

PROGRAM KOMPUTER 2

Desain Pemrograman WEB dengan Bahasa HTML,
PHP, dan Database MySQL



TADRIS MATEMATIKA
FAKULTAS TARBIYAH DAN ILMU KEGURUAN
IAIN PEKALONGAN
2019

DAFTAR ISI

Keterangan	Halaman
HALAMAN JUDUL.....	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR	iv
DAFTAR TABEL.....	vi
DESKRIPSI UMUM.....	1
BAB I PENDAHULUAN.....	3
1.1. Pengenalan Internet	3
1.2. Format Dokumen Di Internet.....	3
1.3. Pemrograman Internet.....	4
1.4. Pemrograman PHP.....	4
1.5. Mengapa Memilih PHP	5
BAB II INSTALASI WEB SERVER	7
2.1. Instalasi Aplikasi Web Server	7
2.2. Macromedia Dreamweaver	11
2.3. Aplikasi Browsing.....	16
BAB III DASAR-DASAR HTML	18
3.1. Mengenal HTML.....	18
3.2. Head, Body Dan Title	20
3.3. Paragraf	21
3.4. Pengaturan Huruf (Font)	23
3.5. Warna Dokumen.....	24
3.6. Link	25
3.7. Menggunakan Tabel	26
3.8. Membuat Form.....	27
BAB IV VARIABEL, KONSTANTA DAN OPERATOR.....	31
4.1. Memulai PHP	31
4.2. Menulis PHP.....	31

4.3. Komentar	32
4.4. Meng-escape Karakter	32
4.5. Variabel Dan Konstanta	33
4.6. Operator	40
BAB V STRUKTUR KONTROL.....	43
5.1. Kontrol Keputusan	43
5.2. Kontrol Pengulangan.....	47
5.3. Kontrol Perpindahan.....	49
BAB VI ARRAY.....	50
6.1. Menciptakan Array	51
6.2. Menampilkan Isi Array.....	51
6.3. Array Asosiatif.....	52
6.4. Array Multimensi	53
6.4. Mengurutkan Array	54
BAB VII STRING DAN FUNGSI.....	58
BAB VIII DATABASE MYSQL.....	63
8.1. Membuat Database	65
8.2. Membuat Tabel	66
8.3. Perintah Dasar SQL.....	68
8.4. Membangun Suatu Aplikasi PHP Dan MySQL.....	69
DAFTAR PUSTAKA	75

DAFTAR GAMBAR

Keterangan	Halaman
2.1. Instalasi Aplikasi <i>Web Server</i> Langkah 1	8
2.2. Instalasi Aplikasi <i>Web Server</i> Langkah 2	8
2.3. Instalasi Aplikasi <i>Web Server</i> Langkah 3	9
2.4. Instalasi Aplikasi <i>Web Server</i> Langkah 4	9
2.5. Instalasi Aplikasi <i>Web Server</i> Langkah 5	10
2.6. Instalasi Aplikasi <i>Web Server</i> Langkah 6	10
2.7. Instalasi Aplikasi <i>Web Server</i> Langkah 7	11
2.8. File Master Dreamweaver CS6.....	11
2.9. <i>Install</i> Aplikasi Macromedia Dreamweaver CS6 Tahap 1	12
2.10. <i>Install</i> Aplikasi Macromedia Dreamweaver CS6 Tahap 2	13
2.11. <i>Install</i> Aplikasi Macromedia Dreamweaver CS6 Tahap 3	14
2.12. <i>File</i> Aktivator Macromedia Dreamweaver CS6	14
2.13. Tampilan Awal Macromedia Dreamweaver CS6.....	15
2.14. Tampilan <i>Script</i> Pada Macromedia Dreamweaver CS6	16
2.15. <i>Browsing</i> Internet.....	16
2.17. <i>Web Server</i> Apache	17
3.1. Hallo.html.....	19
3.2. Tampilan Dokumen HTML <i>File</i> Hallo2.html	20
3.3. Tampilan <i>File</i> Paragraf.html.....	23
3.4. Tampilan <i>File</i> Font.html	24
3.5. Tampilan <i>File</i> Link.html	25
3.6. Tampilan <i>File</i> Tabel.html	27
3.7. Halaman Web Dengan <i>Form</i> Interaktif	28
4.1. Tampilan Halaman Variabel_assignment.php.....	35
4.2. <i>Form</i> Dengan Nama Variabel	37
4.3. Hasil Variabel <i>Form</i> Setelah Disubmit.....	37
4.4. Tampilan <i>Predefine Variable</i>	39
4.5. Tampilan Konstanta	40

4.6. Tampilan Operator Aritmatika.....	41
5.1. Pernyataan Switch Case	47
6.1. Variabel Array \$ProgramStudi	50
6.2. Menampilkan Variabel Array \$ProgamStudi	50
6.3. Menampilkan Variabel Array Asosiatif Dari \$ProgamStudi	53
6.4. Menampilkan Variabel Array Asosiatif Dengan List Dan Each.....	53
6.5. Menampilkan Array 2 Dimensi Dari Matrik.....	54
6.6. Urut Data Array 2 Dimensi Dari Matrik	56
6.7. Urut Data Array.....	57
8.1. Logo Database MySQL.....	63
8.2. Ilustrasi Database MySQL	64
8.3. Layar Awal http://localhost.....	64
8.4. Login Database MySQL	65
8.5. Menciptakan Database Buku	65
8.6. Menciptakan Database Buku Pada Kotak Dialog SQL	66
8.7. Menciptakan Tabel tblbuku Pada Kotak Dialog.....	66
8.8. Menciptakan Tabel tblbuku Dan Primary Key.....	67
8.9. Hasil Tabel tblbuku Dan Primary Key.....	68
8.10. Tampil Data tblbuku	71
8.11. Tambah Data tblbuku	72
8.12. Ubah Data tblbuku.....	74

DAFTAR TABEL

Keterangan	Halaman
3.1. Atribut Warna Body	24
3.2. Atribut Table	26
3.3. Atribut Form	29
3.4. Atribut <i>Type</i> Pada <i>Input</i>	29
4.1. Operator Aritmatika	40
4.2. Operator Logika	42
4.3. Operator Penugasan	42
5.1. Fungsi Mengurutkan Array	55
7.1. Huruf Kapital Dan Kecil	76

DESKRIPSI UMUM

Deskripsi Singkat

Mata kuliah pemrograman web adalah salah satu pemrograman, diantara bahasa pemrograman web yang saat ini sering dipelajari adalah bahasa pemrograman HTML, PHP, ASP, dan Java Script yang sangat dibutuhkan oleh pengguna jaringan internet. Dari hal ini perlu diberikan dasar-dasar pemrograman web khususnya Bahasa Pemrograman PHP yang mana enam puluh persen lebih website dibangun dengan Bahasa Pemrograman PHP. Bahan ajar pemrograman web lebih banyak dibahas tentang Bahasa Pemrograman PHP, dan akan dibahas pula bahasa pemrograman yang terkait yaitu: HTML dan Database MySQL.

Tujuan umum Mata Kuliah Pemrograman Web dengan Bahasa PHP adalah:

- Materi kuliah pemrograman web memberikan pengetahuan konsep dan dasar pemrograman web khususnya Bahasa Pemrograman PHP.
- Membahas perintah dasar dan lanjut Pemrograman PHP.
- Meletakkan pondasi yang mudah dan tingkat mahir Pemrograman PHP.

Tujuan Instruksional Mahasiswa

Mahasiswa mampu mendesain dan membuat aplikasi berbasis web dengan Bahasa Pemrograman PHP dan Database MySQL dengan mudah dan cepat untuk di terapkan dalam dunia pendidikan maupun lembaga lainnya.

Petunjuk Mahasiswa

- a. Mahasiswa diharapkan dapat membaca dahulu bahan ajar secara singkat dan mempraktekkan langsung dilaboratorium komputer atau dirumah.
- b. Apabila mengalami kesulitan seperti instalasi *web server* pada komputer PC atau notebook dan kesulitan memahami materi dapat menanyakan kepada teman yang mampu instalasi komputer dan pemrograman web.

- c. Bahan ajar disajikan dengan mudah, sebaiknya mahasiswa mengikuti prosedur bahan ajar dan penulisan perintah program, karena huruf besar dan kecilnya penulisan, kurang tanda baca, kurang satu huruf perlu diperhatikan yang akan berpengaruh terhadap perintah/eksekusi program.
- d. Mempelajari Bahasa Pemrograman PHP dan Database MySQL melalui buku-buku literatur maupun media jaringan internet sehingga memiliki wawasan dan ilmu pengetahuan yang terbaru.

BAB 1

PENDAHULUAN

Deskripsi Singkat

Bab ini akan mengemukakan internet, sejarah, konsep dan kegunaan Bahasa Pemrograman PHP.

Relevansi

Bab ini membahas dan memberi pengertian kepada mahasiswa atau pembaca tentang penting dan mudahnya Bahasa Pemrograman PHP pada jaringan internet.

Tujuan Instruksional Khusus

Mahasiswa mengerti tentang internet, sejarah, dasar, konsep serta peranan Bahasa Pemrograman PHP.

1.1. Pengenalan Internet

Internet kependekan dari *interconnection-networking*, Jaringan internet merupakan kumpulan jaringan komputer yang saling berkomunikasi dengan aturan komunikasi tertentu. Sebuah situs web di internet umumnya berupa sebuah komputer yang dilengkapi dengan perangkat lunak *web server*, sehingga komputer ini dapat menyediakan informasi yang diakses melalui protokol http atau sistem global *Transmission Control Protocol/Internet Protocol Suite* (TCP/IP).

1.2. Format Dokumen Di Internet

Format dokumen standar di internet saat ini adalah HTML (*Hypertext Markup Language*). HTML menentukan bagaimana informasi ditampilkan sebagai halaman web pada browser. Pemrograman PHP (*Hypertext Preprocessor*) dapat disisipkan kedalam HTML atau berdiri sendiri dalam format dokumen nama-file.PHP.

1.3. Pemrograman Internet

Pemrograman di Internet tidak sama seperti pemrograman pada komputer lokal (misal pada Visual Basic, Visual Foxpro, dan sebagainya). Pemrograman di Internet berkaitan dengan format dokumen yang digunakan di internet. Secara umum ada dua macam pemrograman di internet, yaitu:

Client Side Programming

Server Side Programming

1.3.1. Client Side Programming

Pada pemrograman jenis ini program didownload oleh pengunjung dan dijalankan pada komputer pengunjung. Pemrograman jenis ini memerlukan dukungan *browser* pengunjung. Selain itu ukuran program harus diperhatikan karena jika program terlalu besar, maka waktu yang diperlukan untuk membuka halaman web menjadi lebih lama. Kerugian lainnya adalah pengunjung dapat dengan mudah melihat dan menyalin *source code* program. Contoh program client side adalah : **HTML, JavaScript dan VBScript.**

1.3.2. Server Side Programming

Sesuai dengan namanya, jenis pemrograman ini program dijalankan di dalam *server*. Hasil pengolahannya dikirimkan pada *browser* pengunjung sebagai halaman HTML. Ukuran program relatif dapat lebih besar karena pengunjung tidak perlu men-download programnya, tetapi hanya hasil pengolahannya. Keuntungan lainnya yakni *Source Code* tidak dapat dilihat. Contoh pemrograman

Server Side diantaranya : **CGI/Perl, PHP, ASP, dan JSP.**

1.4. Pemrograman PHP

PHP dikenal sebagai sebuah bahasa *scripting* yang menyatu dengan tag-tag HTML yang dieksekusi di *server* dan digunakan untuk membuat halaman web yang dinamis seperti *guestbook*, statistik pengunjung, polling, email, dan masih banyak lagi. Versi pertama PHP dibuat oleh Rasmus Lerdorf pada tahun

1994 dari setumpuk *script perl* CGI untuk melacak siapa saja yang mengakses resumanya di web. Versi pertama PHP dirilis pada tahun 1995. Rasmus kemudian menulis ulang *script-script perl* tersebut dengan menggunakan bahasa C guna meningkatkan kecepataannya kemudian menambah fasilitas untuk *Form* HTML dan dukungan koneksi ke Database mSQL. Sejak saat itu lahirlah PHP versi kedua yang dikenal sebagai PHP/FI.

PHP versi ketiga dirilis pertengahan tahun 1997. Pada versi ini pembuatannya tidak lagi oleh Rasmus sendiri, tetapi juga melibatkan programmer lain yang antusias untuk mengembangkan PHP. PHP versi 4.0 dirilis bulan oktober 2000. Perubahan mendasar pada PHP 4.0 adalah integrasi Zend Scripting Engine yang dibuat oleh Andi Gutmans dan Zeev Suraski yang menghasilkan banyak peningkatan kecepatan yang sangat besar dibanding versi sebelumnya. Dan sampai saat ini (November 2007) PHP telah mencapai versi 5.0.

Berdasarkan hasil survei di website resmi PHP (www.php.net/usage.php), pada Mei 2002, di seluruh dunia PHP telah dipakai oleh lebih dari 9 juta domain dan pada November 2003 terjadi peningkatan signifikan menjadi lebih dari 14 juta domain. Jumlah ini akan selalu meningkat dengan cepat yang menunjukkan PHP telah semakin populer digunakan di dunia. PHP versi pertama diberi nama Personal Homepage dan entah kapan namanya berubah menjadi PHP Hypertext Preprocessor.

1.5. Mengapa Memilih PHP

Life Cycle yang sangat singkat , sehingga PHP selalu *up to date* mengikuti perkembangan teknologi internet.

Cross Platform, yakni PHP dapat dipakai di hampir semua *web server* yang ada di pasaran (terutama Apache dan Microsoft IIS) dan dijalankan pada berbagai sistem operasi (Linux, Windows, FreeBSD).

PHP mendukung koneksi ke banyak database baik yang gratis maupun komersil, seperti MySQL, mSQL, Oracle, Microsoft SQL Server, Interbase, dan banyak lagi.

PHP bersifat open source dan gratis.

Kemudahan dalam mendapatkan dokumentasi. Di Internet, tidak sulit untuk mencari baik itu referensi, kode-kode PHP yang sudah jadi dan juga mengajukan pertanyaan pada grup-grup diskusi yang di dalamnya banyak sekali para master PHP.

BAB II

INSTALASI WEB SERVER

Deskripsi Singkat

Bab II akan mengemukakan kebutuhan perangkat lunak yang akan digunakan dalam belajar Bahasa Pemrograman PHP dan desain website, dan prosedur instalasi *web server* pada komputer PC maupun notebook.

Relevansi

Bab ini membahas dan memberi petunjuk kebutuhan *web server* dan instalasi sebelum belajar pemrograman web.

Tujuan Instruksional Khusus

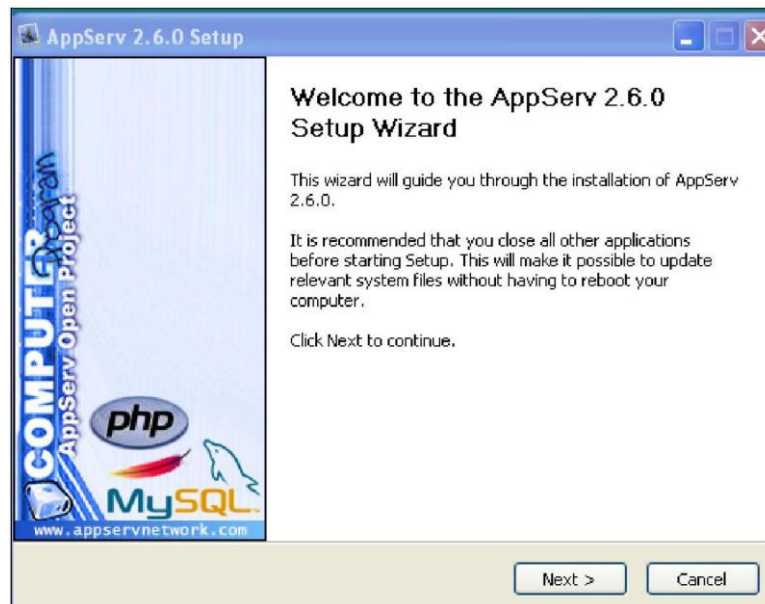
Mahasiswa mengerti tentang kebutuhan pemrograman web dan instalasi perangkat lunak AppServ berbagai versi, Dreamweaver berbagai versi dan aplikasi *browsing* untuk uji coba hasil penulisan Bahasa Pemrograman PHP.

Sebelum dimulai, beberapa peralatan yang harus dipersiapkan diantaranya:

- a. Aplikasi *Web Server*: Appserv-win32-2.6.0
Paket dari AppServ sudah termasuk:
 - 1). Apache sebagai *Web Server*
 - 2). PHP sebagai *script* bahasa pemrograman
 - 3). MySQL sebagai database
 - 4). phpMyAdmin sebagai pengelolaan database
- b. Macromedia Dreamweaver
- c. Aplikasi Browsing

2.1. Instalasi Aplikasi Web Server

- a). Jalankan file Appserv-win32-2.6.0, seperti gambar dibawah ini:



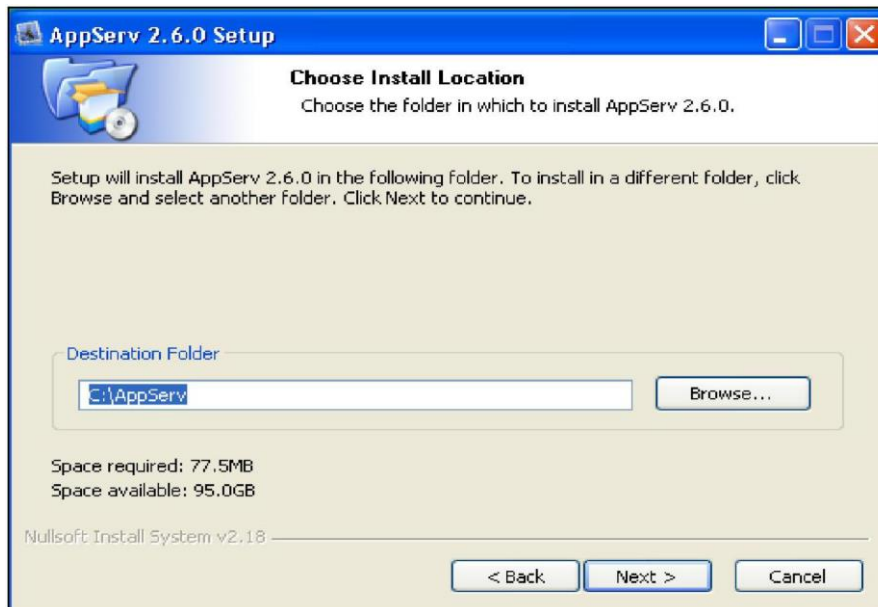
Gambar 2.1. Instalasi Aplikasi *Web Server* Tahap 1

b). Tahap 2 (perjanjian), seperti pada gambar berikut ini:



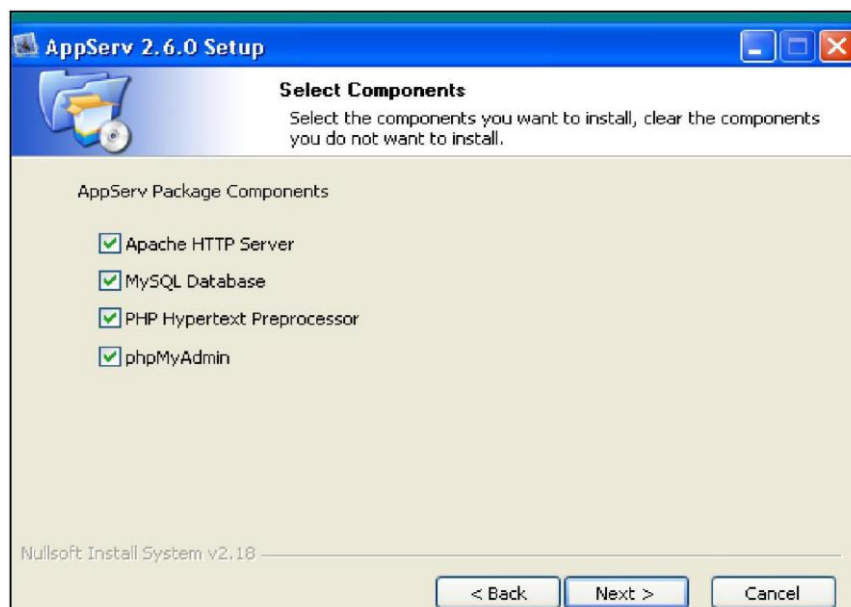
Gambar 2.2. Instalasi Aplikasi *Server* Tahap 2

c). Pilih folder tempat penyimpanan sesuai keinginan (misal D:\AppServ), lalu klik next:



Gambar 2.3. Instalasi Aplikasi Server Tahap 3

- d). Setelah menconteng komponen yang dibutuhkan, pilih 'next', akan terlihat seperti gambar berikut ini:



Gambar 2.4. Instalasi Aplikasi Server Tahap 4

- e). Isi *server name* dan alamat email administrator, lalu pilih 'next'.

AppServ 2.6.0 Setup

Apache HTTP Server Information
Please enter your server's information.

Server Name (e.g. www.appservnetwork.com)

Administrator's Email Address (e.g. webmaster@gmail.com)

Apache HTTP Port (Default : 80)

Gambar 2.5. Instalasi Aplikasi Server Tahap 5

Contoh pengisian pada *form* tersebut diatas:

Isi *Server Name*: www.belajarphp.com

Administrator's Email Address: admin@belajarphp.com.

f). Isi password lalu pilih install, seperti pada gambar berikut:

AppServ 2.6.0 Setup

MySQL Server Configuration
Configure the MySQL Server instance.

Please enter Root password for MySQL Server.

Enter root password

Re-enter root password

MySQL Server Setting

Character Sets and Collations
 UTF-8 Unicode

Old Password Support (PHP MySQL API function.)

Enable InnoDB

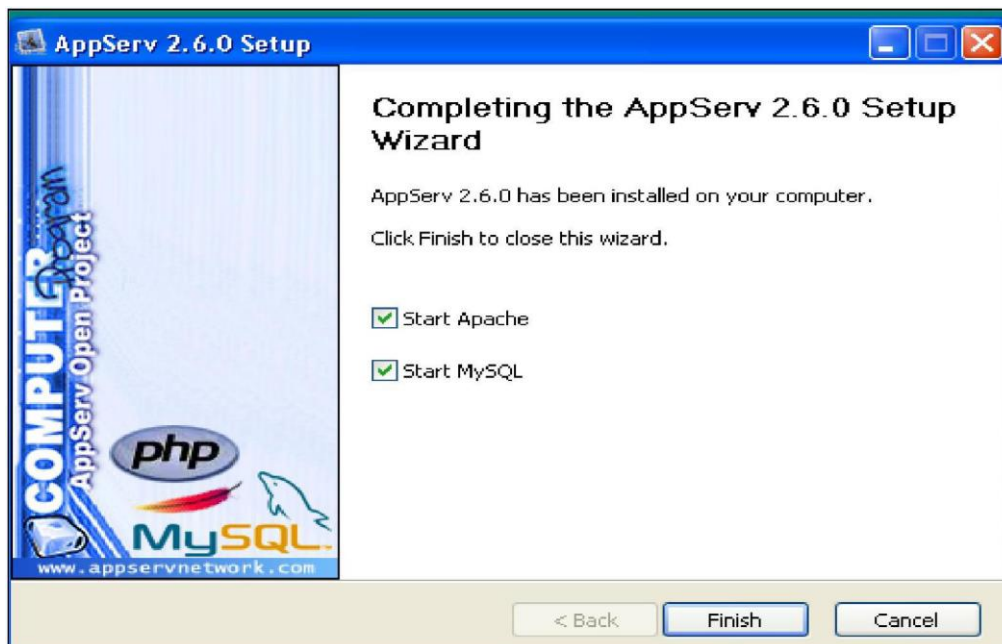
Gambar 2.6. Instalasi Aplikasi Server Tahap 6

Pada *form* tersebut diatas isilah:

Enter root password: 123456

Re-enter root password: 123456

- g). Pilih Finish seperti gambar dibawah ini, maka instalasi sukses dan setelah komputer di *restart* aplikasi dapat digunakan.



Gambar 2.7. Instalasi Aplikasi Server Tahap 7

2.2. Macromedia Dreamweaver

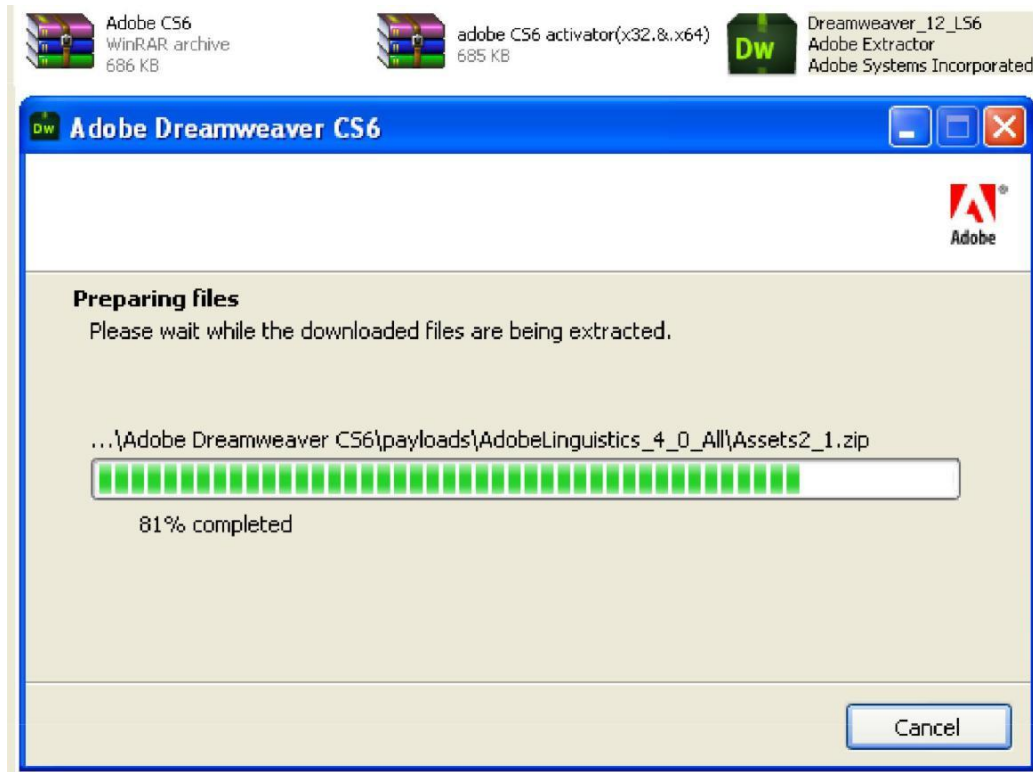
File master yang digunakan tersebut dapat digambarkan dibawah ini:

Aplikasi > Adobe.Dreamweaver.CS6 >			
Name	Date modified	Type	Size
adobe CS6 activator(x32.&x64).zip	10/05/2013 03:39	WinRAR ZIP archive	685 KB
Adobe CS6.rar	10/05/2013 03:49	WinRAR archive	686 KB
Dreamweaver_12_LS6.exe	10/05/2013 01:56	Application	290,086 KB

Gambar 2.8. File Master Dreamweaver CS6

Install Macromedia Dreamweaver CS6 dengan menjalankan *file* Dreamweaver_12_LS6.exe sampai dengan selesai, sehingga hasil *install* akan disimpan pada *directory* Macromedia Dreamweaver CS6 (C:\Program Files (x86)\Adobe\Adobe Dreamweaver CS6).

Setelah file Dreamweaver_12_LS6.exe dijalankan seperti gambar dibawah ini (sudut kanan atas), kemudian instalasi akan dilakukan dengan tahap 1 yaitu “Preparing files” sebagai proses *download file* dan melakukan *extract file*.



Gambar 2.9. Install Aplikasi Macromedia Dreamweaver CS6 Tahap 1

Beberapa *file* yang diekstrak berasal dari Dreamweaver_12_LS6.exe, seperti gambar diatas yaitu Assets2_1.zip.

Tahap 1 telah diproses, selanjutnya instalasi akan dilakukan dengan tahap 2 seperti gambar dibawah ini. Tahap 2 ada dua pilihan, yaitu:

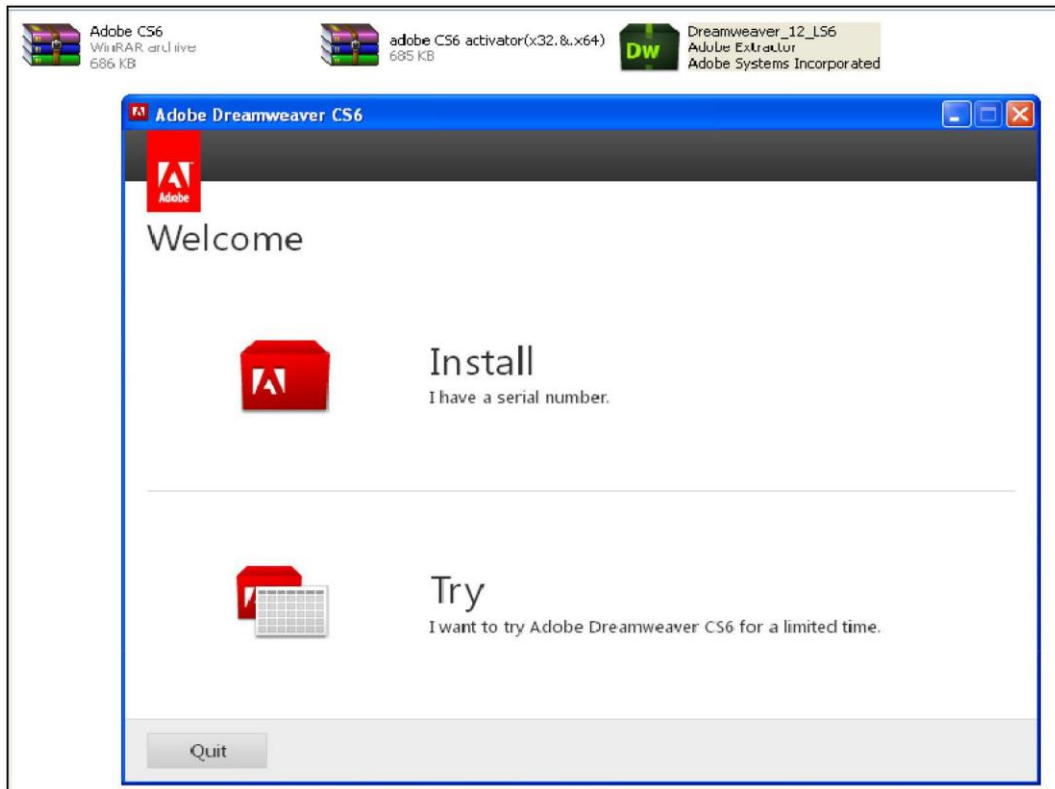
a. Install

Pilihan *install* ini apabila pembeli perangkat lunak Dreamweaver CS6 memiliki nomor Serial Number yang terdapat pada label *compact disk*.

b. Try

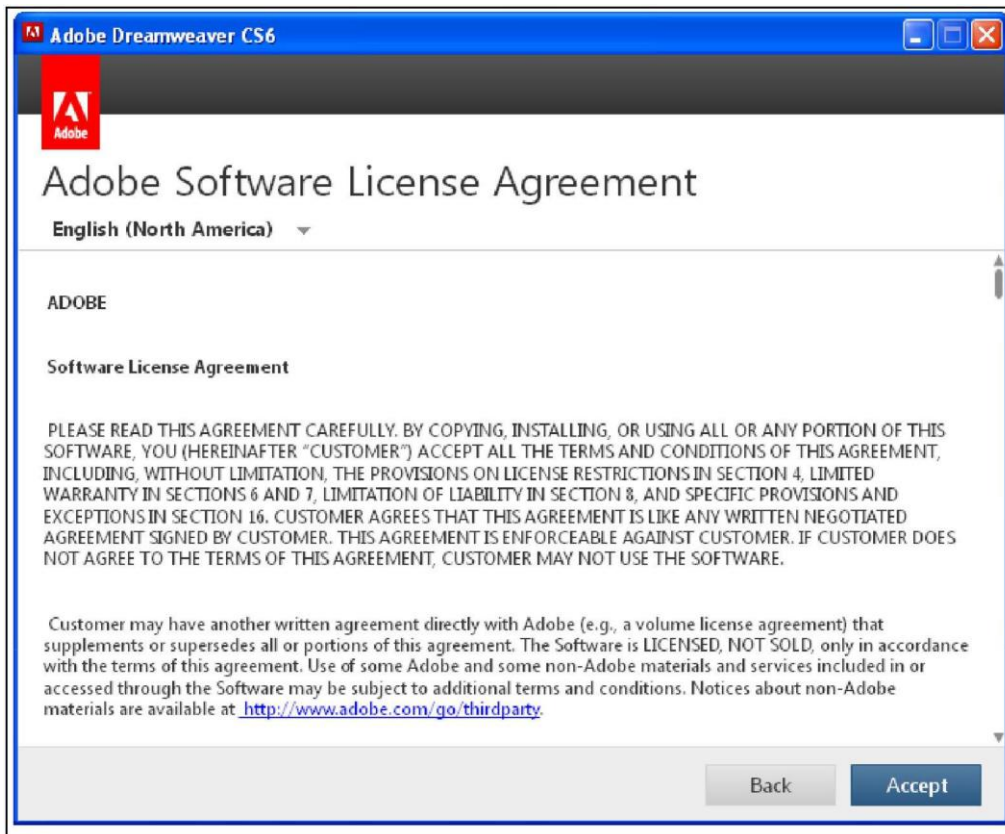
Pilihan ini dilakukan apabila pembeli perangkat lunak tidak memiliki nomor Serial Number yang diberlakukan oleh pembuat perangkat lunak

Dreamweaver CS6. Waktu pemakaian perangkat lunak ini dibatasi selama tiga puluh hari saja.





Gambar 2.10. Install aplikasi Macromedia Dreamweaver CS6 Tahap 2

Tahap 3, instalasi ini memiliki perjanjian terhadap lisensi perangkat lunak Adobe Dreamweaver CS6. Setelah menerima perjanjian tersebut akan dilakukan proses instalasi dan aktivator sampai selesai.

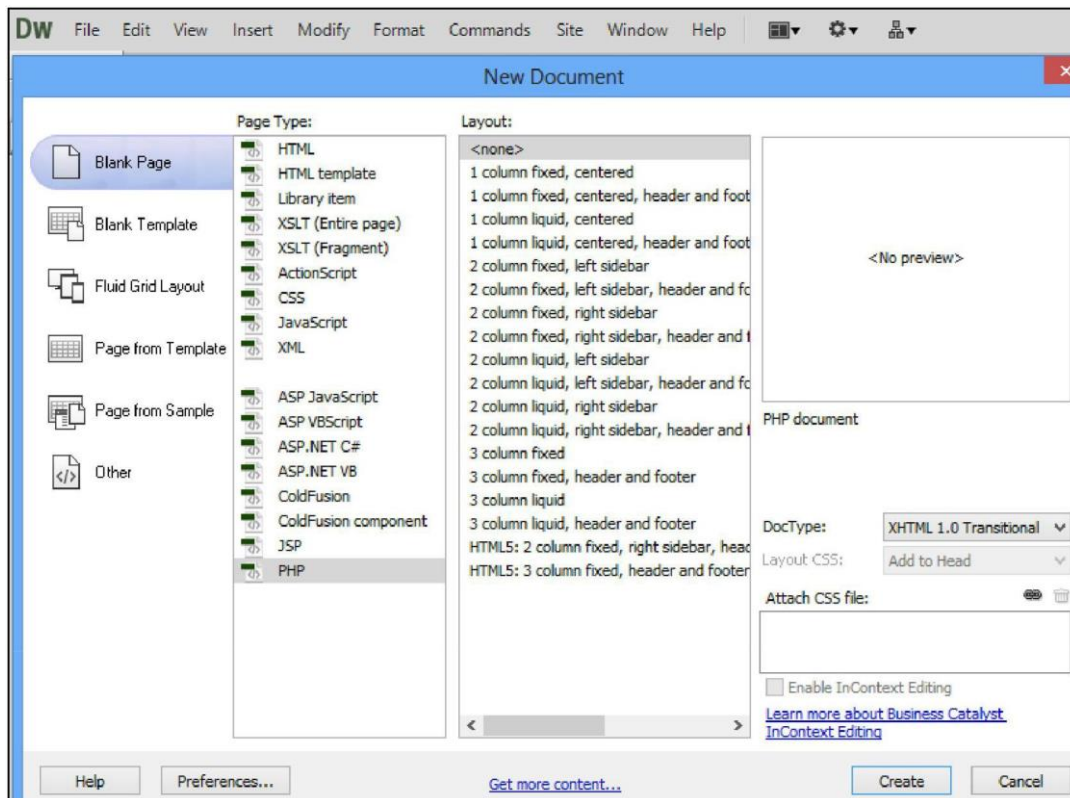


Gambar 2.11. *Install* Aplikasi Macromedia Dreamweaver CS6 Tahap 3

Aplikasi ▸ Adobe.Dreamweaver.CS6 ▸ adobe.cs6.all.products.activator.(x32.&.x64)-MPT

Name	Date modified	Type	Size
 adobe.cs6.all.products.activator.(x32.y.x6...	05/05/2012 20:45	Application	693 KB
 adobe.txt	09/05/2012 23:11	Text Document	1 KB
 MPT.nfo	22/09/2011 23:23	System Informatio...	8 KB

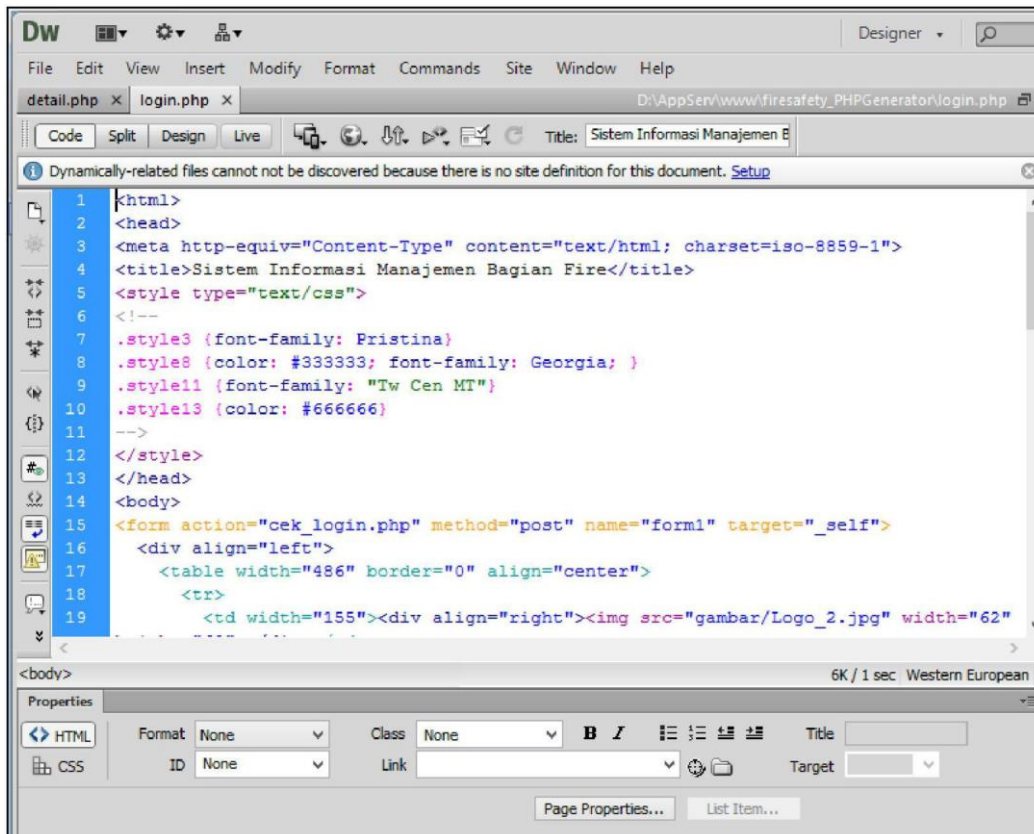
Gambar 2.12. *File* Aktivator Macromedia Dreamweaver CS6



Gambar 2.13. Tampilan Awal Macromedia Dreamweaver CS6

Macromedia Dreamweaver CS6 memiliki banyak penulisan *script* bahasa pemrograman seperti PHP, JSP, ASP, XML, dan sebagainya seperti gambar diatas.

Macromedia sebagai salah satu *utility* yang sangat membantu programmer terhadap penulisan *script* bahasa pemrograman berbasis web, salah satu contoh apabila terjadi kesalahan penulisan *script* bahasa pemrograman akan diberikan pesan kesalahan. Gambar dibawah ini merupakan salah satu proses penulisan *script* bahasa pemrograman PHP dan dipadukan dengan pemrograman HTML, yaitu:



Gambar 2.14. Tampilan *Script* Pada Macromedia Dreamweaver CS6

2.3. Aplikasi Browsing

Browsing dapat digunakan: Internet Explorer, Mozilla Firefox, dan lainnya yang berguna dalam menjalankan perintah-perintah program yang telah dibuat.



Gambar 2.15. *Browsing* Internet

Internet Explorer dan Mozilla Firefox merupakan cara tercepat, teraman, dan terbaik untuk menjelajah dunia maya. Selain itu *software* ini memiliki kelebihan salah satunya, bisa *browsing* apa saja yang ada di perangkat lunak ini, dan biasa aksesnya sangat cepat sekali.

Untuk melihat apakah Aplikasi *Web Server* bekerja dengan baik, caranya adalah dengan mengeceknya melalui *browser*. Jalankan salah satu aplikasi *browser*, dan ketikkan di Address Bar-nya: `http://localhost`. Jika dokumen web yang sudah diinstall akan menampilkan halaman “AppServ Open Project 2.6.0”, maka dapat disimpulkan *Web Server Apache* sudah berjalan dengan baik.



Gambar 2.16. *Web Server Apache*

BAB III

DASAR-DASAR HTML

Deskripsi Singkat

Bab III akan mengemukakan dasar-dasar perintah dalam Bahasa Pemrograman HTML, hal ini diajarkan sebagai langkah awal mendesain *website*.

Relevansi

Dasar mendesain *website* sebaiknya digunakan Bahasa Pemrograman HTML, diharapkan adanya hubungan dengan perintah-perintah Bahasa Pemrograman PHP.

Tujuan Instruksional Khusus

Mahasiswa mengerti tentang pemrograman web yang dibangun dengan Bahasa Pemrograman HTML dan dirancang dengan aplikasi Dreamweaver.

3.1. Mengenal HTML

HTML (HyperText Markup Language) adalah “bahasa” yang digunakan pada untuk membuat halaman web di internet. HTML memungkinkan untuk menampilkan teks, gambar, link ke halaman lain, membuat *form* isian, menyisipkan video, suara, dan lain-lain.

Dokumen HTML sebenarnya hanya berupa file teks biasa yang dilengkapi dengan tanda-tanda khusus (tag) yang menentukan bagaimana teks tersebut ditampilkan. Karena itu, dokumen HTML dapat dibuat menggunakan teks editor sederhana seperti Notepad, EditPlus, vi, dan sebagainya. Namun sekarang ini telah banyak visual editor untuk HTML seperti: Microsoft Front Page, Macromedia Dreamweaver, Adobe GoLive, dan lain-lain.

Selanjutnya dapat membuat satu dokumen HTML sederhana misal dengan menggunakan Macromedia Dreamweaver CS6 sesuai instalasi sebelumnya dan disimpan (*Save as*) sebagai *Webpage* (*.htm, *.html), contoh dengan nama Hallo.html. Perintah HTML ditandai dengan:

```
<html>  
..... perintah program .....  
</html>
```


Tempatkan dokumen tersebut di *directory* (D:\AppServ\www\belajarphp\)\ tertentu. Sisipkan perintah program diantara `<body> </body>`, sehingga seperti perintah program dibawah ini:

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

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Untitled Document</title>
</head>
<body>

<p><strong>Haloooo o... </strong></p> <p><strong>Ini
Homepage Pertama Saya</strong></p>
</body>
</html>
```

Kemudian buat *folder* di *folder* www dengan nama: Belajarphp, dan simpan perintah program tersebut diatas pada folder Belajarphp (D:\AppServ\www\Belajarphp) dengan nama *file* Hallo.html. Dengan aplikasi *browsing* Mozilla Firefox ketik pada *address bar*-nya: localhost/belajarphp/ dan klik Hallo.html, atau localhost/belajarphp/Hallo.html akan terlihat seperti gambar dibawah ini:



Gambar 3.1. Hallo.html

Berikut ini contoh lain HTML sederhana, kemudian simpan dengan nama Hallo2.html pada *folder* D:\AppServ\www\Belajarphp:

```
<html>
```

```

<head>

    <title>Judul Yang Muncul Pada Title Bar Browser</ title >
</head>
<body>

<h1>Helo World</h1>
Di dalam BODY ditempatkan teks, gambar, link atau isi dokumen Anda.

</body>
</html>

```

Jika dokumen ini dibuka pada *web browser* akan tampak seperti pada gambar di bawah ini.



Gambar 3.2. Tampilan dokumen HTML *File* Hallo2.html

3.2. Head, Body Dan Title

Dokumen HTML dapat dibagi menjadi dua bagian, yaitu bagian kepala (**HEAD**) dan tubuh (**BODY**). Bagian kepala ditandai dengan tag **<head> ... </head>**. Pada bagian ini biasanya diisi dengan informasi-informasi tambahan yang berguna bagi *web browser* ketika menampilkan halaman tersebut, atau informasi untuk *search engine*. Bagian tubuh diapit tag **<body>...</body>**, isi dokumen dan perintah-perintah HTML lainnya diletakkan di bagian ini.

Salah satu elemen penting yang terletak pada bagian **head** adalah **TITLE**. Seperti tampak pada Hallo2.html di atas, teks yang diapit tag **<title>...</title>** ditampilkan pada *title bar web browser*. Teks ini adalah judul halaman *web* bersangkutan yang berguna untuk memberi gambaran umum pada pengunjung tentang isi halaman bersangkutan.

Tag *heading*, yang berfungsi untuk memformat *heading* (judul/sub judul) dalam sebuah dokumen. Ada 7 buah heading yang dikenal dalam HTML yaitu

<H1> sampai dengan **<H7>**. Untuk mengetahui ukuran tiap *heading*, dapat dicoba baris-baris berikut:

```
<html>
<head>
  <title>Latihan Heading</ title >
</ head >
<body>
  <h1>Heading 1</h1>
  <h2>Heading 2</h2>
  <h3>Heading 3</h3>
  <h6>Heading 6</h6>
  <h7>Heading 7</h7>
</ body >
</ html >
```

3.3. Paragraf

Untuk memformat paragraf, bisa menggunakan tag **<p>**. Dengan tag ini dapat menentukan alignment (posisi) dari dokumen HTML. Untuk mengatur posisi tulisan dapat menggunakan atribut ALIGN. Atribut Align diikuti dengan posisi yang diinginkan.

Selain tag **<p>** ada pula tag **
** untuk pindah baris, **<hr>** untuk membuat garis, **<blockquote>** untuk membuat kutipan, **<pre>** untuk menampilkan *text* sama dengan yang diketikkan dalam dokumen HTML, misal dengan perintah berikut:

```
<html>
<head>
  <title>TEST</title>
</head>
<body>
Hallo
  Hallo
  Hallo
</body>
</html>
```

Maka yang akan ditampilkan oleh *browser* adalah:

HalloHalloHallo

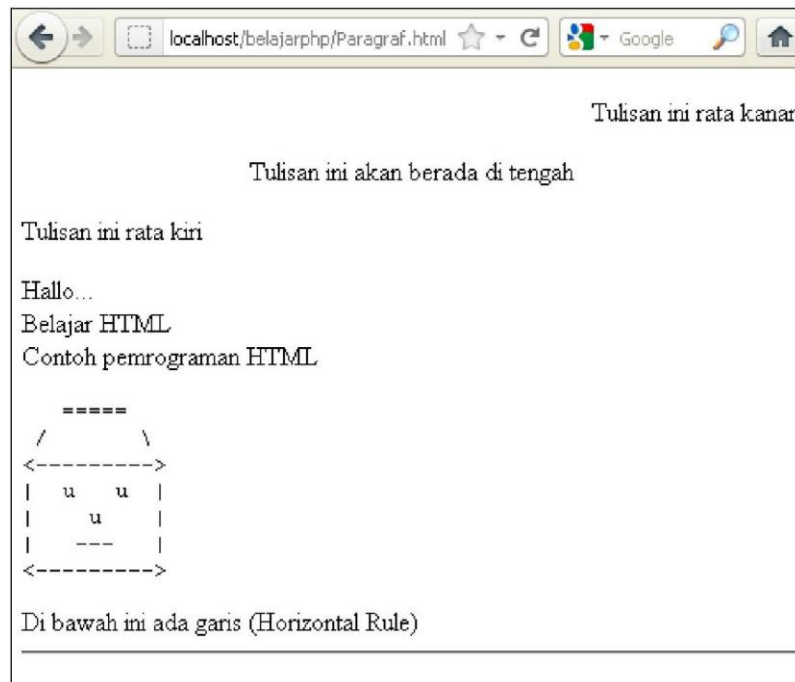
Hal itu terjadi karena HTML tidak mengenali tombol **Enter** sehingga tombol tersebut diabaikan. Untuk mengatasi hal tersebut maka digunakan tag **<pre>** tersebut.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Contoh Paragraf</title>
</head>

<body>
<p align="right"> Tulisan ini rata kanan
<p align="center"> Tulisan ini akan berada di tengah
<p align="left"> Tulisan ini rata kiri <p align="left">

Hallo...<br>
Belajar HTML <br>
Contoh pemrograman HTML<br>
<pre>
=====
/      \
<----->
| u u |
|  u  |
|  --- |
<----->
</pre>
Di bawah ini ada garis (Horizontal Rule) <hr>

</body>
</html>
```



Gambar 3.3. Tampilan *File* Paragraf.html

3.4. Pengaturan Huruf (Font)

Untuk mengatur jenis, ukuran, warna suatu font, bisa menggunakan tag ``, tag `` ini memiliki beberapa atribut diantaranya FACE dan COLOR.

Atribut Face digunakan untuk menentukan jenis font yang akan digunakan, misal menggunakan jenis font Comic Sans MS, maka dapat ditulis perintahnya ``. Sedangkan COLOR berfungsi untuk menentukan warna font, bisa dengan cara menyebutkan nama warna seperti red, green dan blue atau menggunakan nilai RGB (Red Green Blue) dari suatu warna, misalnya FF0000 untuk red, 00FF00 untuk green dan 0000FF untuk blue. Ada juga atribut SIZE untuk menentukan ukuran font.

```
<html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

```
<title>Belajar HURUF</title>
```

```
</head>
```

```
<body>
```

```
<b>Huruf ini ditampilkan jadi Tebal (BOLD)</b> <br>
```

```
<font face="Arial Black, Gadget, sans-serif" color="blue" size=4>
```

Huruf ini Arial Black, warna merah dan ukurannya 4

```
</font><br>
```

```
<font face="Verdana, Geneva, sans-serif" color="red" size=2>
```

Ini menggunakan Verdana, warna biru dan ukurannya 2

```
</font>
```

```
</body>
```

```
</html>
```



Gambar 3.4. Tampilan *File* Font.html

3.5. Warna Dokumen

Suatu dokumen web akan terlihat lebih menarik jika diberi warna-warna atau gambar sebagai *background*-nya. Untuk itu, pada HTML tersedia atribut-atribut untuk mengatur hal tersebut, atribut tersebut diletakkan pada tag <BODY> sehingga efeknya akan dapat dilihat di seluruh dokumen. Atribut yang digunakan dalam tag <BODY> yaitu:

Tabel 3.1. Atribut Warna Body

ATRIBUT	FUNGSI
BGCOLOR	Menentukan warna background suatu halaman web
BACKGROUND	Menentukan gambar/images yang akan dipasang sebagai background halaman web
TEXT	Menentukan warna teks normal
LINK	Menentukan warna teks yang berfungsi sebagai link
ALINK	Menentukan warna teks link ketika diklik
VLINK	Menentukan warna teks link ketika selesai diklik

Contoh pemakaiannya adalah sebagai berikut:

```
<body bgcolor="yellow" text="black" link="blue" alink="red" vlink="green">
```

3.6. Link

Dalam web, selalu ada yang dinamakan link, link berfungsi menghubungkan antara satu halaman dengan halaman lainnya. Untuk link ini HTML menyediakan tag <A> (Anchor). Tag <A> memiliki beberapa atribut yaitu: HREF untuk menghubungkan antara satu halam dengan halaman lainnya, dan NAME untuk memberikan nama pada suatu paragraf/tulisan dalam dokumen HTML, sehingga mempermudah navigasi dalam suatu halaman web.

```
<html>
<head>
<title>Belajar LINK</title>
</head>
<body>
  <a name="atas">Bagian ini tanda bagian atas</a> <br>
  <a href="hallo.html">Link ke Hallo (hallo.html)</a><br>
  <a href="hallo2.html">Link ini halaman Hallo2 (hallo2.html)</a><br>
  <a href="font.html">Link ini halaman Font (font.html)</a><br>
  <a name="#atas">Link ke tempat yang ditandai</a> <br>
</body>
</html>
```



Gambar 3.5. Tampilan *File* Link.html

3.7. Menggunakan Tabel

Untuk membuat tampilan *website* yang bagus, kebanyakan para desainer web menggunakan tabel untuk mempermudah pengerjaan mereka, dimana tabel digunakan untuk mempercantik situs tersebut. Untuk membuat tabel, HTML menyediakan tag <TABLE>. Dalam tag <TABLE> terdapat banyak atribut, yaitu:

Tabel 3.2. Atribut Table

ATRIBUT	FUNGSI
BORDER	Menentukan ukuran border/garis tabel
WIDTH	Menentukan lebar tabel
HEIGHT	Menentukan tinggi tabel
BGCOLOR	Menentukan background tabel
BACKGROUD	Menentukan gambar yang digunakan untuk background tabe

Untuk membuat baris, setelah menuliskan tag <TABLE> maka digunakan tag <TR>. Sedangkan untuk membuat kolom digunakan tag <TD>. Sebagian besar atribut dalam tag <TABLE> dapat digunakan dalam tag <TR> maupun dalam tag <TD>.

Untuk menggabungkan baris/kolom (merge) digunakan atribut COLSPAN dan ROWSPAN. COLSPAN digunakan untuk menggabungkan beberapa kolom menjadi 1 sedangkan ROWSPAN menggabungkan beberapa baris menjadi 1.

CELLPADDING digunakan untuk mengatur spasi antara border dengan tulisan, sedangkan CELLSPACING digunakan untuk mengatur spasi antar dua buah sel.

```

</html>

<title>Belajar TABEL</title>
</head>
<body>
<table width="60%" border="1" cellspacing="2" cellpadding="3">
  <tr bgcolor="green">
    <td width="31">NO.</td>
    <td width="79">NIM</td>
    <td width="165">NAMA</td>
    <td width="269">ALAMAT</td>
  </tr>
  <tr>
    <td align="center">1</td>
    <td>201212013</td>

```



```

<td>Andi Eko</td>

<td>Jl Cemara No.27 Bontang Utara</td>
</tr>
<tr>
<td align="center">2</td>
<td>201312005</td>
<td>Agussalim Muis</td>
<td>Jl Makassar No.22 Bontang Baru</td>
</tr>
<tr>
<td align="center">3</td>
<td>201312018</td>
<td colspan="2">Nur Aeni Jl Biawan No.108 Tanjung Laut. Sudah tidak
aktif kuliah</td>
</tr>
</table>
</body>
</html>

```



NO	NIM	NAMA	ALAMAT
1	201212013	Andi Eko	Jl Cemara No.27 Bontang Utara
2	201312005	Agussalim Muis	Jl Makassar No.22 Bontang Baru
3	201312018	Nur Aeni Jl Biawan No.108 Tanjung Laut. Sudah tidak aktif kuliah	

Gambar 3.6. Tampilan *File* Tabel.html

3.8. Membuat Form

Untuk melakukan interaksi dengan *user*, biasanya suatu web menggunakan *form* untuk menerima masukan/*input* dari *user* dan memproses hasil *input*-an tersebut di *server*. Misalkan saja:

The image shows a screenshot of a Mozilla Firefox browser window. The title bar reads 'Form BIODATA - Mozilla Firefox'. The address bar shows 'localhost/belajarphp/biodata.html'. The main content area displays a form titled 'BIODATA' with the following elements:

- A text input field labeled 'NIM'.
- A text input field labeled 'NAMA'.
- A section titled 'JENIS KELAMIN' with two radio buttons: 'LAKI-LAKI' (unselected) and 'PEREMPUAN' (selected).
- Two buttons at the bottom: 'SIMPAN' and 'BATAL'.

Gambar 3.7. Halaman Web dengan *Form* Interaktif

Dalam gambar diatas dapat dilihat untuk nama, dengan menggunakan tipe textbox, *password* menggunakan tipe password, untuk jenis kelamin menggunakan radio button dan untuk hobi menggunakan checkbox. Untuk mengatur tipe *form*, menggunakan atribut TYPE, struktur *form* dapat dibuat sebagai berikut:

```
<form name="form1" method="....." action="....">
  <input type="....." name="....." id="....." value="....." />
  <input type="....." name="....." id="....." value="....." />
  <input type="submit" name="....." value="....." />
</form>
```

Untuk membuat *form* (biodata.html) dalam HTML dapat ditulis:

```
<html>

<head>
  <title>Form BIODATA</title>
</head>

<body>
<p>BIODATA</p>
<form name="form1" method="post" action="biodata_tambah.php">
```

```

<p>NIM
    <input type="text" name="NIM" id="NIM" />
</p>
<p>NAMA
    <input type="text" name="NAMA" id="NAMA" />
</p>
<p>JENIS KELAMIN</p>
<p>
    <input name="radio" type="radio" id="LK" value="LK" checked="checked" />
    LAKI-LAKI</p>
<p>
    <input name="radio" type="radio" id="PR" value="PR" checked="checked" />
    PEREMPUAN</p>
<p>
    <input type="submit" name="simpan" id="simpan" value="SIMPAN" />
    <input type="reset" name="batal" id="batal" value="BATAL" />
</p>
</form>
</body>
</html>

```

Atribut-atribut diatas dapat dijelaskan sebagai berikut:

Tabel 3.3. Atribut Form

ATRIBUT	FUNGSI
METHOD	Metode pengiriman data ke file tujuan. Ada 2 yaitu POST dan GET
ACTION	Aksi yang akan dilakukan jika user menekan tombol Submit
NAME	Memberikan nama tiap <i>input</i>
VALUE	Memberikan nilai suatu <i>input</i>
TYPE	Tipe <i>form</i> yang akan digunakan

3.8.1. Input

Tag <INPUT> digunakan untuk menampung masukan dari pengunjung. Bentuk masukan dapat diubah-ubah sesuai kebutuhan dengan atribut TYPE. Berikut ini adalah nilai-nilai yang mungkin untuk atribut TYPE pada tag <INPUT>:

Tabel 3.4. Atribut Type Pada *Input*

ATRIBUT	FUNGSI
TEXT	Kotak teks biasa
PASSWORD	Kotak password (teks ditampilkan sebagai *)
CHECKBOX	Checkbox dapat dipilih lebih dari satu pilihan
RADIO	Radio hanya dapat dipilih satu saja
IMAGE	Tombol berupa gambar
BUTTON	Tombol

SUBMIT	Tombol untuk mengirimkan isi <i>form</i>
RESET	Tombol untuk mengembalikan isian ke nilai default

Tag <INPUT> umumnya memiliki atribut NAME dan VALUE. Setelah *form* dikirimkan pada *web server*, NAME biasanya akan dijadikan nama variabel yang menampung nilai VALUE.

3.8.2. Select Dan Option

Tag <SELECT> memungkinkan pengguna memilih salah satu pilihan yang disediakan. Pilihan ditampilkan menggunakan tag <OPTION>.

3.8.3. Textarea

Tag <TEXTAREA> digunakan untuk menerima masukan yang terdiri dari banyak baris, seperti komentar pada buku tamu. Tinggi dan lebar text area dapat ditentukan dari atribut ROWS dan COLS.

3.9. Komentar

Komentar dapat dituliskan di dalam dokumen HTML dengan cara:
<!-- komentar -->

BAB IV

VARIABEL, KONSTANTA DAN OPERATOR

Deskripsi Singkat

Bab ini dijelaskan tentang dasar-dasar perintah PHP, variabel, konstanta dan operator sebagai dasar Bahasa Pemrograman PHP.

Relevansi

Dasar membuat variabel, konstanta dan operator sebagai langkah awal belajar Bahasa Pemrograman PHP yang sangat perlu diketahui.

Tujuan Instruksional Khusus

Mahasiswa mengerti tentang dasar pemrograman PHP, membuat variabel, konstanta dan operator.

4.1. Memulai PHP

Perintah-perintah PHP dapat dipadukan dengan dokumen HTML sehingga menghasilkan halaman web yang dinamis dengan catatan nama *file* disimpan dalam bentuk php (namafile.php). Untuk lebih jelasnya lihat contoh berikut:

```
<HTML>
<HEAD>
<TITLE>Memulai PHP </TITLE>
</HEAD>
<BODY>

<?php
    // memulai php
    echo "Hallo, awal belajar PHP";
?>

</BODY>
</HTML>
```

Pada contoh di atas, yang disebut kode PHP adalah kode yang berada diantara tag pembuka `<?php` dan tag penutup `?>`. Perintah PHP `echo` berfungsi untuk menampilkan tulisan "Hallo, awal belajar PHP" pada halaman web.

4.2. Menulis PHP

Seperti contoh di atas, antara kode PHP dan HTML dipisahkan oleh tag pembuka `<?php` dan tag penutup `?>`. Sebenarnya ada 4 cara untuk memisahkan PHP dan HTML, yaitu:

Pada contoh di atas, umumnya digunakan instruksi dibawah ini:

```
<?php echo ("Hallo, awal belajar PHP"); ?>
```

Dapat digunakan instruksi yang lebih singkat, contoh:

```
<? echo ("Hallo, awal belajar PHP"); ?>
```

Cara ketiga adalah cara yang biasa dipakai untuk JavaScript dan VBScript, contoh:

```
<script language="php"> echo ("Inilah kode PHP-ku"); </script>
```

Sejak PHP 3.0.4, bisa menggunakan tag pembuka dan penutup seperti pada ASP, contoh:

```
<% echo ("PHP = Professional Home Page"); %>
```

4.3. Komentar

Komentar sangat berguna untuk memperjelas isi kode PHP dan memudahkan saat *debugging*. PHP mengenal beberapa cara penulisan komentar, yaitu:

- C style. Komentar diawali dengan tanda pembuka `/*` dan penutup `*/`, umumnya dipakai untuk banyak baris.
- C++ style. Komentar diawali dengan tanda `//`. Hanya untuk 1 baris.
- Unix Shell style. Komentar diawali dengan tanda `#`. Juga hanya untuk 1 baris.

4.4. Meng-escape Karakter

Menulis teks di bawah ini di *browser* dengan menggunakan PHP:
Selamat Datang. "PHP" Awal belajar PHP. "Terima Kasih"

Ada yang berpikiran misalnya adalah begini:

```
<?php
  echo " Selamat Datang. "PHP" Awal belajar PHP. "Terima Kasih' ";

?>
```

Ternyata cara penulisan di atas akan menimbulkan error, karena terdapat tanda double-quote dan apostrof di dalam string yang dicetak. Untuk mengatasinya, tanda double-quote dan apostrof harus diescape terlebih dahulu sebelum di echo. Escape karakter dilakukan dengan menulis tanda backslash (\) sebelum karakter tersebut. Jadi, cara yang benar adalah

```
<?php
    echo " Selamat Datang. \"PHP\" Awal belajar PHP. \"Terima Kasih\" ";
?>
```

Contoh lain adalah jika Anda ingin menulis kode HTML dengan statement echo:

```
<?php
echo "<table border=\"1\" align=\"center\">
<tr>
    <td>
        <font face=\"arial\" size=\"2\"> Selamat Datang. </font>
    </td>
</tr>
</table>";
?>
```

4.5. Variabel dan Konstanta

4.5.1 Variable

PHP sebagai suatu bahasa pemrograman memiliki variabel, sama halnya dengan bahasa pemrograman lainnya. Variabel berfungsi untuk menyimpan beberapa bit data, sehingga untuk memanggil suatu data dapat dipanggil variabel tersebut. PHP, dalam menulis suatu variabel selalu dimulai dengan tanda '\$'. Variabel setelah penulisan diawali '\$' (misal \$a), selanjutnya variabel akan dapat diisi data dalam bentuk integer, real maupun string. PHP akan secara otomatis mengkonversi data menurut tipenya.

Umumnya, variabel berasal dari 3 sumber:

- Pertama, variabel secara eksplisit diisi (*assign*) dalam perintah program.
- Kedua, variabel berasal dari halaman HTML biasanya dari *form input*.
- Ketiga, variabel bisa berasal dari lingkungan PHP, yaitu variabel yang *pre-defined* dalam PHP.

a. Assignment Variabel

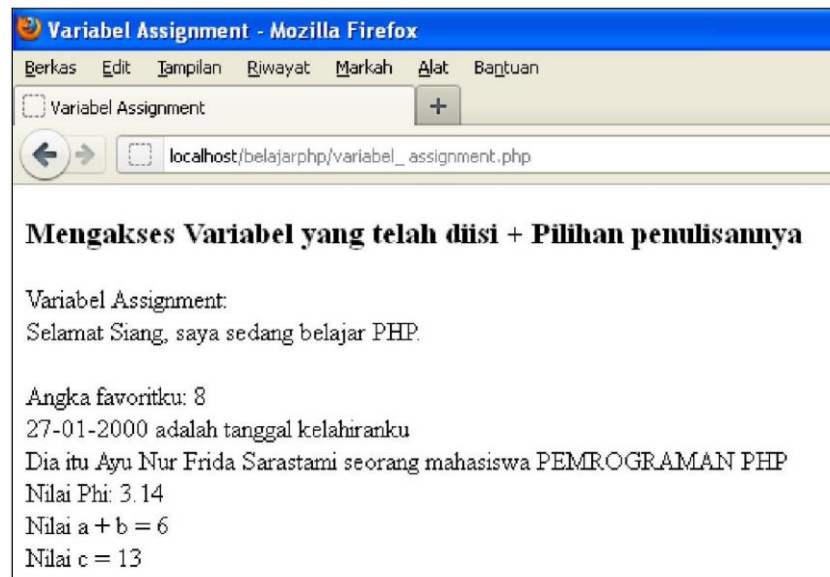
Dengan variabel PHP, membuat tugas selanjutnya akan menjadi sangat mudah. Dalam perintah PHP, tidak perlu mendefinisikan variabel terlebih dahulu sebelum mengisi atau memakainya. Apapun yang dibutuhkan masukkan nilai atau data ke sebuah variabel, baik nilai dalam jenis string, integer, float, array, dan sebagainya, tanpa harus mendefinisikan jenis variabelnya. Contoh *assignment* (pengisian) variabel:

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

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Variabel Assignment</title>
</head>

<body>
<?
$pesan = "Selamat Siang, saya sedang belajar PHP."; //mengisi string
$angkafavoritku = 8; //mengisi integer
$Tglahir = "27-01-2000"; //mengisi string
$phi = 3.14; //mengisi bilangan floating point (real)
$a = 2;
$b = 4;
$c = 9+6-$a;
$nama = "Ayu Nur Frida Sarastami";
echo "<h3>Mengakses Variabel yang telah diisi + Pilihan penulisannya</h3>";
echo "Variabel Assignment:<br> $pesan <br><br>"; //penulisan variable biasa dalam " "
echo "Angka favoritku: $angkafavoritku<br>"; //penulisan variable biasa dalam " " echo
$Tglahir ." adalah tanggal kelahiranku <br>"; //penulisan variable di depan
echo "Dia itu ".$nama." seorang mahasiswa PEMROGRAMAN PHP <br>"; // disisipkan
echo "Nilai Phi: ". $phi." <br>"; //penulisan variable di belakang
echo "Nilai a + b = ", $a+$b." <br>"; //proses penjumlahan variabel hasilnya 6
echo "Nilai c = ", $c; //proses penjumlahan variabel hasilnya 6 ?>

</body>
</html>
```

Gambar 4.1. Tampilan Halaman Variabel_ assignment.php

b. Mengakses Variabel yang berasal dari form input HTML

Pada *form* biodata.html sebelumnya, yang telah dibuat pada Bab III (3.8. Membuat *Form*) yang merupakan *form* biodata, *form* tersebut belum bisa berfungsi karena apabila diklik button SIMPAN, *form* tersebut tidak melaksanakan perintah selanjutnya. Pada *form* biodata2.php seperti dibawah ini akan melakukan perintah selanjutnya yaitu SIMPAN.

Pada halaman biodata2.php dapat mengakses nilai dari variabel-variabel yang telah dikirimkan (disubmit). Berikut adalah kode dari halaman biodata2.php:

```
<body>
<p>BIODATA</p>
<form id="form1" name="form1" method="post" action="">
  <p>NIM
    <input type="text" name="NIM" id="NIM" />
  </p>
  <p>NAMA
    <input type="text" name="NAMA" id="NAMA" />
  </p>
  <p>PASSWORD ANDA
    <input type="password" name="katakunci" id="katakunci" />
  </p>
  <p>JENIS KELAMIN:</p>
  <p>
    <input type="radio" name="kelamin" id="kelamin" value="LK" checked="checked"
  />
  LAKI-LAKI</p>
```

```

<p>

    <input type="radio" name="kelamin" id="kelamin" value="PR" checked="checked"
/>
PEREMPUAN</p>
<p>HOBI:</p>
<p>
    <input type="checkbox" name="checkbox1" id="checkbox1" Value="Sepak Bola"/>
SEPAK BOLA</p>
<p>
    <input type="checkbox" name="checkbox2" id="checkbox2" Value="Mancing"/>
MANCING
</p>
<p>
    <input type="checkbox" name="checkbox3" id="checkbox3" Value="Olah Raga"/>
OLAH RAGA
</p>
<p>
<input type="submit" name="simpan" id="simpan" value="SIMPAN" />
    <input type="reset" name="batal" id="batal" value="BATAL" />
</p>
</form>

<?php
if (isset($_POST['simpan'])) {
    echo "<H2>Form setelah disubmit</H2>";
    echo "<b>Biodata:</b> <br>";
    echo "Nama: $NAMA ($NIM)<br>";
    echo "Password: $katakunci <br>";
    echo "Jenis Kelamin: $kelamin <br>";
    echo "Hobi:<br>";
    echo " =>".$checkbox1." <br>";
    echo " =>".$checkbox2." <br>";
    echo " =>".$checkbox3." <br>";
    echo "Pesan Anda: $pesan";
}
?>

</body>

```

BIODATA

NIM

NAMA

PASSWORD ANDA

JENIS KELAMIN:

LAKI-LAKI

PEREMPUAN

HOBI:

SEPAK BOLA

MANCING

OLAH RAGA

Gambar 4.2. Form Dengan Nama Variabel

Apabila tombol SIMPAN disubmit, maka *form* akan melakukan *action* ke perintah selanjutnya yaitu SIMPAN (`if (isset($_POST['simpan'])) { }`), hasilnya akan seperti di bawah ini:

BIODATA

NIM

NAMA

PASSWORD ANDA

JENIS KELAMIN:

LAKI-LAKI

PEREMPUAN

HOBI:

SEPAK BOLA

MANCING

OLAH RAGA

Form setelah disubmit

Biodata:
 Nama: HIBATEN WAFIROH (201212046)
 Password: 123456
 Jenis Kelamin: PR
 Hobi:
 =>Sepak Bola
 =>
 =>Olah Raga

Gambar 4.3. Hasil Variabel *Form* Setelah Disubmit

Dari program diatas, dapat terlihat bahwa atribut NAME yang terdapat pada *form* secara otomatis diubah oleh PHP ke dalam variabel-variabel, sehingga memudahkan untuk mengidentifikasi data yang dikirim oleh *form*. Penggunaan variabel ini lebih terasa gunanya jika data yang disubmit itu akan disimpan ke dalam varibel memori atau dengan perintah MySQL dapat disimpan dalam suatu tabel (database).

c. Predefined Variabel

PHP memiliki variabel-variabel yang telah didefinisikan sebelumnya, sehingga tidak dapat membuat variabel yang sama. *Predefined Variabel* yang sudah ditentukan, diantaranya: \$GLOBALS , \$_SERVER , \$_GET , \$_POST , \$_FILES , \$_COOKIE , \$_SESSION , \$_REQUEST , \$_ENV, \$php_errormsg, \$HTTP_RAW_POST_DATA, \$http_response_header, \$argc, \$argv, \$this.

Predefined Variable nilainya dapat bergantung pada beberapa faktor, seperti jenis *server*, *setting server*, dan lain-lain. *Predefine variable* dapat dilihat dengan memakai fungsi phpinfo(). Berikut ini adalah beberapa *predefine variable* yang dibuat pada *Web Server Apache*:

- a. REMOTE_ADDR: IP address dari komputer yang dipakeai untuk mengakses.
- b. HTTP_USER_AGENT: Keterangan jenis browser yang dipakai untuk mengakses, misal: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
- c. SERVER_NAME: Nama dari *server host* dimana *script diexecute*. Misal: localhost.
- d. SERVER_SOFTWARE: Menunjukkan *web server* yang digunakan, contoh: APACHE/1.3.12.

Berikut ini adalah contoh penggunaan *predefine variable*.

```
<html>

<body>

<?php
//file predefined.php
echo "<h2>Varibel Predefine: </h2>";
echo "Alamat IP: ",$REMOTE_ADDR;
```

```

echo "<br>Browser: ", $HTTP_USER_AGENT;

echo "<br>Nama Server: ",$SERVER_NAME;
echo "<br>Server Software: ",$SERVER_SOFTWARE;

?>

</body>
</html>

```



Gambar 4.4. Tampilan *Predefine Variable*

4.5.2. Konstanta

Konstanta adalah nilai yang tak pernah berubah. Misalnya selama aktif kuliah Nama Mahasiswa dan NIM Mahasiswa tidak berubah. Jika mencetak NIM dan Nama Mahasiswa pada seluruh halaman web, daripada mencetak string NIM dan Nama Mahasiswa lebih baik mendefinisikan konstanta, misal konstanta NAMA dan NIM.

Tidak seperti variabel, konstanta tidak didahului tanda \$. Konstanta dapat berisi string atau angka. Untuk membuat sebuah konstanta dalam PHP, dapat menggunakan fungsi *built-in* PHP yaitu `define()`. Contoh:

```

define("NIM","201112002");
define("NAMA","Fatimah Azahrah");
define("PTS", "STITEK Bontang");
define("ALAMAT", "Jl. Pupuk Raya No.28");

```

Dalam contoh diatas, membuat empat buah konstanta yaitu NIM, NAMA, PTS dan ALAMAT yang berisi identitas mahasiswa. Menampilkan konstanta tersebut, dapat dilihat perintah-perintah dibawah ini:

```

<?php

//konstanta.php
define("NIM","201112002");

```

```

define("NAMA","Fatimah Azahrah");

define("PTS", "STITEK Bontang");
define("ALAMAT", "Jl. Pupuk Raya No.28");
echo PTS."<br>";
echo "NIM      :". NIM ."<br>";
echo "NAMA    :". NAMA ."<br>";
echo "Alamat: ". ALAMAT ."<br>";
?>

```

Jika dijalankan perintah tersebut, maka secara otomatis akan didapatkan hasil:



Gambar 4.5. Tampilan Konstanta

4.6. Operator

Operator digunakan untuk menghasilkan sebuah nilai dengan melakukan operasi pada satu atau lebih nilai. Nilai yang dioperasikan (digunakan dalam operasi) disebut operand. PHP mengenal beberapa jenis operator, yang kebanyakan hampir sama dengan operator C/C++. Berikut ini akan dijabarkan satu persatu operator yang umum dipakai.

4.6.1. Operator Aritmatika

Hampir disetiap bahasa pemrograman selalu mengenal operator aritmatika. Operator aritmatika yang dikenal di PHP terdapat pada tabel berikut:

Tabel 4.1. Operator Aritmatika

Operator	Operasi
+	Penambahan
-	Pengurangan
*	Perkalian
/	Pembagian

%	Modulus
++	Inkrementasi
--	Dekrementasi

Catatan : PHP mentolerir adanya *white space* antara operator dan operand. Jadi, statement `9+2` sama saja dengan `9 + 2`.

Contoh:

```
<html>
<head>
<title>Operator ARITMATIKA</title>
</head>

<body>
<?php
    $banyak = 2.5;
    $harga = 8500;
    print("Pembelian bensin saat ini $banyak liter <br>\n");
    print("Total Bayar: Rp. ");
    print(number_format($banyak * $harga,0));
?>

</body>
</html>
```



Gambar 4.6. Tampilan Operator Aritmatika

4.6.2. Operator String

Hanya ada satu operator string, yaitu operator concatenation (`.`).

Operator ini digunakan untuk menyambung dua buah ekspresi string.

```
<?php
//string.php
$a="Selamat Belajar, ";
$b= $a . " Bahasa Pemrograman PHP";
echo "$b <br>";
```

//\$b berisikan " Selamat Belajar, Bahasa Pemrograman PHP "

?>

4.6.3 Operator Logika

Operator logika atau operator perbandingan untuk menyusun ekspresi kondisional. Ekspresi yang menggunakan operator logika selalu menghasilkan nilai TRUE atau FALSE. Misal Nilai \$a=5 dan \$b=10, akan menghasilkan nilai 0/FALSE dan 1/TRUE.

Tabel 4.2. Operator Logika

Operator	Operasi	Ekspresi	Hasil
==	Sama dengan	\$a == \$b	False
!=	Tidak sama dengan	\$a != \$b	True
<	Lebih kecil	\$a < \$b	True
>	Lebih besar	\$a > \$b	False
<=	Lebih kecil sama dengan	\$a <= \$b	True
>=	Lebih besar sama dengan	\$a >= \$b	False
AND atau &&	Logika AND	\$a AND \$b atau \$a && \$b	True
OR atau	Logika OR	\$a OR \$b atau \$a \$b	True
XOR	Logika XOR	\$a XOR \$b	False
!	Logika NOT	!\$a atau !\$b	False

4.6.4. Operator Penugasan

Operator penugasan (*assignment*) digunakan sebagai jalan pintas dalam melakukan suatu operasi diantara dua operant. Biasanya operant pertama adalah variable dan hasilnya disimpan pada variable yang sama. Misal Nilai \$a=10 dan \$b=5, hasilnya dapat dilihat sebagai berikut:

Tabel 4.3. Operator Penugasan

Operator	Penggunaan	Sama Dengan	Hasil
+=	\$a += \$b	\$a = \$a + \$b	15
-=	\$a -= \$b	\$a = \$a - \$b	10
*=	\$a *= \$b	\$a = \$a * \$b	50
/=	\$a /= \$b	\$a = \$a / \$b	10
%=	\$a %= \$b	\$a = \$a % \$b	0
.=	\$a .= \$b	\$a = \$a . \$b	105
++	\$a++	\$a = \$a + 1	11
--	\$a--	\$a = \$a - 1	9

BAB V

STRUKTUR KONTROL

Deskripsi Singkat

Bab ini dijelaskan tentang struktur kontrol, yaitu kontrol keputusan, kontrol pengulangan dan kontrol perpindahan.

Relevansi

Membuat struktur kontrol untuk membuat kontrol dan keputusan jalannya program selanjutnya sesuai kebutuhan dalam mengendalikan jalannya eksekusi program.

Tujuan Instruksional Khusus

Mahasiswa mengerti tentang struktur kontrol If, If Else, Switch Case, While, Do While, For, Exit, Break, Continue dan Return.

Struktur kontrol merupakan elemen yang sangat penting dalam tiap bahasa pemrograman, dengan struktur kontrol tersebut dapat mengendalikan jalannya eksekusi program. Berikut ini dijabarkan beberapa elemen penting struktur kontrol yang sering digunakan dalam pemrograman web.

5.1. Kontrol Keputusan

5.1.1. Pernyataan IF

Pernyataan IF berfungsi untuk mengambil keputusan secara bersyarat. Dengan pernyataan kondisional IF, sebuah atau lebih pernyataan hanya akan dieksekusi ketika sebuah ekspresi yang dievaluasi bernilai TRUE.

```
if (kondisi) {  
    pernyataan  
    ...  
}
```

Contoh, apabila mahasiswa memiliki nilai dibawah 60, maka tidak lulus. Perintah programnya akan seperti ini:

```
<?php  
  
//pernyataanIF.php  
$nilai=49;
```

```

if ($nilai < 60) {

    echo "Nilai $nilai tidak lulus <br>";
}
?>

```

Hasilnya: Nilai 49 tidak lulus

Contoh lain:

```

<?php

//pernyataanIFNilaiHuruf.php
$NilaiHuruf="B";
if ($NilaiHuruf == "B") {
    $NilaiBobot = 3;
    echo "$NilaiHuruf, Nilai Bobot = $NilaiBobot <br>";
}
?>

```

Hasilnya: B, Nilai Bobot = 3

Ingat bahwa operator perbandingan (==) berbeda dengan operator penugasan (=). Kondisi (\$NilaiHuruf == "B") akan TRUE, jika (\$NilaiHuruf == "B ") bernilai FALSE karena ada spasi setelah "B ". Kondisi TRUE akan melaksanakan perintah diantara tanda kurung kurawal atau {}.

5.1.2. Pernyataan ELSE

Jika ekspresi bernilai FALSE, akan mengeksekusi setelah pernyataan ELSE. Sintaks:

```

If (kondisi) {
    Statement1;
}else{
    Statement2;
}

```

Contoh menentukan kelulusan mahasiswa, jika nilai mulai 60 sampai 100 dinyatakan lulus dan jika dibawah 60 tidak lulus. Perintah program sebagai berikut:

```

<?php

//PernyataanIF_ELSE.php
$nilai=55;
if ($nilai >= 60) {
    echo "Nilai $nilai, adalah Lulus <br>";
} else {

```

```

echo "Nilai $nilai, Tidak Lulus <br>";

}
?>

```

Hasilnya: Nilai 55, Tidak Lulus

5.1.3. Pernyataan ELSEIF

Dapat membuat sebuah rangkaian dari banyak pilihan dengan menggunakan pernyataan ELSEIF. Pernyataan ELSEIF merupakan kombinasi dari pernyataan ELSE dan IF. Dengan memberikan serangkaian kondisi, perintah program bisa memeriksanya sampai ditemukan salah satu yang bernilai TRUE.

Perintah selanjutnya menentukan Nilai Angka (0 sampai dengan 100) menjadi Nilai Huruf (A, B, C, D dan E) dengan pernyataan IF dan ELSE. Perintahnya sebagai berikut:

```

<?php
//PernyataanIF_ELSE2.php
$NilaiAngka=66;
if ($NilaiAngka <= 40) {
    $NilaiHuruf = "E";
} elseif ($NilaiAngka < 60) {
    $NilaiHuruf = "D";
} elseif ($NilaiAngka < 70) {
    $NilaiHuruf = "C";
} elseif ($NilaiAngka < 80) {
    $NilaiHuruf = "B";
} else {
    $NilaiHuruf = "A";
}
echo "Nilai $NilaiAngka, adalah Nilai Huruf = $NilaiHuruf <br>";
?>

```

Hasilnya: Nilai 66, adalah Nilai Huruf = C

5.1.4. Pernyataan Switch..Case

Pernyataan SWITCH ... CASE digunakan pada suatu kondisi yang memiliki banyak kemungkinan jawaban. Bentuk umum switch adalah:

```

switch ($variabel) {
    case nilai_1:

```

```

        perintah_1;
        break;
    case nilai_2:
        perintah_2;
        break;
    default:
        perintah_3;
        break;
}

```

Apabila \$variabel = nilai_1, maka jalankan perintah_1.

Apabila \$variabel = nilai_2, maka jalankan perintah_2.

Apabila \$variabel tidak sama nilai_1 dan nilai_2, maka jalankan perintah_3.

```

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

    <title>PERNYATAAN SWITCH CASE</title>
</head>

<body>
<form method="post">
<tr>
    <td>Makanan kesukaan Anda:</td>
    <td><select name="makanan">
        <option value = "a">Nasi Campur
        <option value = "b">Gado-Gado
        <option value = "c">Lalapan Ayam
        <option value = "d">Mie Goreng
    </select>
    </td>
</tr>
<tr>
    <td><input type="submit" name="proses" value="PROSES" /></td>
</tr>
</form>

<?php
//PernyataanSwitch_Case.php
if (isset($_POST['proses'])) {
    switch($makanan) {
        case "a" :
            echo "<P>Nasi Campur Rp. 15.000";
            break;
        case "b" :

```

```

echo "<P>Gado-Gado Rp. 10.000";

        break;
case "c" :
    echo "<P>Lalapan Ayam Rp. 18.000";
    break;
case "d" :
    echo "<P>Mie Goreng Rp. 10.000";
    break;
default :
    echo "<P>Makanan lainnya Rp. 12.000";
    break;
}
}
?>

</body>
</html>

```



Gambar 5.1. Pernyataan Switch Case

5.2. Kontrol Pengulangan

5.2.1. Pernyataan While

Pernyataan While, selama pernyataan benar atau terpenuhi akan melaksanakan secara berulang-ulang, akan berhenti apabila pernyataan tidak benar atau tidak terpenuhi. Standar penulisannya adalah:

```

while (kondisi) {
    pernyataan;
    ...
}

```

Pengulangan while akan memeriksa apakah kondisi bernilai TRUE atau FALSE. Jika TRUE, satu atau sekumpulan pernyataan di dalam kurung kurawal akan dieksekusi, setelah itu kondisi akan dievaluasi lagi. Jika hasil evaluasi

masih TRUE, semua pernyataan dalam kurung kurawal akan di eksekusi lagi dan seterusnya, hingga evaluasi terhadap kondisi bernilai FALSE. Contoh perintah program menampilkan bilangan 1 sampai 9:

```
<?php
//PernyataanWhile.php
$a=1;
while ($a<10)
{
    echo($a)."<br>";
    $a++;
}
?>
```

5.2.2. Pernyataan Do..While

Standar penulisannya adalah:

```
do
{
    pernyataan;
    ...
}
while ();
```

Do...while hampir sama dengan while. Do...while akan menjalankan pernyataan lebih dulu dalam kurung kurawal, jika hasil pemeriksaan kondisi bernilai TRUE, pengulangan diteruskan. Jika hasilnya FALSE, pengulangan berhenti. Do...while, akan menjalankan pernyataan paling tidak satu kali. Contoh:

```
<?php

//PernyataanDoWhile.php
$Nilai =2;
do {
    echo "$Nilai ";
    $Nilai=$Nilai+2;
} while ($Nilai<=10)
?>
```

Hasilnya: 2 4 6 8 10

5.2.3. Pernyataan For

For digunakan untuk melaksanakan pernyataan perulangan sesuai jumlahnya. Standar penulisannya adalah:

```
for (expr1; expr2; expr3) {
    pernyataan;
}
```

Contoh:

```
<?php
//PernyataanFor.php
for ($nilai = 0; $nilai <= 10; $nilai += 2){
    echo "<tr> <td> $Nilai </td><td>". $nilai ." </td></tr>";
}
?>
```

Hasilnya: 0 2 4 6 8 10

5.3. Kontrol Perpindahan

Pernyataan yang termasuk dalam Kontrol Perpindahan, yaitu: Exit, Break, Continue dan Return. Contohnya

```
<?php
//PernyataanFor.php
for ($nilai = 0; $nilai <= 10; $nilai += 2){
    if ($nilai == 6) {
        break; // dapat diganti pernyataan Exit atau Return
    }
    echo "<tr> <td> $Nilai </td><td>". $nilai ." </td></tr>";
}
?>
```

Hasilnya: 0 2 4

```
<?php
//PernyataanFor.php
for ($nilai = 0; $nilai <= 10; $nilai += 2){
    if ($nilai == 6) {
        continue; //Nilai 6 sebagai continue tidak ditampilkan
    }
    echo "<tr> <td> $Nilai </td><td>". $nilai ." </td></tr>";
}
?>
```

Hasilnya: 0 2 4 8 10

BAB VI ARRAY

Deskripsi Singkat

Bab ini dijelaskan tentang penggunaan array dan array multidimensi, disertai dengan cara membuat dan penerapannya.

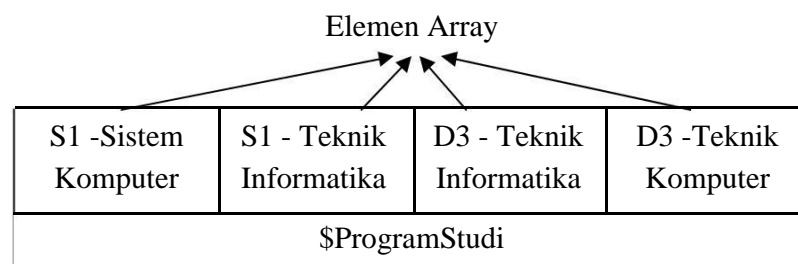
Relevansi

Pernyataan array memungkinkan penyimpanan data dalam suatu memori sementara dapat dihemat, penulisan perintah-perintah program dapat dihemat, perhitungan dalam matematik, rumus dan sebagainya dapat dihemat, dan berguna dalam pembuatan aplikasi yang besar.

Tujuan Instruksional Khusus

Mahasiswa mengerti tentang membuat dan menerapkan array dan array multidimensi sebagai penghematan memori dan penulisan perintah yang berulang-ulang.

Pernyataan Array dalam suatu variabel adalah salah satu aspek yang cukup penting untuk dibahas dalam pemrograman PHP. Array berupa variabel yang dapat menyimpan satu nilai, sekumpulan atau serentetan nilai. Contoh, variabel \$ProgramStudi memiliki sekumpulan data program studi yang ada.



Gambar 6.1. Variabel Array \$ProgramStudi

Satu variabel \$ProgramStudi memiliki 4 elemen data, setiap elemen berisi sebuah data. Elemen ke 0 berisi S1-Sistem Komputer, elemen ke 1 berisi S1-

Teknik Informatika, element ke 2 berisi D3-Teknik Informatika, dan element ke 3 berisi D3-Teknik Komputer.

6.1. Menciptakan Array

Suatu array dapat diciptakan dengan menggunakan standar penulisan array. Sebagai contoh dari variabel \$ProgramStudi diciptakan array dengan dua cara, yaitu:

```
$ProgramStudi=array("S1-Sistem Komputer", "S1-Teknik Informatika", "D3-
Teknik Informatika", "D3-Teknik Komputer");
```

Cara kedua dapat diciptakan secara berderet, yaitu:

```
$ProgramStudi[0] = "S1-Sistem Komputer";
$ProgramStudi[1] = "S1-Teknik Informatika";
$ProgramStudi[2] = "D3-Teknik Informatika";
$ProgramStudi[3] = "D3-Teknik Komputer";
```

6.2. Menampilkan Isi Array

Standar menampilkan/mengambil isi array, yaitu:

```
$variable[indeks]
```

Contoh:

```
echo $ProgramStudi[0];
```

Perintah programnya adalah:

```
<?php
//Array1.php
$ProgramStudi[0] = "S1-Sistem Komputer";
$ProgramStudi[1] = "S1-Teknik Informatika";
$ProgramStudi[2] = "D3-Teknik Informatika";
$ProgramStudi[3] = "D3-Teknik Komputer";
for ($no=0; $no <= 4; $no+=1) {
    echo "Elemen $no: $ProgramStudi[$no] <br>";
}
?>
```



Gambar 6.2. Menampilkan Variabel Array \$ProgramStudi

Sesuai variabel array \$ProgramStudi berisi empat elemen dari 0 sampai 3, sedangkan elemen ke lima (elemen 4) tidak berisi, maka hasilnya seperti gambar diatas.

6.3. Array Asosiatif

Sifat asosiatif, dapat mengasosiasikan setiap indeks/key sesuai dengan nilainya. Perintah inisialisasi array asosiatif, misal dengan nama \$ProgramStudi sebagai indeks/key dan Jumlah Mahasiswa sebagai nilainya.

```
$ProgramStudi = array("S1-Sistem Komputer"=>76, "S1-Teknik Informatika"
=>117, "D3-Teknik Informatika"=>152, "D3-Teknik Komputer"=>89);
```

Bagian sebelum tanda => disebut indeks atau key. Bagian setelah => disebut value. Variabel array asosiatif di atas dapat juga sama perintahnya dengan:

```
<?php
//Array2.php
$ProgramStudi["S1_Sistem_Komputer"] = 76;
$ProgramStudi["S1_Teknik_Informatika"] = 117;
$ProgramStudi["D3_Teknik_Informatika"] = 152;
$ProgramStudi["D3_Teknik_Komputer"] = 89;
echo "Jumlah Mahasiswa S1-Sistem Komputer =
$ProgramStudi[S1_Sistem_Komputer] <br>";
echo "Jumlah Mahasiswa S1-Teknik Informatika =
$ProgramStudi[S1_Teknik_Informatika] <br>";
echo "Jumlah Mahasiswa D3-Teknik Informatika =
$ProgramStudi[D3_Teknik_Informatika] <br>";
echo "Jumlah Mahasiswa D3-Teknik Komputer =
$ProgramStudi[D3_Teknik_Komputer] <br>";
```

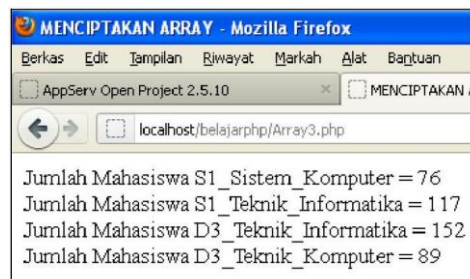
```
?>
```



Gambar 6.3. Menampilkan Variabel Array Asosiatif Dari \$ProgramStudi

Perintah program lainnya digunakan fungsi list() dan each(), yaitu:

```
<?php
//Array3.php
$ProgramStudi["S1_Sistem_Komputer"] = 76;
$ProgramStudi["S1_Teknik_Informatika"] = 117;
$ProgramStudi["D3_Teknik_Informatika"] = 152;
$ProgramStudi["D3_Teknik_Komputer"] = 89;
while (list($kunci,$nilai) = each($ProgramStudi)) {
    echo "Jumlah Mahasiswa $kunci = $nilai <br>";
}
?>
```



Gambar 6.4. Menampilkan Variabel Array Asosiatif Dengan List Dan Each

Pernyataan list() untuk memisahkan sebuah elemen array menjadi sejumlah nilai (\$kunci) dan nilainya(\$nilai), ((\$kunci,\$nilai)), sedangkan pernyataan each() untuk mengambil kunci (\$kunci) dan nilai (\$nilai) dari elemen array (\$ProgramStudi).

6.4. Array Multidimensi

Variabel Array selain menampung satu nilai, sekumpulan atau serentetan nilai, tetapi lokasi dalam array dapat menampung array yang lainnya. Hal ini akan dapat membuat array menjadi 2 dimensi. Sebuah array yang 2 dimensi dapat digunakan pada sebuah matrix yang memiliki baris dan kolom.

A = $\begin{Bmatrix} 3 & 6 & 5 \\ 4 & 9 & 5 \\ 2 & 7 & 2 \end{Bmatrix}$

```

//Array2D_1.php
$matrikA = array
(
    array(3,6,5),
    array(4,9,5),
    array(2,7,2)
);
echo $matrikA[0][0]." ".$matrikA[0][1]." ".$matrikA[0][2]." <br>";
echo $matrikA[1][0]." ".$matrikA[1][1]." ".$matrikA[1][2]." <br>";
echo $matrikA[2][0]." ".$matrikA[2][1]." ".$matrikA[2][2]." <br>";
?>

```

Cara lain menampilkan Matrik A terdiri 3 kolom dan 3 baris, adalah:

```

<?php
//Array2D_1_For.php
$matrikA = array
(
    array(3,6,5),
    array(4,9,5),
    array(2,7,2)
);
for ($baris = 0; $baris < 3; $baris++) {
    for ($kolom = 0; $kolom < 3; $kolom++) {
        echo $matrikA[$baris][$kolom]." ";
    }
    echo " <br>";
}
?>

```



Gambar 6.5. Menampilkan Array 2 Dimensi Dari Matrik

6.5. Mengurutkan Array

Pada array, nilai dari suatu elemen dapat diurutkan. Pernyataan untuk mengurutkan array diantaranya:

Tabel 5.1. Fungsi Mengurutkan Array

FUNGSI	KETERANGAN
<code>SORT</code>	Urut data array secara naik (ascending)
<code>ASORT</code>	Urut data array secara naik (ascending) dengan tetap mempertahankan indeksnya
<code>ARSSORT</code>	Urut data array secara turun (descending) dengan tetap mempertahankan indeksnya
<code>KSORT</code>	Urut data array berdasarkan indeksnya secara turun (descending)
<code>KRRSORT</code>	Urut data array berdasarkan indeksnya secara turun (ascending)
<code>SHUFFLE</code>	Mengacak data array

Apabila Matrik A tersebut diurutkan, maka hasilnya:

```
<?php
//Array2D_2_Sort.php
$matrikA = array
(
    array(3,6,5),
    array(4,9,5),
    array(2,7,2)
);

sort($matrikA);
echo "Hasil setelah diurut: <br>";
for ($baris = 0; $baris < 3; $baris++) {
    for ($kolom = 0; $kolom < 3; $kolom++) {
        echo $matrikA[$baris][$kolom]." ";
    }
    echo " <br>";
}
?>
```


BAB VII

STRING DAN FUNGSI

Deskripsi Singkat

Bab ini dijelaskan tentang String dan Fungsi yang sering digunakan dalam Bahasa Pemrograman PHP. String dan Fungsi dalam Bab VII ini, hanya sedikit yang dibahas dari banyaknya Pernyataan String dan Fungsi dalam Bahasa Pemrograman PHP, untuk selanjutnya String dan Fungsi dapat dibaca pada literatur lainnya.

Relevansi

String dan Fungsi dipergunakan untuk memberikan informasi dalam manipulasi string, angka maupun simbol yang lebih baik dan efisien pemrograman web dan sebelum disimpan dalam suatu database/tabel.

Tujuan Instruksional Khusus

Mahasiswa mengerti tentang manipulasi String dan pemanfaatan Fungsi yang sudah ada pada Bahasa Pemrograman PHP.

Penggunaan fungsi string untuk memformat dan memanipulasi teks, sedangkan pada ekspresi regular untuk mencari (dan mengganti) kata, frase atau pola lain yang ada di dalam string. Pada keadaan tertentu, akan ditemukan bagaimana memfilter data yang dimasukkan pemakai ke dalam database berasal dari string yang benar-benar valid. Dalam hal string perlu meningkatkan kemampuan memanipulasi string dan ekspresi reguler atas terhadap pemakai.

Fungsi-fungsi yang dapat digunakan untuk memformat ulang string, yaitu:

a. Fungsi trim()

Fungsi trim() untuk penghapusan karakter spasi putih (whitespace) yang berada di awal atau di belakang string. Fungsi ini berguna pada saat menyimpan string ke dalam database ataupun ke file. Contoh:

```
$nama=trim(" Siti Ambarwati ");  
$email=trim(" zuliatin_aries88@yahoo.co.id");
```

```
// hasilnya: $nama = Siti Ambarwati
```



```
// hasilnya: $email = zuliatin_aries88@yahoo.co.id
```

Yang termasuk spasi putih diantaranya:

“ ” : spasi

“\n” : baris baru

“\t” : tab horizontal

“\v” : tab vertikal

b. Fungsi nl2br()

Newline To Break atau fungsi nl2br(), berguna mengambil string sebagai parameter dan mengganti semua string newline(“\n”) dengan tag html
. Apabila string terlalu panjang, fungsi nl2br() digunakan untuk memotong dan tampilkan ke browser.

c. Mengubah huruf string: Huruf Kapital atau Huruf Kecil

Memformat ulang huruf string dari Huruf Kecil atau Huruf Kapital/Besar sesuai kebutuhan, dan disimpan lagi ke dalam suatu variable, database atau file. Misal, \$asal = Belajar bahasa php:

Tabel 7.1. Huruf Kapital Dan Kecil

Fungsi	Keterangan	Contoh	Hasil
strtoupper()	Mengubah semua huruf jadi kapital/besar	strtoupper(\$a)	BELAJAR BAHASA PHP
strtolower()	Mengubah semua huruf jadi kecil	strtolower(\$a)	belajar bahasa php
ucfirst()	Mengubah huruf awal jadi besar	ucfirst(\$a)	Belajar bahasa php
ucwords()	Mengubah huruf depan kata jadi besar	ucwords(\$a)	Belajar Bahasa Php

d. Stripslashes

Fungsi stripslashes() untuk menghilangkan garis miring tanda kutip di string. Misal

```
$a = stripslashes("http:\\Gado-Gado\Lontong ");
echo $a;
// hasil: http:Gado-GadoLontong
```

e. Explode dan Implode

Fungsi Explode() untuk memisahkan string menjadi array berdasarkan berdasarkan ketentuan. Perintahnya:

```
array explode(string pemisah, string input);
```

Contoh:

```
<?php
    $str = "Satu/Dua/Tiga/Empat/Lima";
    $hasil = explode("/", $str);
    $no = 0;
    while ($no <= 5) {
        echo $hasil[$no]. "<br>";
        $no++;
    }

    $hasil = explode("/", $str, 3);
    $no = 0;
    while ($no <= 5) {
        echo $hasil[$no]. "<br>";
        $no++;
    }
?>
```

Hasil :

Satu
Dua
Tiga
Empat
Lima

Satu
Dua
Tiga/Empat/Lima

Contoh:

```
<?php
    $str[0] = "Satu";
    $str[1] = "Dua";
    $str[2] = "Tiga";
    $str[3] = "Empat";
    $str[4] = "Lima";
    $hasil = implode("-", $str);
```

```

echo $hasil;

?>

```

Hasil :
 Satu-Dua-Tiga-Empat-Lima

f. strcmp dan strcasecmp

Untuk perbandingan dua buah string, digunakan fungsi strcmp() dan strcmpi(). Standar perintahnya:

```

int strcmp(string string1, string string2)
int strcmpi(string string1, string string2)

```

Fungsi strcmp(), membandingkan dua buah string benar-benar sama huruf besar dan kecilnya, sedangkan fungsi strcmpi() memberlakukan huruf kecil dan huruf kapital dianggap sama. Nilai balik (return) fungsi ini :

- ✓ <0 : Jika string1 lebih kecil dari string2
- ✓ ==0 : Jika string1 sama dengan string2 (equivalen dengan FALSE)
- ✓ >0 : jika string1 lebih besar daripada string2 (equivalen dengan TRUE)

contoh :

```

<?php
$a = "Belajar PHP";
$b = "belajar php";
if (strcmpi($a, $b)==0) {
    echo "$a sama dengan $b";
} else{
    echo "$a tidak sama dengan $b";
}
?>

```

Hasilnya:
 Belajar PHP sama dengan belajar php

g. strlen()

Menghitung panjang karakter suatu string dengan fungsi strlen(). Contoh:

```

echo strlen("Belajar Pemrograman PHP");

```

```

// hasil: 23

```

h. date()

Date() merupakan menyatakan waktu. Di dalamnya dapat disimpan format string yang menunjukkan tanggal maupun waktu. Di bawah ini format string menyatakan waktu:

- ✓ d - tanggal hari dalam setiap bulan, misal "01" to "31"
- ✓ D - nama hari yang dinyatakan dengan 3 huruf pertama; misal "Fri"
- ✓ F - nama bulan dalam format lengkap. misal "January"
- ✓ h - jam dalam format 12, misal "01" to "12"
- ✓ H - jam, dalam format 24, misal "00" to "23"
- ✓ i - menit; i.e. "00" to "59"
- ✓ j - tanggal hari dalam setiap bulan tanpa embel-embel nol, misal "1" to "31"
- ✓ m - nomor bulan dalam setahun; misal "01" to "12"
- ✓ M – nama bulan dalam format 3 karakter; misal "Jan"
- ✓ n – nomor bulan tanpa nol di depan, misal "1" to "12"
- ✓ s - detik; misal. "00" to "59"
- ✓ Y – tahun berformat 4 digits; misal "1999"
- ✓ y – year berformat 2 digits, misal "99"

Contoh: echo date("H:m:s d-M-Y");

Hasilnya: 11:12:28 03-Dec-2014

i. Require() dan Include()

Perintah program require() dan include() akan menyertakan sebuah file ke dalam file tersebut berada. Perbedaannya:

- ✓ Require(), file yang disertakan terjadi suatu *fatal error* (*E_COMPILE_ERROR*) dan stop menjalankan perintah selanjutnya.
- ✓ Include(), file yang disertakan terjadi suatu peringatan kesalahan/*warning* (*E_WARNING*) dan perintah selanjutnya tetap dijalankan.

Standar penulisannya:

```
require "nama_file";
include "nama_file";
```

Perintah program require_once() dan include_once(), file yang disertakan hanya sekali dieksekusi, walaupun file tersebut beberapa kali disertakan.

BAB VIII

DATABASE MYSQL

Deskripsi Singkat

Bab ini dijelaskan tentang peranan Database MySQL, perintah dasar SQL dan hubungan dengan Bahasa Pemrograman PHP serta penerapan dalam suatu aplikasi.

Relevansi

Database yang berada pada *web server* dan jaringan internet, salah satunya adalah Database MySQL banyak digunakan secara gratis tetapi bukan digunakan untuk komersial. Bahasa Pemrograman PHP dan MySQL serta mesin *server* dengan Apache mudah didapat dalam dunia internet secara gratis dan sudah jadi satu paket instalasi (AppServ berbagai versi).

Tujuan Instruksional Khusus

Mahasiswa mengerti tentang Database MySQL, perintah dasar SQL, koneksi Bahasa Pemrograman PHP dan MySQL, dan penerapannya dalam membuat aplikasi berbasis web.

Database MySQL adalah perangkat lunak sistem manajemen basis data SQL (Structured Query Language) atau DBMS (Database Management System) yang multithread dan multiuser. MySQL salah satu Relational Database Management System (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License), tetapi tidak boleh digunakan dalam bentuk komersial. Database MySQL dapat dikatakan lebih unggul dari database lainnya, dimana keunggulan tersebut diantaranya cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, sebagai *database server*, lebih cepat.



Gambar 8.1. Logo Database MySQL

SQL adalah bahasa database standar yang memudahkan menampilkan, penyimpanan, perubahan dan akses hapus record data. MySQL dikenal istilah database, table dan record. Database terdiri dari satu atau lebih table, sedangkan table adalah sebuah struktur data dua dimensi yang terdiri dari baris-baris record dan kolom.

DATABASE			
Table1			
Table2			
Tabel			
	Field Name 1	Field Name 2	Field Name
Record1			
Record2			
Record			

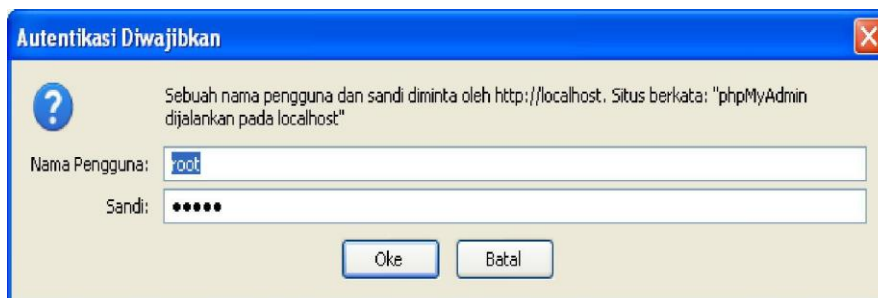
Gambar 8.2. Ilustrasi Database MySQL

Sebelum menggunakan MySQL, pemakai harus login terlebih dahulu.



Gambar 8.3. Layar Awal http://localhost

Pada gambar tersebut diatas, pilih “phpMyAdmin Database Manager Version 2.10.3”, akan menampilkan login untuk Database MySQL.



Gambar 8.4. Login Database MySQL

8.1. Membuat database

Standar perintah membuat database pada SQL:

```
CREATE DATABASE nama_database;
```

Setelah login berhasil, dihadapkan pada “Ciptakan database baru” dengan gambar dibawah.



Gambar 8.5. Menciptakan Database Buku

Cara lain dengan mengetik pada kotak dialog SQL, dengan mengetik:

```
CREATE DATABASE buku;
```

Menghapus database yang sudah dibuat dengan perintah:

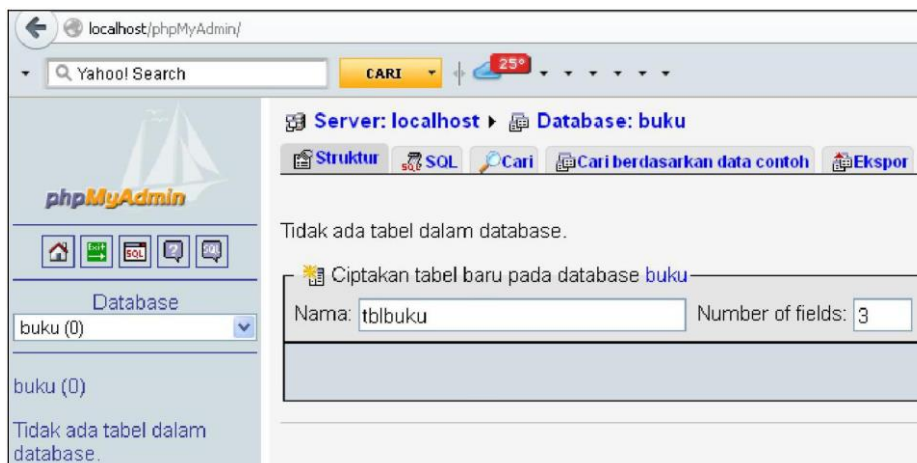
```
DROP DATABASE buku;
```



Gambar 8.6. Menciptakan Database Buku Pada Kotak Dialog SQL

8.2. Membuat Tabel

Dalam satu database terdiri dari beberapa tabel, untuk membuat tabel dapat digunakan kotak yang tersedia atau pada kotak dialog SQL. Berikut ini dibuat tabel dalam database buku, yaitu tblbuku dengan 3 nama field (Nama: tblbuku, Number of fields: 3), kemudian pilih Go.



Gambar 8.7. Menciptakan Tabel tblbuku Pada Kotak Dialog

Akan tersedia kotak dialog berikutnya dengan tiga baris sesuai kebutuhan.

- ✓ Pada kolom Field diisi nama field sesuai kebutuhan (b_kode, b_nama dan b_harga).
- ✓ Pada kolom Jenis diisi Varchar (untuk karakter) dan Int (Integer).
- ✓ Ukuran Panjang/Nilai, diisi sesuai kebutuhan dengan maksimal panjang karakter yang digunakan untuk memasukkan data.
- ✓ Pada gambar kunci, sebagai primary key dari suatu tabel tblbuku, yaitu hanya b_kode. Nama field b_kode sebagai primary key karena isi datanya tidak mungkin sama dengan data lain, contoh lain Nomor Karyawan, Nomor Mahasiswa, Nomor Pegawai, dan lainnya.

Field	Jenis ?	Ukuran Panjang/Nilai*1
tb_kode	VARCHAR	13
tb_nama	VARCHAR	30
tb_harga	INT	9

Komentar tabel:

Mes

My

Gambar 8.8. Menciptakan Tabel tblbuku Dan Primary Key

Atau dapat diketik pada kotak dialog SQL, yaitu:

```
CREATE TABLE `tblbuku` (
  `tb_kode` VARCHAR( 13 ) NOT NULL ,
  `tb_nama` VARCHAR( 30 ) NOT NULL ,
  `tb_harga` INT( 9 ) NOT NULL ,
  PRIMARY KEY ( `tb_kode` )
) ENGINE = MYISAM ;
```

Hasilnya:

	Field	Jenis	Penyortiran	Atribut	Kosong
<input type="checkbox"/>	tb_kode	varchar(13)	utf8_general_ci		Tidak
<input type="checkbox"/>	tb_nama	varchar(30)	utf8_general_ci		Tidak
<input type="checkbox"/>	tb_harga	int(9)			Tidak

[Pilih semua](#) / [Balik pilihan yang ditandai:](#)

[Pandangan cetak](#) [Menganalisa struktur tabel](#)

Menambahkan field(s) pada akhir tabel

Indeks:				
Nama Kunci	Jenis	Bilangan Pokok	Aksi	Field
PRIMARY	PRIMARY	0		tb_kode

Ciptakan indeks pada kolom

Gambar 8.9. Hasil Tabel tblbuku Dan Primary Key

8.3. Perintah Dasar SQL

a. Select

Menampilkan isi tabel. Perintah dasar:

```
SELECT columns FROM tablename WHERE condition(s)
```

b. Insert

Menambah record pada suatu tabel. Perintah dasar:

- ✓ `INSERT INTO table-name VALUES(value1, value2, ...)`
- ✓ `INSERT INTO table-name (field1, field2,)
VALUES (value1, value2, ...)`
- ✓ `INSERT INTO table-name SET (field1=value1, field2=
value2,)`

c. Update

Merubah isi field dalam suatu record. Perintah dasar:

```
UPDATE nama_tabel SET field1=value1, field2=value2,....  
WHERE condition(s)
```

d. Delete

Menghapus satu atau beberapa record dalam suatu tabel. Perintah dasar:

```
DELETE FROM table-name WHERE condition(s)
```

8.4. Membangun Suatu Aplikasi PHP Dan MySQL

Membangun suatu aplikasi dapat dimulai dari hal yang kecil, tetapi untuk mengembangkan akan mudah karena dasar pemahaman pembuatan aplikasi sudah dimiliki. Aplikasi yang dibangun berbasis web ada tujuh kategori yang perlu dipelajari yaitu:

- a. Koneksi ke database
- b. Menampilkan Data
- c. Menambah Data
- d. Mengubah Data
- e. Menghapus Data
- f. Mencetak/Laporan Detail
- g. Penutupan koneksi

a. Membuat Koneksi Ke Database

Pernyataan `mysql_connect()` digunakan untuk membuat koneksi ke MySQL. Perintah dasarnya adalah:

```
mysql_connect(host, user, password);
```

Fungsi ini akan mengkoneksikan ke database mysql berada. Biasanya nama host-nya adalah localhost, user dan password sama dengan pada saat masuk "phpMyAdmin Database Manager Version 2.10.3" pada Gambar 8.4 sebelumnya.

Kemudian password adalah password dari user tersebut. Nilai kembalian ini adalah integer positif (true) yakni apabila proses koneksi berhasil. Jika proses koneksi gagal, maka akan memberikan nol (false).

Fungsi `mysql_select_db()` digunakan untuk memilih database yang aktif. Prototipenya:

```
mysql_select_db(string nama_database);
```

Contoh file Buku_Koneksi.php:

```
<?
    $koneksi = mysql_connect("localhost","root","junai");
    mysql_select_db("buku",$koneksi);
?>
```

b. Menampilkan Tabel

Sebelum menampilkan data, fungsi yang perlu diketahui adalah fungsi `mysql_query` untuk mengeksekusi query SQL, standar perintahnya adalah:

```
mysql_query(string query);
```

Fungsi `mysql_num_rows($query)` yang berfungsi untuk menghitung jumlah baris hasil query.

Fungsi `mysql_fetch_array($query)` atau `mysql_fetch_object($query)`. `mysql_fetch_array` dan `mysql_fetch_object` memiliki fungsi yang sama, yakni mengambil hasil query SQL. Perbedaannya, `mysql_fetch_array` outputnya berupa array dengan nama field; sedangkan `mysql_fetch_object` outputnya berupa object, penulis lebih senang menggunakan `mysql_fetch_array`. Begitu pula dengan `mysql_fetch_row($query)` yang mana menggunakan nomor indeks yang mana penulis harus mengingat nomor indeks nama fieldnya.

Setelah dieksekusi (`mysql_query`), diperlukan untuk menampung data, penulis lebih senang menggunakan perintah:

```
mysql_fetch_array(mysql_query(string query));
```

Secara lengkap perintah menampilkan data dengan fungsi `SELECT columns`

`FROM tablename` adalah:

```
<html>
<body>
<table width="623" border="1">
<tr>
    <td width="5%">NO</td>
    <td width="7%">KODE BUKU</td>
    <td width="30%">NAMA BUKU</td>
    <td width="10%">HARGA</td>
    <td width="10%">AKSI</td>
```

```

</tr>

<?php
$koneksi = mysql_connect("localhost","root","junai");
mysql_select_db("buku",$koneksi);
$no=1;
$hasil = mysql_query("SELECT * FROM tblbuku");
while ($stampil=mysql_fetch_array($hasil)) { ?>

    <tr>
        <td align="right"><?=$no?></td>
        <td><?=$stampil['tb_kode']?></td>
        <td><?=$stampil['tb_nama']?></td>
        <td align="right"><?=number_format($stampil['tb_harga'],0)?></td>
        <td><a href="Buku_Ubah.php?kode=<?=$stampil['tb_kode']?>">Ubah</a>
            <a
href="Buku_Hapus.php?kode=<?=$stampil['tb_kode']?>">Hapus</a></td>
    </tr>
<?php
    $no++;
}
?>
</table>
</body>
</html>

```

NO	KODE BUKU	NAMA BUKU	HARGA	AKSI
1	6120487439849	PEMROGRAMAN WEB PHP-MYSQL	57,000	Ubah Hapus
2	6207484372947	BELAJAR PEMROGRAMAN VB 2014	75,500	Ubah Hapus

Gambar 8.10. Tampil Data tblbuku

c. Menambah Data

Sebelum menampilkan data, masih diperlukan lagi fungsi `mysql_query`. Fungsi yang digunakan untuk menambah data: `INSERT INTO table-name SET (field1=value1, field2= value2,)`, secara lengkap perintah program adalah:

Kode	<input type="text" value="6207484372947"/>
Nama	<input type="text" value="BELAJAR.JSP"/>
Harga	<input type="text" value="42500"/>
	<input type="submit" value="Simpan"/>

Gambar 8.11. Tambah Data tblbuku

```

<html>

<body>
<form name="form1" action="" method="post">
  <dl>
    <dt>Kode</dt>
    <dd><input type="text" name="kode"/></dd>
    <dt>Nama</dt>
    <dd><input type="text" name="nama"/></dd>
    <dt>Harga</dt>
    <dd><input type="text" name="harga"/></dd>
    <dt></dt>
    <dd><input type="submit" value="Simpan"/></dd>
  </dl>
</form>
<?php
include "Buku_Koneksi.php";
if($_POST) {
  $sql = "INSERT INTO tblbuku SET
          tb_kode = '$_POST[kode]',
          tb_nama = '$_POST[nama]',
          tb_harga = '$_POST[harga]'";
  mysql_query($sql, $koneksi)
    or die ("Gagal Perintah SQL". mysql_error());
  echo 'data tersimpan';
}
?>
</body>
</html>

```

d. Mengubah Data

Sebelum menghapus data, juga memerlukan lagi fungsi `mysql_query`.

Fungsi yang digunakan untuk menghapus data, yaitu: `DELETE FROM table-name WHERE condition(s)`, secara lengkap perintah program adalah:

```
<html>
<body>
<?php
include "Buku_Koneksi.php";
$kode = $_GET[kode];
if($kode<>"") {
    $sql = mysql_query("SELECT * FROM tblbuku WHERE tb_kode ='$kode'");
    $hasil = mysql_fetch_array($sql);
}
?>
<form name="form1" action="" method="post">
  <dl>
    <dt>Kode</dt>
    <dd><?=$hasil[tb_kode]?></dd>
    <dt>Nama</dt>
    <dd><input type="text" name="nama" value="<?=$hasil[tb_nama]?>"
size="40"/></dd>
    <dt>Harga</dt>
    <dd><input type="text" name="harga"
value="<?=$hasil[tb_harga]?>" /></dd>
    <dt>
    <dd><input type="hidden" name="hkode"
value="<?=$hasil[tb_kode]?>"></dd>
    </dt>
    <dd><input type="submit" value="Simpan"/></dd>
  </dl>
</form>
<?php
if($_POST) {
    $sql = "UPDATE tblbuku SET
            tb_nama = '$_POST[nama]',
            tb_harga = '$_POST[harga]'
            WHERE tb_kode = '$_POST[hkode]'";
    mysql_query($sql, $koneksi)
        or die ("Gagal Perintah SQL". mysql_error());
    echo 'Data telah diubah <br>';
    echo "<a href='Buku_Tampil.php'>Tampil Buku</a>"; }
?>
</body>
</html>
```

localhost/belajarphp/Buku_Ubah.php?kode=6120487439849

Yahoo! Search

Kode
6120487439849

Nama

Harga

Data telah diubah
[Tampil Buku](#)

Gambar 8.12. Ubah Data tblbuku

e. Menghapus Data

Sebelum menghapus data, juga memerlukan lagi fungsi `mysql_query`. Fungsi yang digunakan untuk menghapus data, yaitu: `DELETE FROM table-name WHERE condition(s)`, secara lengkap perintah program adalah:

```
<html>
<body>
<?php
include "Buku_Koneksi.php";
$kode = $_GET[kode];
if($kode<>"") {
    $sql = "DELETE FROM tblbuku WHERE tb_kode = '$kode'";
    mysql_query($sql)
        or die ("Gagal Perintah SQL". mysql_error());
    echo "Data dihapus";
    echo "<a href='Buku_Tampil.php'>Tampil Buku</a>";
}
?>
</body>
</html>
```


DAFTAR PUSTAKA

- Ralph Moseley, 2011. *Developing Web Applications*. India: Wiley India Pvt. Limited
- Janner Simamarta, 2010. *Rekayasa Web*. Yogyakarta: Andi
- Lynch, Patrick J., Sarah Horton. 2009. *Web Style Guide Basic Design Principles for Creating Web Sites 3rd Edition*. New Haven & London: Yale University Press.
- Wang, Paul, Sanda Katila. 2002. *An Introduction to Web Design + Programming*. Course Technology/Cengage Learning.
- Sherry Bishop, 2002. *Macromedia Web: Flash MX, Dreamweaver MX, and Fireworks MX Professional Projects*. Boston: Cengage Learning
- Junaini, 2015. *Pemrograman WEB dengan bahasa HTML, PHP, dan Database MySQL Edisi Ke Satu*. Universitas Nahdlatul Ulama Samarinda, Kalimantan Timur.
- <http://www.w3schools.com/php>
- <http://php.net>
- <http://www.mysql.com>

