT>
```

Keterangan :

Kode <!-- // -->

umumnya disertakan dengan tujuan agar sekiranya browser tidak mengenali JavaScript maka browser akan memperlakukannya sebagai komentar sehingga tidak ditampilkan pada jendela browser.

JavaScript sebagai bahasa berorientasi pada objek

Properti

Properti adalah atribut dari sebuah objek. Contoh, objek mobil punya properti warna mobil.

Penulisan :

```
Nama_objek.nama_properti = nilai
```

```
window.defaultStatus = "Selamat Belajar JavaScript";
```

Metode

Metode adalah suatu kumpulan kode yang digunakan untuk melakukan sesuatu tindakan terhadap objek.

Penulisan :

Nama_objek.nama_metode(parameter)
document.write ("Hallo")

Letak JavaScript dalam HTML

Skrip Javascript dalam dokumen HTML dapat diletakkan pada :

1. Bagian Head
2. Bagian Body (jarang digunakan).

Latihan :

DASAR-DASAR JAVASCRIPT

1. Pemakaian alert sebagai property window
2. Pemakaian metode dalam objek.
3. Pemakaian prompt

```
1 <HTML>
2 <HEAD>
3 <TITLE>Alert Box</TITLE>
4 </HEAD>
5 <BODY>
6 <SCRIPT LANGUAGE = "JavaScript">
7 //<!--
8 window.alert("Ini merupakan pesan untuk Anda");
9 //-->
10 </SCRIPT>
11 </BODY>
12 </HTML>
13
```

```
1 <HTML>
2 <HEAD>
3 <TITLE>Skrip JavaScript</TITLE>
4
5 </HEAD>
6 <BODY>
7 Percobaan memakai JavaScript:<BR>
8
9 <SCRIPT LANGUAGE = "JavaScript">
10 //<!--
11
12 document.write("Selamat Mencoba JavaScript<BR>"); document.write("Semoga sukses!");
13 //-->
14 </SCRIPT>
15
16 </BODY>
17 </HTML>
```

```
1 <HTML>
2
3 <HEAD>
4 <TITLE>Pemasukan Data</TITLE>
5 </HEAD>
6
7 <BODY>
8 <SCRIPT LANGUAGE = "JavaScript">
9 //<!--
10
11 var nama = prompt("Siapa nama Anda?", "Masukkan nama anda"); document.write("Hai, " + nama);
12 //-->
13
14 </SCRIPT>
15 </BODY>
16 </HTML>
```

```

1  <HTML>
2  <HEAD>
3
4  <TITLE>Contoh Program Javascript</TITLE>
5  </HEAD>
6  <SCRIPT language="Javascript">
7
8  function pesan(){
9
10 alert ("memanggil javascript lewat body onload")
11 }
12
13 </SCRIPT>
14 <BODY onload=pesan()>
15 </BODY>
16
17 </HTML>

```

```

1
2  <HTML>
3  <HEAD>
4
5  <TITLE>Contoh Program Javascript</TITLE>
6  </HEAD>
7  <SCRIPT language="Javascript">
8
9  function test (val1,val2)
10 {
11 document.write("<br>"+"Perkalian : val1*val2 "+"<br>")
12
13 document.write(val1*val2)
14 document.write("<br>"+"Pembagian : val1/val2 "+"<br>")
15 document.write(val1/val2)
16
17 document.write("<br>"+"Penjumlahan : val1+val2 "+"<br>")
18
19 document.write(val1+val2)
20 document.write("<br>"+"Pengurangan : val1-val2 "+"<br>")
21
22 document.write(val1-val2)
23 document.write("<br>"+"Modulus : val1%val2 "+"<br>")
24 document.write(val1%val2)
25
26 }
27 </SCRIPT>
28 <BODY>
29
30 <input type="button" name="button1" value="arithmetic"
31 onclick=test(9,4)>
32 </BODY>
33
34 </HTML>

```

Operasi relational

Seleksi kondisi (if..else)


```

1 <HTML>
2 <HEAD>
3 <TITLE>Contoh Program Javascript</TITLE>
4 </HEAD>
5 <SCRIPT language="Javascript">
6 function test () {
7     val1=window.prompt("Nilai I :")
8     val2=window.prompt("Nilai II :")
9     document.write("<br>"+"val1==val2"+"<br>")
10    document.write(val1==val2)
11    document.write("<br>"+"val1!=val2"+"<br>")
12    document.write(val1!=val2)
13    document.write("<br>"+"val1>val2"+"<br>")
14    document.write(val1>val2)
15    document.write("<br>"+"val1<val2"+"<br>")
16    document.write(val1<val2) }
17 </SCRIPT>
18 <BODY>
19 <input type="button" name="button1" value="relational" onclick=test()>
20 </BODY>
21 </HTML>

```

```

1 <HTML>
2 <HEAD>
3 <TITLE>Contoh Program Javascript</TITLE>
4 </HEAD>
5 <SCRIPT LANGUAGE = "JavaScript">
6 //<!--
7 var nilai = prompt("Nilai (0-100): ", 0);
8 var hasil = "";
9 if (nilai >= 60)
10 hasil = "Lulus";
11 else
12 hasil = "Tidak Lulus";
13 document.write("Hasil: " + hasil);
14 //-->
15 </SCRIPT>
16 </BODY>
17 </HTML>

```

Penggunaan operator switch untuk seleksi kondisi *Pemakaian looping <for>*

```

1  <HTML>
2  <HEAD>
3  <TITLE>Contoh Program Javascript</TITLE>
4  </HEAD>
5  <SCRIPT language="javascript">
6  function test ()
7  {
8
9  val1=window.prompt("Input Nilai (1-5):")
10 switch (val1)
11 {
12
13 case "1" :
14 document.write("bilangan satu")
15 break
16
17 case "2" :
18 document.write("bilangan dua")
19 break
20
21 case "3" :
22 document.write("bilangan tiga")
23 break
24
25 case "4" :
26 document.write("bilangan empat")
27 break
28
29 case "5" :
30 document.write("bilangan lima")
31 break
32
33 default :
34 document.write("bilangan lainnya")
35 }
36
37 }
38 </SCRIPT>
39 <BODY>
40 <input type="button" name="button1" value="switch"
41 onclick=test()>
42 </BODY>
43 </HTML>

```

```

1  <HTML>
2  <HEAD>
3
4  <TITLE>Contoh Program Javascript</TITLE>
5  </HEAD>
6  <BODY>
7
8  <SCRIPT language="javascript">
9  // | -
10 for (x=0;x<=10;x++)
11
12 document.write(x+"<br>")
13
14
15 // -> </SCRIPT>
16 </BODY>
17 </HTML>

```

Pemakaian looping < do..while >

Pemakaian looping < while >

```

1 <HTML>
2
3 <HEAD>
4 <TITLE>Contoh Program Javascript</TITLE>
5 </HEAD>
6
7 <BODY>
8 <SCRIPT language="javascript">
9 //<!--
10
11 var x=0
12 do{
13 document.write(x+"<br>")
14
15 x++;
16 }
17 while (x<=10)
18
19 // --> </SCRIPT> </BODY>
20 </HTML>

```

Form

1. Form input :

2. Form button :

```

1 <HTML>
2 <head> </head>
3 <SCRIPT language="javascript">
4
5 function test () {
6 var val1=document.kirim.T1.value
7 if (val1%2==0)
8
9 document.kirim.T2.value="bilangan genap"
10 else
11 document.kirim.T2.value="bilangan ganjil"
12
13 }
14 </SCRIPT>
15 <body>
16
17 <form method="POST" name="kirim">
18
19 <p>BIL <input type="text" name="T1" size="20"> MERUPAKAN BIL <input type="text" name="T2" size="20"> </p>
20
21
22 <p><input type="button" value="TEBAK" name="B1" onclick=test()> </p>
23
24 </form>
25 </body>
26 </html>

```

```

1 <HTML>
2 <HEAD>
3 <TITLE>Objek document</TITLE>
4 </HEAD>
5 <BODY>
6 <SCRIPT LANGUAGE = "JavaScript">
7
8 //<!--
9 function ubahWarnaLB(warna) {
10 document.bgColor = warna;
11
12 }
13 function ubahWarnaLD(warna) {
14 document.fgColor = warna;
15 }
16 //-->
17 </SCRIPT>
18
19 <H1>TES</H1>
20 <FORM>
21 <INPUT TYPE = "BUTTON"
22
23 VALUE = "Later Belakang Hijau"
24 onClick = "ubahWarnaLB('GREEN')">
25 <INPUT TYPE = "BUTTON"
26
27 VALUE = "Later Belakang Putih"
28 onClick = "ubahWarnaLB('WHITE')">
29 <INPUT TYPE = "BUTTON"
30
31 VALUE = "Teks Kuning"
32 onClick = "ubahWarnaLD('YELLOW')">
33 <INPUT TYPE = "BUTTON"
34
35 VALUE = "Teks Biru"
36 onClick = "ubahWarnaLD('BLUE')">
37 </FORM></BODY></HTML>

```

4.6 Latihan

1. Buat halaman html untuk mengkonversi nilai angka menjadi nilai huruf dengan menggunakan javascript . Konversi : 0-40 =E
41-55=D
56-60=C
61-65=BC
66-70=B
71-80=AB
81-100=A
2. Buat halaman html untuk menampilkan aplikasi program kalkulator sederhana dengan menggunakan javascript. Contoh tampilan :
Bil 1 dan Bil 2 merupakan text box, dapat diisi angka, bila tombol + atau - atau x atau / ditekan, maka akan keluar bilangan pada text box hasil, dimana bilangan ini merupakan operasi arithmetic sesuai dengan tombol yang ditekan.
`<SCRIPT LANGUAGE = "JavaScript">`
`<!--`
`document.write("Dimodifikasi terakhir pada " +`
`document.lastModified);`
`//--> </SCRIPT>`
`</BODY>`
`</HTML>`

MODUL 5

Java Scripts 2 (jQuery)

5.1 Tujuan :

- Mahasiswa dapat memahami mengenai *HTML dengan jQuery*
- Mahasiswa dapat mengaplikasikan penggunaan *jQuery* untuk pembuatan *template website* agar lebih menarik.

5.2 Materi :

- HTML
- jQuery
- CSS

5.3 Dasar Terori jQuery :

jQuery adalah Javascript Library → Kumpulan kode/fungsi Javascript siap pakai, sehingga mempermudah dan mempercepat kita dalam membuat kode Javascript. Secara standar, apabila kita membuat kode Javascript, maka diperlukan kode yang cukup panjang, bahkan terkadang sangat sulit dipahami, misalnya fungsi Javascript untuk membuat stripe (warna selang seling) pada baris suatu table adalah sebagai berikut :

```
function stripe(id) {
    var even = false;
    var evenColor = arguments[1] ? arguments[1] : "#fff";
    var oddColor = arguments[2] ? arguments[2] : "#eee";
    var table = document.getElementById(id); if (! table)
    {return; }
    var tbodies = table.getElementsByTagName("tbody");
    for (var h = 0; h < tbodies.length; h++) {
        var trs = tbodies[h].getElementsByTagName("tr");
        for (var i = 0; i < trs.length; i++) {
            if (! hasClass(trs[i]) && ! trs[i].style.backgroundColor){
                var tds = trs[i].getElementsByTagName("td"); for (var j =
                0; j < tds.length; j++) {
                    var mytd = tds[j];
                    if (! hasClass(mytd) && ! mytd.style.backgroundColor) {
                        mytd.style.backgroundColor = even ? evenColor : oddColor;
                    }
                }
            }
        }
    }
}
```

Nah, disinilah peran jQuery sebagai Javascript Library, dimana kita bisa langsung memanggil fungsi yang terdapat didalam Library tersebut, menariknya kita hanya memerlukan satu/dua baris membuat warna selang – seling pada suatu table, yaitu :

```
jQuery('table tr:nth-child(odd)').addClass('odd');
```

Kesimpulannya, JQuery menyederhanakan kode javascript.

1. Kemampuan jQuery

- **Mempermudah akses dan manipulasi elemen tertentu pada dokumen**
Biasanya diperlukan baris program yang cukup panjang untuk mengakses suatu elemen dokumen. Namun, jQuery dapat melakukannya hanya dalam beberapa baris program saja, karena jQuery mempunyai selector yang sangat efisien untuk mengakses suatu elemen tertentu pada dokumen yang selanjutnya bisa dimanipulasi sesuai dengan keinginan kita.
- **Mempermudah modifikasi/perubahan tampilan halaman web**
Biasanya untuk memodifikasi tampilan halaman web digunakan CSS. Permasalahannya, CSS sangat dipengaruhi oleh web browser yang digunakan sehingga sering terjadi halaman web yang dibuat sudah rapi dan bagus tampilannya di browser Mozilla, namun ketika ditampilkan dioperaa menjadi berantakan. Namun jQuery dapat menyesuaikan style CSS pada semua browser, sehingga permasalahan tersebut dapat dihindari.
- **Mempersingkat Ajax (Asynchronous Javascript and XML)**
Kemampuan Favorit dari Ajax adalah mampu mengambil informasi dari server tanpa melakukan refresh pada halaman web, artinya halaman web terlihat berganti secara otomatis. Apabila kita menuliskan kode Ajax secara manual, biasanya diperlukan baris yang cukup panjang, namun jQuery dapat mempersingkatnya menggunakan ajax call, perbandingannya 25 baris kode ajax dapat disingkat menjadi 5 baris kode saja dengan jQuery.
- **Memiliki API(Application Programming Interface)**
Dengan API, jQuery dapat memanipulasi content pada suatu halaman web, seperti pengubahan teks, memanipulasi gambar (resize, rotate, crop), penyusunan daftar (list), pengurutan list, paging, dan lain – lain.
- **Mampu merespon interaktif antara user dengan halaman web dengan lebih cepat**
- **Menyediakan fasilitas untuk membuat animasi sekelas Flash dengan mudah**

2. Menggunakan jQuery

1. Download library jQuery di <http://jquery.com> silahkan pilih versi jQuery yang diinginkan, misalnya Development, kemudian klik tombol Download, nanti nama file yang terdownload adalah jquery-1.4.js
2. Simpan file jquery-1.4.js pada folder latihan anda, misalnya di C:\latih. Kemudian buat dokumen baru yang isinya seperti berikut :

```
<html>
<head>
</head>
<body>
</body>
</html>
```

3. Selanjutnya panggil library jQuery diantara tag <head>...</head>, maka sekarang skripnya menjadi (perhatikan teks yang ditebalkan):

```
<html>
<head>
<b><script type = "text/javascript" src="jquery 1.4.js">
</script>
</head>
<body>
</body>
```

```
</html>
```

4. Sebagai contoh, kita akan membuat dua buah tombol yang akan mengontrol sebuah foto, contoh skripnya adalah (perhatikan teks yang ditebalkan):

```
<html>
<head>
<script type = "text/javascript" src="jquery-1.4.js">
</script>
</head>
<body>
<bbutton
class="sembunyi">sembunyikan</bbutton
> <bbutton
class="tampil">tampilkan</bbutton>
<p></p> </body>
</html>
```

5. Selanjutnya, buat skrip jQuery dan sesuaikan dengan id atau class yang telah kita definisikan pada langkah 4. Skripnya sebagai berikut (perhatikan teks yang ditebalkan)

```
1 <html>
2 <head>
3 <script type = "text/javascript" src="jquery-1.4.4.js"></script>
4 <script type="text/javascript">
5 $(document).ready(function(){
6 $(".sembunyi").click(function() { $( "#foto" ).hide("slow");
7 $(".tampil").click(function() { $( "#foto" ).show("slow");
8 });
9 });
10 });
11 </script>
12 </head>
13 <body>
14 <bbutton class="sembunyi">sembunyikan</bbutton> <bbutton class="tampil">
tampilkan</bbutton> <p></p> </body>
15 </html>
```

Simpan filenya, misalkan dengan nama pertama.html, untuk menjalankannya, anda bisa double klik pada nama filenya.

3. Cara Kerja jQuery

- a. jQuery akan memastikan bahwa semua elemen atau elemen yang dibagikan sudah ditampilkan semua dihalaman web, fungsi yang digunakan adalah :

```
$document.ready(function() {
//baris kode jQuery akan dijalankan
//apabila semua elemen sudah ditampilkan semua
})
```

- b. Setelah semua elemen ditampilkan, tahap berikutnya adalah memilih elemen berdasarkan class atau id yang telah didefinisikan. Dalam hal ini, jQuery menggunakan fungsi Selector. Contoh skripnya :

```

$("#foto")
$(".sembu
nyi")
$(".tampi
1")

```

- c. Setelah elemen dipilih, tahap berikutnya adalah memberikan aksi/operasi terhadap elemen yang sudah dipilih. Misalnya, operasi klik dan hide, dimana apabila tombol diklik, maka akan menyembunyikan elemen foto.

```

$(".sembunyi").click(function(){
    $("#foto").hide("slow") });

```

4. Memahami selectors

Selectors berfungsi untuk memilih dan mengambil elemen – elemen tertentu yang ingin dilakukan operasi terhadap elemen tersebut. Atau manipulasi terhadap elemen – elemen tersebut.

- a. **selector Nama tag** -> cara menggunakan selector tag adalah dengan langsung menyebut nama tag elemennya contoh :\$("div") untuk lebih jelasnya perhatikan skrip berikut :

```

1  <html>
2  <head>
3  <script type="text/javascript" src="jquery-1.4.4.min.js"></script>
4  <script type="text/javascript">
5  $(document).ready(function(){
6  $("div").addClass("biru");});
7  </script>
8  <style type="text/css">
9  .biru {
10 color: blue;
11 font-weight: bold;
12 }
13 </style>
14 </head>
15 <body>
16 <div>
17 <p>paragraf pertama</p>
18 </div>
19 <div>
20 <ul>
21 <li>item 1</li>
22 <li>item 2</li>
23 <li>item 3</li>
24 </ul>
25 </div>
26 </body>
27 </html>

```

b. Selectors id

cara penggunaan selectors id adalah menyertakan tanda kres(#) sebelum nama elemennya -> \$("#nama_elemen_id"). Untuk lebih jelasnya perhatikan skrip berikut :

```
1 <html>
2 <head>
3 <script type="text/javascript" src="jquery-1.4.4.min.js"></script> <script type=
"text/javascript">
4 $(document).ready(function(){
5 $("#daftar").addClass("merahkuning");});</script>
6 <style type="text/css">
7 .merahkuning {
8 color: orange;
9 font-weight: bold;
10 background-color: green;
11 padding: 3px;
12 }
13 </style>
14 </head>
15 <body>
16 <div>
17 <p>paragraf pertama</p>
18 </div>
19 <div>
20 <ul id="daftar">
21 <li>item 1</li>
22 <li>item 2</li>
23 <li>item 3</li>
24 </ul>
25 </div>
26 </body>
27 </html>
```

c. Selectors Class

cara menggunakan selectors class adalah dengan menyertakan tanda titik(.) sebelum nama elemennya -> \$(".nama_elemen_class"). Untuk lebih jelasnya perhatikan skrip berikut :


```

1 <html>
2 <head>
3 <script type="text/javascript" src="jquery-3.3.1.min.js"></script> <script type="text/
  javascript">
4 $(document).ready(function(){
5 $("#daftar").addClass("merahkuning");
6 $(".subdaftar").addClass("kuninghitam");});</script>
7 <style type="text/css">
8 .merahkuning {
9   color: red;
10  font-weight: bold;
11  background-color: yellow;
12  padding: 3px;
13 }
14 .kuninghitam{
15   color : blue;
16   background-color: green;
17 }
18 </style>
19 </head>
20 <body>
21 <div>
22 <p>paragraf pertama</p>
23 </div>
24 <div>
25 <ul id="daftar">
26 <li>item 1</li>
27 <li class="subdaftar">item 2</li>
28 <li>item 3</li>
29 </ul>
30 </div>
31 </body>
32 </html>

```

Keterangan : tag “li” dengan class “subdaftar” akan terpilih oleh selectors, kemudian diberikan operasi “kuninghitam” terhadap elemen terpilih tersebut, sehingga semua teks yang berada dalam tag li class subdaftar akan berwarna kuning dan tebal dengan background hitam.

5. Memahami Events

Sebelumnya kita telah belajar selectors yang mencari elemen – elemen apa saja yang akan dilakukan operasi terhadapnya, maka events memutuskan kapan dilakukan operasi tersebut, misalnya elemen yang telah dipilih (selectors) akan dilakukan operasi terhadapnya setelah diklik(clik) atau setelah klik 2x(dbclick), setelah mouse berada diatas objek(hover) dsb. Contoh event

```

1 <html>
2 <head>
3 <script type="text/javascript" src="jquery-1.4.4.min.js"></script>
4 <script type="text/javascript">
5 $(document).ready(function(){
6 $("a").click(function(){
7   alert("selamat datang di website facebook");
8 });
9 });
10 </script>
11 </head>
12 <body>
13 <a href="http://facebook.com">
14 klik disini untuk membuka website facebook</a> </body>
15 </html>

```


6. Memahami Effects

Effects mengenai tentang menampilkan dan menyembunyikan suatu elemen, menggerakkan elemen, dan menganimasi elemen.

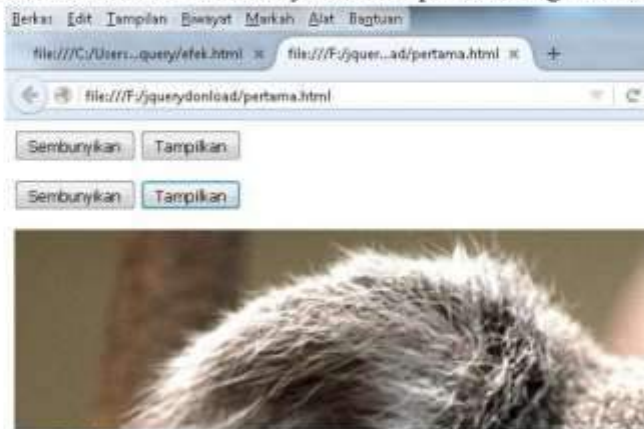
Pada sub bab sebelumnya kita telah menggunakan event klik pada tombol, dimana apabila tombol sembunyikan diklik, maka elemen berupa foto akan hilang dan apabila diklik tombol tampilkan maka foto tersebut akan ditampilkan kembali. Pada contoh tersebut kita telah menggunakan dua buah effect yaitu show dan hide.

```
1 <html>
2 <head>
3 <script type="text/javascript" src="jquery-1.4.4.min.js"></script>
4 <script type="text/javascript">
5 $(document).ready(function(){
6   setTimeout(function(){
7     $("#kotak").fadeOut("slow",function(){
8       $("#kotak").remove();
9     });
10   }, 3000);
11 });
12 </script>
13 <style type="text/css">
14 #kotak {
15   width : 300px;
16   height : 50px;
17   background: lightyellow;
18   border : 1px solid black;
19   padding : 5px;
20 }
21 </style>
22 </head>
23 <body>
24 <h3 id="kotak">
25 Perhatikan baik -baik teks ini, karena
26 teks akan hilang dalam hitungan 3 detik</h3>
27 </body>
28 </html>
```

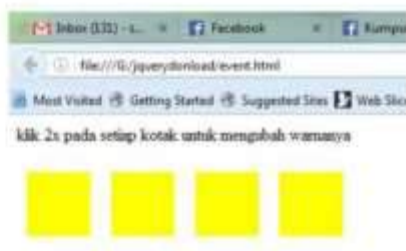
5.4 TUGAS

Buatlah seperti tampilan dibawah ini

1. Buatlah tombol sembunyi dan tampilkan dengan beberapa gambar



2. Buatlah menggunakan event double klik, ketika pada salah satu kotak di dobel klik maka warna pada kotak bisa berubah



3. Buatlah menggunakan efek fade ()



MODUL 6

Java Scripts (AJAX)

6.1 Tujuan :

- Mahasiswa dapat memahami mengenai *HTML dengan Ajax*.
- Mahasiswa dapat mengaplikasikan penggunaan *Ajax* untuk pembuatan *template website* agar lebih menarik.

6.2 Materi :

- HTML
- Ajax
- CSS

6.3 Latihan :

latihan_01.js

```
function createXMLHttpRequest() {
    var xmlhttp = false;

    try {
        xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try {
            xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (e) {
            xmlhttp = false;
        }
    }

    if (!xmlhttp && typeof XMLHttpRequest != 'undefined') {
        xmlhttp = new XMLHttpRequest();
    }

    if (!xmlhttp) {
        alert("Terjadi kesalahan saat pembuatan XMLHttpRequest object!");
    } else {
        alert("XMLHttpRequest berhasil dibuat.");
    }

    return xmlhttp;
}
```

Program tersebut diatas berguna untuk membuat sebuah object XMLHttpRequest. Pertama kali program akan membuat object XMLHttpRequest untuk browser Internet Explorer versi 7 keatas melalui statement:

```
try {
    xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
}
```

Jika terjadi kegagalan atau error dalam pembuatan object, program akan melakukan antisipasi dengan cara membuat object XMLHttpRequest untuk browser Internet Explorer versi 6 kebawah melalui statement

```

catch (e) {
    try {
        xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
    } catch (e) {
        xmlhttp = false;
    }
}

```

Jika pembuatan object XMLHttpRequest masih juga gagal, maka dapat disimpulkan bahwa browser yang digunakan bukanlah Internet Explorer. Selanjutnya akan dibuat object XMLHttpRequest untuk browser lain (Firefox, Safari, Opera atau Mozilla based) melalui statement:

```

if (!xmlhttp && typeof XMLHttpRequest != 'undefined') {
    xmlhttp = new XMLHttpRequest();
}

```

Bagian skrip program selanjutnya akan memeriksa kondisi dari object XMLHttpRequest.

```

if(!xmlhttp) {
    alert("Terjadi kesalahan saat pembuatan XMLHttpRequest object!");
} else {
    alert("XMLHttpRequest berhasil dibuat.");
}

```

Jika variabel xmlhttp yang merupakan perwujudan object XMLHttpRequest masih bernilai false, maka kemungkinan besar browser yang digunakan bukanlah 5 browser yang disebutkan diatas atau browser tersebut belum mendukung object XMLHttpRequest.

Sekarang, marilah kita coba gabungkan antara skrip latihan sebelumnya dengan file HTML untuk membuat sebuah object XMLHttpRequest.

latihan_02.html

```

<html>
<head>
    <title>Pembuatan Object XMLHttpRequest</title>
    <script language="JavaScript" src="latihan_01.js"></script>
</head>
<body onload="createXMLHttpRequest();" >
    Halaman ini memanggil fungsi createXMLHttpRequest() yang berguna untuk
    membuat object XMLHttpRequest.
</body>
</html>

```

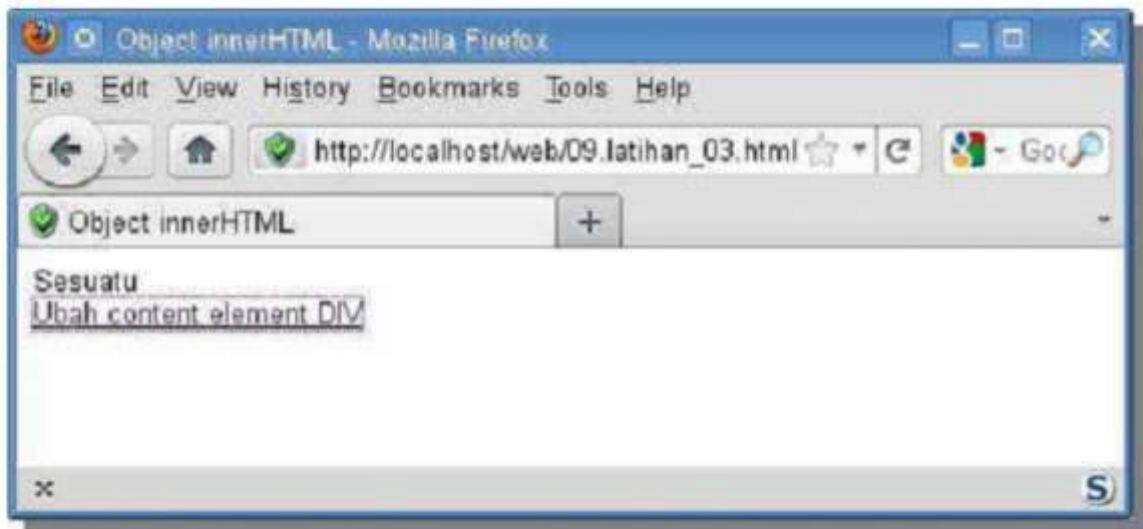
latihan_03.html

```
<html>
  <head>
    <title>Object innerHTML</title>
    <script language="JavaScript">
      function ubahInnerHTML() {
        document.getElementById("divContent").innerHTML = "Sesuai";
      }
    </script>
  </head>
  <body>
    <div id="divContent">
      innerHTML adalah sebuah turunan dari object document.
    </div>
    <a href="#" onClick="ubahInnerHTML();" >Ubah content element DIV</a>
  </body>
</html>
```

Saat user menekan link, event onClick akan menjalankan fungsi `ubahInnerHTML()`. Fungsi ini mempunyai tugas untuk mengubah content element HTML yang mempunyai ID "divContent". Seperti yang ditunjukkan pada baris perintah berikut ini:

```
document.getElementById("divContent").innerHTML = "Sesuai";
```

Sehingga hasilnya menjadi seperti screenshot berikut:




```

<html>
<head>
<title>Contoh implementasi object XMLHttpRequest</title>
<script language="JavaScript">
    function createXMLHttpRequest() {

        var xmlhttp = false;
        try {
            xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
        } catch (e) {
            try {
                xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
            } catch (e) {
                xmlhttp = false;
            }
        }

        if (!xmlhttp && typeof XMLHttpRequest != 'undefined'){
            xmlhttp = new XMLHttpRequest();
        }

        if(!xmlhttp){
            alert("Terjadi kesalahan saat pembuatan XMLHttpRequest object!");
        }

        return xmlhttp;
    }

    function requestContent(){
        reqObj = createXMLHttpRequest();
        reqObj.open("GET", "http://localhost/web/dataLoad.html", true);

        reqObj.onreadystatechange = function(){
            if(reqObj.readyState == 4 && reqObj.status == 200){
                document.getElementById("divContent").innerHTML = reqObj.responseText;
            } else {
                document.getElementById("divContent").innerHTML = "loading..";
            }
        }

        reqObj.send(null);
    }
</script>
<head>
<body>
<div>
    Contoh implementasi object XMLHttpRequest untuk berkomunikasi
    dengan server.
</div>
<a href="#" onClick="requestContent();">Klik disini untuk load data.</a>
<div id="divContent"></div>
</body>
</html>

```

dataLoad.html

```
<html>
  <head>
    <title>Untitled Document</title>
  </head>
  <body>
    <div style= "padding:6px;border:1px solid
      #4da8f7;background:#ffff99;"> <b>File dataLoad.html</b>
    <div>
      Apapun isi dari file ini akan ditampilkan pada element
      HTML yang dituju.
    </div>
    <div>
      Data yang dimuat disini dapat berupa data statis maupun
      data yang sudah diolah terlebih dahulu oleh server.
    </div>
  </div>
</body>
</html>
```

Setelah program ditampilkan pada browser sebagaimana yang ditunjukkan oleh screenshot diatas, cobalah untuk menekan link yang tersedia pada halaman web tersebut. Pada link tersebut terdapat event onClick yang memicu berjalannya fungsi requestContent(). Fungsi ini digunakan untuk mengambil data dari server kemudian menampilkannya pada element yang diinginkan.

Pertama kali, program akan membuat sebuah object XMLHttpRequest melalui statement:

```
reqObj = createXMLHttpRequest();
```

Kemudian program akan memulai melakukan request kepada server secara *asynchronous* menggunakan metode GET dengan alamat server yang dituju adalah " http://localhost/web/dataLoad .html" seperti yang ditunjukkan oleh statement:

```
reqObj.open("GET","http://localhost/web/dataLoad.html",true);
```

Kemudian program akan melakukan pemeriksaan status dari request yang disampaikan kepada server melalui event onreadystatechange. Event ini akan terus memeriksa status komunikasi, sampai data yang diinginkan diterima dengan seutuhnya.

Jika data yang diminta telah selesai ditransfer (ditandai dengan properti readyState = 4 dan status = 200), maka data tersebut akan dituliskan ke element HTML yang dituju, yaitu "divContent".

Sedangkan jika data yang diminta belum terpenuhi, maka "divContent" akan ditulisi "loading..".

Seperti yang ditunjukkan oleh statement berikut:

```
reqObj.onreadystatechange = function(){
  if(reqObj.readyState == 4 && reqObj.status == 200){
    document.getElementById("divContent").innerHTML = reqObj.responseText;
  } else {
    document.getElementById("divContent").innerHTML = "loading..";
  }
}
```

Request akan dikirimkan ke server melalui statement:

```
reqObj.send(null);
```



contactList.xml

```
<xml id="contactList">
  <contact>
    <nama>Rajiva</nama>
    <alamat>Jl. Taman Siswa Yogyakarta</alamat>
    <phone type="mobile">0852 9233 1xxx</phone>
    <phone type="home">0274 212 404</phone>
    <phone type="office">0274 313 501</phone>
    <email>rajiva.root@gmail.com</email>
  </contact>
</xml>
```

```

function createXMLHttpRequest(){
    var xmlhttp = false;
    try {
        xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try {
            xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (e) {
            xmlhttp = false;
        }
    }

    if (!xmlhttp && typeof XMLHttpRequest != 'undefined'){
        xmlhttp = new XMLHttpRequest();
    }

    if(!xmlhttp){
        alert("Terjadi kesalahan saat pembuatan XMLHttpRequest object!");
    }

    return xmlhttp;
}

function requestContent(){
    reqObj = createXMLHttpRequest();
    reqObj.open("GET", "http://localhost/web/contactList.xml", true);

    reqObj.onreadystatechange = function(){
        if(reqObj.readyState == 4 && reqObj.status == 200){
            dataXML = reqObj.responseXML;
            contact = dataXML.getElementsByTagName("contact")[0];

            nama      = contact.getElementsByTagName("name")[0].firstChild.data;
            alamat    = contact.getElementsByTagName("address")[0].firstChild.data;
            phone_1   = contact.getElementsByTagName("phone")[0].firstChild.data;
            phone_2   = contact.getElementsByTagName("phone")[1].firstChild.data;
            phone_3   = contact.getElementsByTagName("phone")[2].firstChild.data;
            email     = contact.getElementsByTagName("email")[0].firstChild.data;
            document.getElementById("txtName").innerHTML = nama;
            document.getElementById("txtAddress").innerHTML = alamat;
            document.getElementById("txtMobilePhone").innerHTML = phone_1;
            document.getElementById("txtHomePhone").innerHTML = phone_2;
            document.getElementById("txtOfficePhone").innerHTML = phone_3;
            document.getElementById("txtEmail").innerHTML = email;

            document.getElementById("divContent").innerHTML = "";
        } else {
            document.getElementById("divContent").innerHTML = "loading..";
        }
    }

    reqObj.send(null);
}

```


latihan_08.html

```
<html>
  <head>
    <title>Contoh implementasi object XMLHttpRequest dan data XML</title>
    <script language="JavaScript" src="latihan_07.js"></script>
  </head>
  <body>
    <div>Contoh implementasi object XMLHttpRequest dengan data XML.</div>
    <a href="#" onClick="requestContent();" >
      Klik disini untuk load data.
    </a>
    <div id="divContent"></div>
    <h4>Detail Kontak</h4>
    <table width="500px" border="1">
      <tr>
        <td width="25%">Nama</td>
        <td><div id="txtName"></div></td>
      </tr>
      <tr>
        <td>Address</td>
        <td><div id="txtAddress"></div></td>
      </tr>
      <tr>
        <td>Mobile Phone</td>
        <td><div id="txtMobilePhone"></div></td>
      </tr>
      <tr>
        <td>Home Phone</td>
        <td><div id="txtHomePhone"></div></td>
      </tr>
      <tr>
        <td>Office Phone</td>
        <td><div id="txtOfficePhone"></div></td>
      </tr>
      <tr>
        <td>Email</td>
        <td><div id="txtEmail"></div></td>
      </tr>
    </table>
  </body>
</html>
```

Contoh latihan tersebut diatas digunakan untuk mengambil data dalam bentuk XML yang berupa detail kontak. Kemudian program mengolah data tersebut menjadi bagian-bagian yang diinginkan. Cara kerja dari program dapat dijelaskan sebagai berikut:

Ketika link pada program ditekan, event onClick akan menjalankan fungsi requestContent(). Fungsi tersebut kemudian membuat sebuah object XMLHttpRequest dan membuat koneksi secara *asynchronous* dengan metode GET seperti yang ditunjukkan oleh statement berikut:

```
reqObj = createXMLHttpRequest();
reqObj.open("GET", "http://localhost/web/contactList.xml", true);
```

Pada statement diatas, dibelakang fungsi getElementByTagName ditambahi string [0] pada bagian belakang statement. Ini karena JavaScript mengolah data hasil pengolahan fungsi getElementByTagName sebagai sebuah array.

Dengan ditambahkannya string [0] pada akhir statement, maka data kontak yang diambil adalah data yang pertama (element pertama dari array dimulai dari index 0). Setelah statement tersebut dijalankan, variabel contact akan berisi semua data yang terletak diantara tag <contact> ... </contact>.

Proses selanjutnya adalah mengolah data yang terkandung pada variabel contact menjadi bagianbagian yang kita inginkan. Proses pembacaanya datanya masih menggunakan fungsi

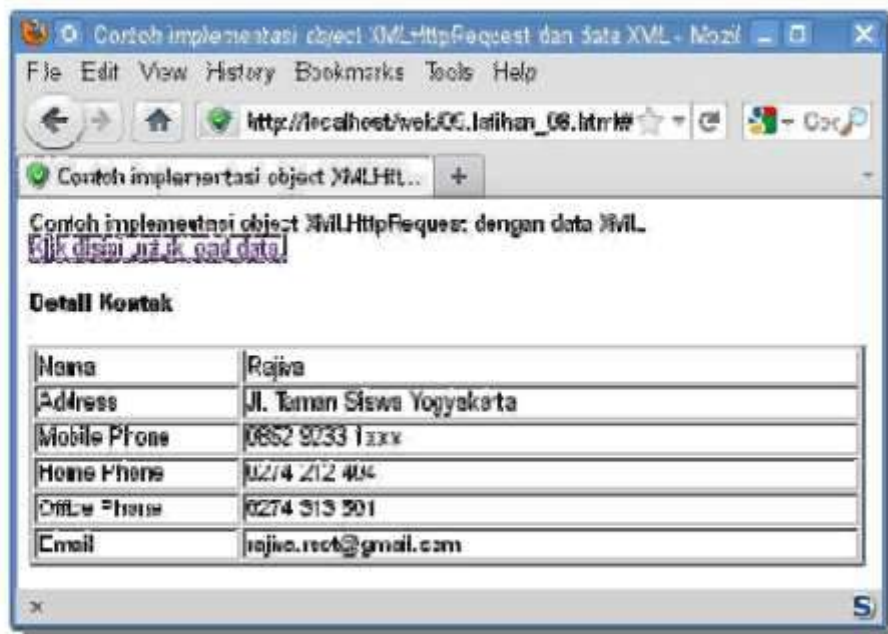
getElementsByTagName, namun menggunakan object firstChild.data untuk membaca data text (content) yang terkandung pada masing-masing tag. Berikut ini adalah statement yang digunakan:

```
nama      = contact.getElementsByTagName("name")[0].firstChild.data;
alamat     = contact.getElementsByTagName("address")[0].firstChild.data;
phone_1    = contact.getElementsByTagName("phone")[0].firstChild.data;
phone_2    = contact.getElementsByTagName("phone")[1].firstChild.data;
phone_3    = contact.getElementsByTagName("phone")[2].firstChild.data;
email      = contact.getElementsByTagName("email")[0].firstChild.data;
```

Proses pengolahan data seperti yang ditunjukkan oleh potongan program diatas hampir sama untuk data satu item dengan yang lainnya. Yang menjadi perhatian disini adalah pengolahan data untuk tag phone. Oleh karena terdapat tiga tag phone didalam satu data kontak, maka hasil pengolahan data oleh fungsi getElementsByTagName juga akan menghasilkan tiga item yang ditampung dalam satu array. Oleh karena itu, masing-masing element pada array tersebut dapat dipanggil melalui index 0, 1 atau 2.

Proses selanjutnya adalah menempatkan data-data hasil pengolahan tadi kedalam element HTML yang diinginkan melalui properti innerHTML. Perhatikan potongan program berikut ini:

```
document.getElementById("txtName").innerHTML = nama;
document.getElementById("txtAddress").innerHTML = alamat;
document.getElementById("txtMobilePhone").innerHTML = phone_1;
document.getElementById("txtHomePhone").innerHTML = phone_2;
document.getElementById("txtOfficePhone").innerHTML = phone_3;
document.getElementById("txtEmail").innerHTML = email;
```



MODUL 7

Pengenalan PHP

7.1 Tujuan

- c. Mengenalkan kepada mahasiswa tentang pemrograman PHP
- d. Mahasiswa dapat membuat script PHP secara sederhana
- e. Mahasiswa dapat membuat halaman web dengan menggunakan script HTML dan PHP

7.2 Dasar Teori

PHP (Preprocessor Hypertext) adalah bahasa scripting yang menyatu dengan HTML dan dijalankan pada server side. Artinya semua sintaks yang diberikan akan sepenuhnya dijalankan pada server sedangkan yang dikirimkan ke browser hanya hasilnya saja berupa HTML. Untuk membedakan perintah HTML dan PHP digunakan tanda `<?...?>` atau `<?php ... ?>`

PHP dapat diaplikasikan dengan berbagai macam database, seperti MySQL, PostgreSQL, Oracle, dan lainnya. Beberapa script dasar PHP

g. Menampilkan text

```
echo "...text..... <br>"  
<br> : ganti baris
```

h. Variabel

Untuk membuat variabel diberi tanda dollar (\$). Variabel berfungsi untuk menyimpan suatu nilai dan dapat berubah-ubah. Penulisan variable yang benar adalah :

- Karakter pertama tidak boleh berupa angka (harus berupa huruf atau garis bawah)
- Tidak mengandung spasi
- Pemakaian huruf kapital dan huruf kecil dibedakan

Contoh penulisan variable :

```
$data, $data1, $data_ku
```

χ. Array

Array merupakan suatu variabel yang dapat berisi banyak data dalam waktu yang sama.

Pendefinisian Array dapat dibentuk dengan format berikut :

```
$nama_array = array(elemen_1, ..., elemen_n);
```

Untuk menghitung jumlah elemen array digunakan fungsi `count()`, dengan format `count($nama_array)`

ε. Operator

Dalam PHP juga dapat melakukan proses operasi, baik itu penjumlahan, operasi logika, ataupun operasi perbandingan. Operator Matematika yang digunakan dalam PHP yaitu :

Operator	Fungsi	Operator	Fungsi
+	Penjumlahan	-	Pengurangan
*	Perkalian	/	Pembagian
%	Sisa pembagian	++, --	Penaikan, penurunan

3. Operator Perbandingan yang digunakan dalam PHP yaitu :

Operator	Fungsi	Operator	Fungsi
==	Sama dengan	<	Kurang dari
>	Lebih dari	<=	Kurang dari atau sama dengan
>=	Lebih dari atau sama dengan	!= , <>	Tidak sama dengan

Selain itu, operator Logika juga dapat digunakan di PHP, antara lain, and (&&), or (||), xor, dan !.

4. Penulisan Karakter Khusus dengan tanda \

Karakter yang ditulis dengan diawali tanda (\) yaitu:

Karakter	Keterangan	Karakter	Keterangan
\"	Tanda petik ganda	\\	Tanda backslash
\\$	Tanda \$	\n	Pindah baris
\t	tab	\x00 s.d \xFF	heksadesimal

• Tipe Data

Tipe data yang dikenal pada pemrograman PHP yaitu :

Tipe data	Keterangan
Integer	Tipe data bilangan bulat
Double	Tipe data bilangan real
String	Tipe data teks

• Konversi Tipe Data

Fungsi-fungsi yang digunakan dalam PHP untuk mengkonversikan tipe data ke tipe data yang lain yaitu dengan memberikan fungsi intval, doubleval, dan strval. Atau dengan menggunakan teknik cast, yaitu dengan mengubah tipe ekspresi yang akan dikonversi.

3. Menampilkan Tanggal dan Waktu

Untuk menampilkan tanggal dan waktu secara update, dapat menggunakan perintah date dengan format-format berikut :

Format	Keterangan
a,A	am atau pm, AM atau PM
d,D	hari/tanggal dalam 2 digit, hari (Sun..Sat)
F	nama bulan (January..December)
g,G	jam (1..12), jam (0..23)
Z	hari dari tahun (0..365)
y,Y	tahun dalam 2 digit, tahun dalam 4 digit
h,H	jam (01..12), (00..23)
l	menit (00..59)
m,M	Nama bulan (01..12), nama bulan (Jan..Dec)
S	Detik (00..59)
W	hari (0=Sunday..6=Saturday)

7.3 Pelaksanaan Praktikum

1. Ketikkan program berikut ini :

```

1  <html>
2  <head>
3  <title> Variabel </title>
4  </head>
5  <body>
6  <?php
7  $nilai_1 = 10;
8  $nilai_2 = 3;
9  $nilai_3 = 2 * $nilai_1 + 8 * $nilai_2;
10 echo "nilai = ", $nilai_3;
11 echo "<br>";
12 $jumlah = $nilai_1 + $nilai_2;
13 echo "hasil dari $nilai_1 + $nilai_2 adalah : $jumlah"; echo "<br><br>";
14 echo "\"Nama : Agung Teguh Wibowo Almais\" <br>";
15 echo "NIM : 146060300111010";
16 ?>
17 </body>
18 </html>

```

Nama Agung Teguh Wibowo Almais dan NIM : 146060300111010 diatas ganti dengan nama masing-masing).

3. Simpan dengan menggunakan ekstensi php (misalnya mod_1.php), dan jalankan di web browser.

3. Ketikkan program berikut :

```
1 <html>
2 <head>
3 <title> Pemrograman PHP dengan Array</title>
4 </head>
5 <body>
6 <?php
7 //penulisan array dapat dibuat seperti berikut
8 $nama[] = "Agung Teguh";
9 $nama[] = "Wibowo";
10 $nama[] = "Almais";
11 echo $nama[1] . $nama[2] . $nama[0];
12 echo "<br>";
13 ?>
14 </body>
15 </html>
```

4. Simpan script diatas lalu lihat hasilnya di web browser. Perhatikan urutan nama-nama yang tampil dari perintah echo \$nama[1] . \$nama[2] . \$nama[0];
5. Tambahkan fungsi count untuk menghitung jumlah elemen array \$nama dengan menambahkan script berikut :

```
//menghitung jumlah elemen array
$jumlah_array = count($nama);
echo "jumlah elemen array = ". $jumlah_array;
```

6. Ketikkan script berikut ini untuk mengkonversi tipe data.

```
1 <html>
2 <head>
3 <title>Konversi Tipe</title>
4 </head>
5 <body>
6 <?php
7 $a = 300.4;
8 echo $a;
9 echo "<br>";
10 echo "tipe Double : ", doubleval($a), "<br>";
11 echo "tipe Integer : ", intval($a), "<br>";
12 echo "tipe string : ", strval($a);
13 ?>
14 </body>
15 </html>
```

3. Ketikkan script berikut untuk menampilkan tanggal dengan mengambil tanggal di server.

```
1 <html>
2 <head>
3 <title>Tanggal</title>
4 </head>
5 <body>
6 <?php
7 echo date("m-F-Y, g:i:s a"); ?>
8 </body>
9 </html>
```

7.4 Latihan

- Buatlah aplikasi sederhana untuk menghitung luas persegi panjang

7.5 Tugas

- Modifikasilah program pada praktik 7 untuk menampilkan informasi tanggal dan waktu dengan mencoba menggunakan format-format date lainnya.
- Buatlah program sederhana tentang biodata tanpa database.

MODUL 8

MATERI PHP LANJUT PART 2

Pada minggu lalu, kita telah mengenal perintah-perintah dasar PHP, fungsi- fungsi PHP, dsb. Praktikum kali ini akan sedikit mengulang materi minggu lalu. Pertemuan kali ini kita akan sedikit mengenal variabel yang dideklarasikan melalui form melalui dokumen HTML. Variabel tersebut diantaranya adalah `$_POST` ["variable"], `$_GET` ["variable"], dan `$_REQUEST` ["variable"].

1. `$_POST` ["variable"]

Berfungsi untuk mengambil nilai variabel karena pengaruh method = post pada suatu form. Contoh :

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title></title>
5 </head>
6 <body>
7 <form method="POST" action="postAct.php">
8 <table width="400" align="center" cellpadding="2" cellspacing="2">
9 <tr>
10 <td width="130">Nama</td>
11 <td><input type="text" name="nama"></td>
12 </tr>
13 <tr>
14 <td width="130">Email</td>
15 <td><input type="text" name="email"></td>
16 </tr>
17 <tr>
18 <td>
19 <input type="submit" name="btnLogin" value="Login">
20 <input type="reset" name="reset" value="Reset">
21 </td>
22 </tr>
23 </table>
24 </form>
25 </body>
26 </html>
```

Buatlah file post.php seperti di bawah ini sebagai form input. Kemudian hasil dari input akan dikirim ke

file php lain dengan nama postAct.php seperti file di bawah ini :

```
1 <?php
2 echo "<center>Nama :".$_POST['nama']."</center><br>";
3 echo "<center>Email :".$_POST['email']."</center><br>";
4 ?>
```

2. `$_GET` ["variable"]

Berfungsi untuk mengambil nilai variabel karena pengaruh method = get pada suatu form. Untuk penggunaan `$_POST` dan `$_GET` hampir sama, silahkan ubah file dengan menggunakan `$_POST` di atas dengan menggunakan `$_GET`.

3. `$_REQUEST` ["variable"]

Berfungsi untuk mengambil nilai variabel karena pengaruh dari hasil link.

Contoh : `<a href="http://localhost/cek.php?nama=rahmi">Link ke Cek</a>`

Include dan Require

Fungsi include dan require berfungsi untuk mengeksekusi suatu file dengan file lainnya. Dengan kedua fungsi ini, dapat dibuat fungsi- fungsi, konstanta, atau perintah operasi biasa pada file terpisah yang dapat dipanggil dengan menggunakan file lain.

Bentuk umum penulisannya :

Include "nama file"

Require "nama file"

Contoh :

Buatlah dua buah file dimana file pertama merupakan file sumber sedangkan file kedua merupakan file pengesekusi file sumber.

File sumber bernama inc.php

```
1 <?php
2 $angka=90;
3 ?>
```

File pengesekusi bernama proses.php

```
1 <?php
2 include "inc.php";
3 echo $angka;
4 echo "<br>";
5 if ($angka==100){
6     echo "Memuaskan";
7 } elseif ($angka<100&&$angka>=85) {
8     echo "Sangat Baik";
9 } elseif ($angka<85&&$angka>=70) {
10    echo "Baik";
11 } elseif ($angka<70&&$angka>=55) {
12    echo "Cukup";
13 } elseif ($angka<55&&$angka>=0) {
14    echo "Kurang";
15 }
16 ?>
```

Cookies

Cookies merupakan proses penyimpanan nilai sementara dari suatu variable pada client browser. Fungsi yang dibutuhkan untuk menerapkan fungsi cookies adalah setcookie(). Cookies merupakan bagian dari HTTP sehingga diperlukan pendeklarasian cookies agar dapat mengirimkan output ke client browser.

Bentuk umum penulisan :

Setcookie(variable cookies, nilai cookies) atau

Setcookie(variable cookies, nilai cookies, lama waktu)

Contoh penerapan cookies :

Buatlah dua buah file yaitu file linkcookies.php dan cekcookies.php.

File linkcookies.php :

```
1 <?php
2 setcookie("variable_cookies","Ini adalah variable cookies",time()+60);
3 echo "<a href=cekcookies.php>Cek Cookies</a>";
4 ?>
```

File cekcookies.php

```
1 <?php
2 if (isset($variable_cookies)) {
3     echo "Variable cookiesnya \"$variable_cookies\" nilainya adalah" . $
    variable_cookies;
4 } else {
5     echo "Variable cookies belum diterapkan";
6 }
7 ?>
```

Redirect Halaman

Redirect halaman merupakan bentuk pengalihan halaman ke halaman lain. Fungsi redirect halaman sangat diperlukan apabila pada suatu situs sedang mengalami perbaikan sehingga homepage kita dapat dialihkan ke halaman lain. Misalnya file index.php kita sedang diperbaiki, maka ketika seseorang melakukan browsing ke halaman index.php akan dialihkan ke halaman lain misalnya repair.php. Redirect halaman dapat juga diterapkan dengan menggunakan tag html.

```

1 <html>
2 <head>
3 <meta http-equiv="refresh" content="2; url=http://localhost/modul8/test.html">
4 </head>
5 <body>
6 Tes Redirect
7 </body>
8 </html>

```

Penjelasan ;

Ketika anda melakukan browsing terhadap halaman redirect.html, maka secara otomatis akan dialihkan ke halaman tes.php dalam waktu 2 detik. Untuk pengaturan redirect dengan php script menggunakan aturan sbb :

Bentuk umum penulisan : `header("Location:halaman tujuan");`

Contoh :

Silahkan buat tiga buah file, file pertama merupakan form yang menerima inputan silahkan beri nama input.php.

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title></title>
5 </head>
6 <body>
7 <form method="POST" action="hasilkirim.php">
8 <table width="400" align="center" cellpadding="2" cellspacing="2">
9 <tr>
10 <td width="130">Nama</td>
11 <td><input type="text" name="nama"></td>
12 </tr>
13 <tr>
14 <td colspan="2" align="center">
15 <input type="submit" name="btnLogin" value="Kirim">
16 <input type="reset" name="reset" value="Reset">
17 </td>
18 </tr>
19 </table>
20 </form>
21 </body>
22 </html>

```

Kemudian file kedua untuk melakukan pengecekan beri nama hasilkirim.php.

```

1 <?php
2 if (empty($_POST['nama'])) {
3     header("Location:kosong.php");
4 } else {
5     echo "<center>Nama :".$_POST['nama']."</center><br>";
6 }
7 ?>

```

File terakhir adalah file redirect silahkan beri nama kosong.php.

```

1 <?php
2 echo "Data kosong";
3 ?>

```

Penjelasan :

Ketika kita memasukkan data, maka akan tampil nama hasil inputan pada browser. Akan tetapi ketika data kosong maka akan mer-redirect ke halaman lain yaitu kosong.php.

TUGAS :

1. Buatlah form login dengan melakukan include pada halaman web anda yang akan menampilkan nama anda, email, jam, hari, dan tanggal login setelah login tanpa menggunakan database dengan menggunakan method get atau post!
2. Dari soal nomor satu, buatlah halaman lain untuk redirect yang menunjukkan bahwa data tidak lengkap ketika user tidak memasukkan nama atau email pada saat login!

MODUL 9

KONEKSI PHP DENGAN MySQL

A. Tujuan :

1. Memahami langkah-langkah koneksi PHP dengan MySQL.
2. Memahami perbedaan penggunaan mysql dengan mysqli.

B. Dasar Teori

• Mengapa menggunakan Mysqli Extension ??

Pada PHP versi 5.5, extension mysql tidak direkomendasikan lagi karena berstatus *deprecated*, yang kemungkinan besar akan dihapus pada versi PHP berikutnya.

• Apa perbedaan mysql extension dan mysqli extension ??

Secara umum tidak ada perbedaan yang signifikan. Sebagian besar fungsi – fungsi sangat mirip dengan mysql extension.

Sebagai contoh dalam membuat sebuah koneksi dengan MySQL Server, di dalam mysql kita menggunakan fungsi `mysql_connect()`; sedangkan dalam mysqli berubah menjadi `mysqli_connect()`; Kita tinggal menambahkan 'i' saja dalam penulisan fungsinya. Ada satu lagi perbedaan dalam menuliskan sebuah argument di dalam sebuah fungsi. Dalam mysql argument resources diletakkan di akhir, sedangkan mysqli diletakkan di awal.

Contoh :

```
Mysql
mysql_query("SELECT * FROM tb_anggota", $link);
Mysqli
mysqli_query($link, "SELECT * FROM tb_anggota");
```

Langkah-langkah koneksi PHP- MySQL

1. Membuka koneksi ke server MySQL

`mysql_connect()`

Digunakan untuk melakukan uji dan koneksi kepada server database MySQL.

Sintaks :

```
$conn = mysql_connect ("host", "username", "password");
```

\$conn adalah nama variabel penampung status hasil koneksi kepada database.

host adalah nama host atau alamat server database MySQL.

username adalah nama user yang telah diberi hak untuk dapat mengakses server database.

password adalah kata sandi untuk username untuk dapat masuk ke dalam database

2. Memilih database yang akan digunakan di server

`mysql_select_db()`

Digunakan untuk melakukan koneksi kepada database yang dalam server yang berhasil dikoneksi dengan perintah `mysql_connect()`.

Sintaks :

```
$pilih = mysql_select_db("namadatabase", $conn);
```

\$pilih berisi status koneksi kepada database.

\$conn merupakan koneksi kepada server database yang berhasil.

namadatabase adalah nama database yang akan dikenai proses.

3. Mengambil sebuah query dari sebuah database.

mysql_query()

Digunakan untuk melakukan eksekusi perintah SQL untuk memanipulasi database yang berhasil dilakukan koneksinya menggunakan mysql_select_db().

Sintaks :

\$hasil = mysql_query("SQL Statement");

\$hasil akan berupa record set apabila SQL Statement berupa perintah select.

4. Mengambil record dari tabel

a. mysql_fetch_array()

Digunakan untuk melakukan pemrosesan hasil query yang dilakukan dengan perintah mysql_query(), dan memasukkannya ke dalam array asosiatif, array numeris atau keduanya.

Sintaks :

\$row = mysql_fetch_array(\$hasil);

\$row adalah array satu record dari record \$hasil yang diproses nomor record sesuai dengan nomor urut dari proses mysql_fetch_array yang sedang dilakukan. \$hasil adalah record set yang akan diproses.

b. mysql_fetch_assoc()

Fungsi ini hampir sama dengan fungsi mysql_fetch_array(), hanya saja array yang dihasilkan hanya array asosiatif.

Sintaks :

\$row = mysql_fetch_assoc(\$hasil);

c. mysql_fetch_row()

Fungsi ini hampir sama dengan fungsi mysql_fetch_array(), hanya saja array yang dihasilkan hanya array numeris.

Sintaks :

\$row = mysql_fetch_row(\$hasil);

d. mysql_num_rows()

Fungsi ini digunakan untuk menghitung jumlah record yang ada pada database.

Sintaks :

\$jml = mysql_num_rows(\$hasil);

\$jml akan memiliki nilai sesuai dengan jumlah record yang ada.

C. PERCOBAAN

1. Menguji interkoneksi PHP dengan MySQL.

```
1 <html>
2 <head>
3   <title>Koneksi Database MySQL</title>
4 </head>
5 <body>
6 <h1>Demo koneksi database MySQL</h1>
7 <?
8 $conn=mysql_connect ("localhost","root","");
9 if ($conn) {
10   echo "OK";
11 } else {
12   echo "Server not connected";
13 }
14 ?>
15 </body>
16 </html>
```

2. Melihat perbedaan antara `mysql_fetch_array()`, `mysql_fetch_assoc()`, `mysql_fetch_row()`.

a. Buatlah tabel liga berikut ini, dengan 3 field : kode, negara, champion.

Create table liga (
kode char(3) not null,
negara char(15),
champion int
);

b. Isilah tabel dengan data berikut ini :

kode	negara	Champion
Jer	Jerman	4
Spa	Spanyol	3
Eng	English	3

c. Akses databases menggunakan `mysql_fetch_array()`

```
1 <HTML>
2 <HEAD>
3 <title>Koneksi Database MySQL</title>
4 </HEAD>
5 <BODY>
6 <h1>Koneksi database dengan mysql_fetch_array</h1> <?
7 $conn=mysql_connect ("localhost","root","")
8 or die ("koneksi gagal");
9 mysql_select_db("faruq",$conn);
10 $hasil = mysql_query("select * from liga",$conn); while ($row=
    mysql_fetch_array($hasil)) {
11
12   echo "Liga " . $row["negara"];
13 }
14 ?>
15 </BODY>
16 </HTML>
```

d. Akses databases menggunakan `mysql_fetch_row()`

```

1 <HTML>
2 <HEAD>
3 <title>Koneksi Database MySQL</title>
4 </HEAD>
5 <BODY>
6 <h1>Koneksi database dengan mysql_fetch_assoc</h1> <?
7 $conn=mysql_connect ("localhost","root","") or die ("koneksi gagal");
8 mysql_select_db("faruq",$conn);
9 $hasil = mysql_query("select * from liga",$conn); while ($row=
    mysql_fetch_row($hasil)) {
10 echo "Liga " . $row[1];
11 echo " mempunyai " . $row[2];
12 echo " wakil di liga champion <br>";
13 }
14 ?>
15 </BODY>
16 </HTML>

```

D. TUGAS

1. Buatlah buku tamu dengan menggunakan *mysql extension*. Dengan nama tabel **buku_tamu** dan tabel yang kalian butuhkan terdapat 3 field seperti dibawah ini:

#	Nama	Tipe data	Panjang/Batas
1	ID_BT	INT	10
2	NAMA	VARCHAR	200
3	EMAIL	VARCHAR	50
4	ISI	TEXT	

2. Pada pembuatan buku tamu Tugas No.1 gunakan fungsi *include* untuk menggabungkan file koneksi database dengan file yang lainnya.

MODUL 10

Bahasa PHP dan MySQL

Tujuan Pembelajaran

Mempraktekkan pembuatan program-program dasar pengolahan data dengan menggunakan bahasa PHP

Pokok Bahasan

- Program Menambah Data
- Program Mencetak Data
- Program Login
- Program-program Administrator

Indikator Keberhasilan

Mahasiswa memahami langkah-langkah pengolahan data menggunakan PHP dan dapat menerapkannya untuk membuat program input dan cetak data

Penjelasan

Php merupakan bahasa script yang dijalankan pada sisi server (SSS : Server Side Scripting). Database yang didukung PHP antara lain : MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC.

PHP adalah software Open Source, bebas untuk diunduh dan digunakan.

File PHP dapat berisi teks, tag HTML, dan Script. File PHP dikembalikan ke browser dalam bentuk plain HTML. File PHP dapat berekstensi .php, .php3, atau .phtml.

Kelebihan PHP sebagai bahasa script adalah dapat dijalankan pada berbagai platform (windows, linux, dan sebagainya), kompatibel terhadap hampir semua server yang digunakan saat ini, bebas

diunduh dari situs resmi PHP www.php.net, mudah dipelajari dan berjalan dengan efisien pada sisi server.

Untuk menggunakan PHP harus menginstall terlebih dahulu web server Apache (atau IIS) pada komputer/server yang akan digunakan, serta menginstall PHP dan MySQL. Atau dapat menggunakan layanan hosting yang menyediakan dukungan terhadap PHP dan MySQL.

Proses mengolah data menggunakan PHP pada dasarnya menggunakan tiga langkah berikut:

1. Membuat sambungan dari PHP ke server database (MySQL)

Langkah pertama memerlukan fungsi *mysql_connect()* dengan tiga parameternya yaitu: nama host tempat server database mysql berada, nama user yang terdaftar pada server database MySQL, dan password dari nama user. Sintaksnya:

```
mysql_connect("nama_host","nama_username", "password_username");
```

Contoh:

```
mysql_connect("192.168.10.250","a121204567","");
```

2. Menentukan dan membuka database yang akan digunakan

Langkah kedua menggunakan fungsi *mysql_select_db()* dengan parameter tunggal yaitu nama database yang akan digunakan. Sintaks:

```
mysql_select_db("nama_db");
```

Contoh:

```
mysql_select_db("a121204567");
```

3. Menjalankan query

Langkah ketiga menggunakan fungsi *mysql_query()* dengan parameter berupa perintah sql yang akan dijalankan. Sintaksnya:

```
mysql_query("perintah_sql");
```

Contoh:

```
mysql_query("insert mahasiswa set nim='$nim', nama='$nama', ipk='$ipk'");
```

Jika untuk membuat program cetak, maka ditambah satu langkah lagi yaitu menguraikan data yang akan dicetak dengan menggunakan fungsi *mysql_fetch_row()* dengan parameter tunggal yaitu hasil query.

Contoh:

```

$sql="select * from mahasiswa";
$qry=mysql_query($sql)
While($hasil=mysql_fetch_row($qry)){
    echo "Nim :".$hasil[0]."<br />
        Nama: ".$hasil[1]."<br />
        IPK:".$hasil[2]."<br />";
}

```

Praktikum

Pada praktikum kali ini Anda akan dituntun untuk membuat program-program dasar menggunakan PHP. Salah satu program dasar tersebut adalah program input data. Untuk membuat program input data ini, silahkan gunakan form yang telah dibuat pada praktikum sebelumnya (kontak.html) yang berisi inputan data kontak untuk pengguna, yaitu NAMA, JENIS KELAMIN, EMAIL, ALAMAT, KOTA, PESAN. Form tersebut belum dilengkapi dengan file pengolah datanya. Oleh karena itu pada praktikum ini, form tersebut akan dilengkapi dengan program pengolah data. Selanjutnya ikuti instruksi-instruksi berikut ini:

1. Buat tabel dengan nama kontak dalam database Anda (database Anda biasanya sudah dibuatkan dengan nama sesuai username, a12xxxxxx. Jika belum, buat database dengan nama username Anda). Field-fieldnya adalah:

Field	Tipe Data	Panjang	Keterangan
id	integer	4	Not null, auto_increment
Nama	Varchar	30	Not null
jkel	Varchar	10	-
Email	Varchar	40	-
Alamat	Varchar	50	-
Kota	Varchar	20	-
Pesan	text	-	-

2. Tambahkah pada atribut action form nama file pengolah datanya yaitu **simpan_kontak.php**. File ini yang akan dibuat pada praktikum kali ini.
3. Buat file **simpan_kontak.php** dan simpan pada direktori yang sama dengan direktori dimana file **kontak.html** tersimpan.
4. Tulis kode berikut pada file **simpan_kontak.php**:

```

<?php
$host="nama_host";
$username="nama_username";

```

```

$password="password_username";
mysql_connect($host, $username, $password) or die("Koneksi gagal dibangun");
mysql_select_db("situspersonal") or die("Database tidak dapat dibuka");
//memindahkan nilai data form ke variabel sederhana agar mudah ditulis
$vnama=$_POST['nama'];
$vjkel=$_POST['jkel'];
$vemail=$_POST['email'];
$valamat=$_POST['alamat'];
$vkota=$_POST['kota'];
$vpesan=$_POST['pesan'];

$sql="insert kontak set nama='$vnama',
                                jkel='$vjkel',
                                email='$vemail',
                                alamat='$valamat',
                                kota='$vkota',
                                pesan='$vpesan'";
mysql_query($sql) or die("Proses simpan ke database gagal");
mysql_close();
header("location:kontak.html");
?>

```

Sebelumnya, ganti nilai *nama_host* dengan *dinus.net* atau *192.168.10.250*, *nama_username* dengan nama user Anda (*al2xxxxxxx*), dan *password_username* untuk username Anda biasanya dikosongkan. Pada program diatas, perintah sql disimpan dulu dalam sebuah variabel, kemudian variabel ini yang digunakan pada fungsi *mysql_query()*. Perintah terakhir, yaitu *header("location:kontak.html")*, digunakan untuk mengarahkan browser agar menampilkan form setelah proses penyimpanan ke database selesai

5. Jalankan form kontak lalu isi dengan data kemudian tekan tombol submitnya. Jika program sudah benar maka sebuah rekod akan ditambahkan kedalam database.
6. Setelah program input selesai, berikutnya adalah membuat program cetak untuk menampilkan data yang sudah tersimpan dalam database. Buat sebuah file php baru beri nama **cetak.php**. Simpan file ini dalam direktori baru yang perlu Anda buat terlebih dahulu yaitu **admin**. Program cetak disimpan dalam direktori admin karena program ini tergolong program administrator situs yang hanya boleh dibuka oleh pemilik situs.
7. Sebelum melanjutkan membuat program cetak, buat terlebih dahulu file php dan beri nama **koneksi.inc.php** serta simpan dalam direktori admin, yang berisi kode:

```

$host="isi_dengan_nama_host";
$username="isi_dengan_username";
$password="";
mysql_connect($host, $username, $password) or die("Koneksi gagal dibangun");
mysql_select_db("situspersonal") or die("Database tidak dapat dibuka");

```

Perintah-perintah yang ada didalam file koneksi.inc.php ini adalah perintah yang selalu digunakan setiap kali kita membuat program php. Oleh karena itu, biasanya perintah-perintah ini disimpan dalam file terpisah lalu dipanggil dari file php yang membutuhkannya.

Hapus perintah-perintah yang sama dalam file simpan_kontak.php dan ganti dengan perintah:

```
include "admin/koneksi.inc.php";
```

Perintah ini akan memuat isi file koneksi.inc.php kedalam file simpan_kontak.php sehingga dapat digunakan olehnya.

8. Pada file cetak.php tulis kode berikut:

```
<?php
include "koneksi.inc.php";
$sql="select * from kontak order by nama;";
$qry=mysql_query($sql) or die("Proses cetak gagal");
echo "<table width='75%' cellpadding='2' cellspacing='0' border='1'>
    <tr>
        <th>No</th>
        <th>Nama</th>
        <th>Email</th>
        <th>Alamat</th>
        <th>Kota</th>
        <th>Pesan</th>
        <th></th>";
$no=1;
while($hasil=mysql_fetch_row($qry)){
    echo "<tr>
        <td>$no</td>
        <td>$hasil[0]</td>
        <td>$hasil[1]</td>
        <td>$hasil[2]</td>
        <td>$hasil[3]</td>
        <td>$hasil[4]</td>
        <td>$hasil[5]</td>
        </tr>";
    $no++;
}
echo "</table>";
?>
<a href="index.html">Kembali</a>
```

Tugas

Cari di Internet program login. Setelah diperoleh ganti namanya dengan index.php dan simpan dalam direktori admin. Buatlah tabel mysql yang diperlukan oleh program login ini dalam database Anda. Modifikasi kodenya agar Jika program login ini dijalankan kemudian diisi dengan username

dan password yang benar, maka akan menampilkan program cetak.php milik Anda. Jadi program cetak.php milik Anda hanya bisa dibuka melalui program login.