

Menjadi WebMaster Dalam 30 Hari

By Rio Astamal

© Copyright 2005-2006 www.rahasia-webmaster.com ™

Hak Cipta Menjadi WebMaster Dalam 30 Hari

E-book Menjadi Webmaster Dalam 30 Hari merupakan hasil kerja keras penulis. Dilarang keras untuk untuk menggandakan, mereproduksi, atau menyebarluaskan e-book ini tanpa ijin tertulis dari saya Rio Astamal selaku penulis.

E-book ini hanya dipublikasikan secara resmi melalui situs www.rahasia-webmaster.com. Semua materi dan gambar yang terdapat pada e-book ini merupakan hak cipta dari penulis.

Anda dapat memperoleh penghasilan instan dengan mengikuti program reseller kami. Setiap member baru yang masuk lewat rekomendasi anda. Maka secara otomatis anda akan memperoleh komisi 50% dari harga produk kami.

Penting!, jika anda menemukan situs atau orang yang menawarkan, mereproduksi dan menjual e-book ini tanpa melalui sistem reseller kami. Saya mohon dengan sangat kepada anda untuk melaporkannya kepada saya. Akan ada **bonus spesial** dari saya untuk anda atas kesediaan anda melaporkan hal tersebut.

Kirim laporan anda ke email saya di admin@rahasia-webmaster.com.

© Copyright 2005-2006 www.rahasia-webmaster.com

Hak Cipta Dilindungi oleh Undang-undang

Alamat Download Materi E-book

Semua materi dan source code yang dibahas pada e-book ini dapat anda download di alamat:

<http://materi.rahasia-webmaster.com/>

Tentang Materi

Semua source kode yang ada pada e-book ini telah saya tes dan bekerja. Akan tetapi mungkin dalam penyalinan ke e-book ini ada suatu kesalahan yang menyebabkan script tersebut tidak bekerja.

Oleh karena itu, jika dalam e-book ini ada source code yang error atau tidak bisa jalan. Saya mohon anda untuk segera memberitahukan kepada saya error tersebut agar dapat segera saya update.

Laporkan error tersebut ke bug@rahasia-webmaster.com.

Ketika anda mengirimkan e-mail laporan, mohon cantumkan file pada hari dan bab keberapa yang terdapat error didalamnya. Terima Kasih.

Merek dagang

Semua merek dagang yang disebutkan dalam e-book ini merupakan Hak Cipta dari masing-masing vendor pembuatnya.

Microsoft Windows, Windows 98, Windows 2000, Windows XP, MS-DOS, Microsoft Notepad, Internet Explorer merupakan merek dagang dari Microsoft Corp.

Adobe, Adobe Acrobat, Adobe Acrobat reader, Adobe Photosop merupakan merek dagang dari Adobe Systems Inc.

AceFTP, AceFTP 3 Freeware, AceFTP 3 Pro merupakan merek dagang dari Visicom Media Inc.

Special Thanks To:

"I say a lot of thanks to everyone that have supported my hard work. Especially to all my closest friends for their support to me. Thanks guys."

Kata Pengantar

Puji syukur saya panjatkan kehadirat Allah SWT. Karena hanya dengan rahmat dan ridhoNya saya dapat menyelesaikan e-book "Menjadi Webmaster Dalam 30 Hari". Saya ucapan terima kasih kepada orang-orang yang disekitar saya atas support yang telah mereka berikan.

Dalam e-book ini anda akan mengetahui bagaimana menjadi seorang webmaster. Dan cara membuat website yang bisa "menjual" informasi yang anda buat sehingga dapat menghasilkan profit untuk anda.

Dalam e-book ini software-software yang akan kita gunakan sifatnya Free dan Open source. Diantaranya PHP, MySQL, Apache dan phpMyAdmin. Alasan kita menggunakan software-software tersebut sudah jelas karena selain handal software tersebut bersifat gratis.

Selain keempat software yang telah disebutkan, ada beberapa software tambahan yang sifatnya opsional. Artinya anda dapat install atau tidak. Namun jika anda benar-benar ingin merubah komputer anda menjadi server virtual saya sarankan untuk menginstall program seperti Mail Server (hMailServer).

Cara pandang buku ini sedikit berbeda dengan buku lainnya. Dimana kebanyakan buku hanya mengajarkan teori saja tanpa penerapan contoh yang nyata. Sehingga pembaca kurang bisa mengaplikasikan ilmunya setelah membaca. Untuk itu pada buku ini anda akan diajarkan teori terlebih dahulu lalu anda akan dihadapkan pada sebuah masalah yaitu membangun sebuah proyek website yang memiliki komponen cukup lengkap. Seperti sistem member, sistem reseller, admin area dan masih banyak lainnya.

E-book ini disusun bab demi bab sesuai dengan tingkat kesulitan dan terdapat sub bab dari mulai hari ke 1 hingga hari ke 30. Sehingga tepat pada hari ke 30 anda diharapkan sudah dapat menguasai semua materi pada e-book ini.

Pada bab 1 akan dibahas dasar-dasar HTML dan PHP. Pada bab 2 anda akan diajarkan teknik PHP lebih lanjut yaitu fungsi dan pemrograman berorientasi objek. Bab 3 akan membahas penggunaan MySQL dan phpMyAdmin. Keamanan dan contoh-contoh aplikasi dikupas pada Bab 4. Pada Bab 5 anda akan diajarkan bagaimana membuat website yang "menjual" dan contoh websitenya yaitu menggunakan sistem reseller. Pada bab terakhir yaitu Bab 6 membahas bagaimana mengupload file kita ke server sekaligus mengkonfigurasi website kita pada server.

Saya harap dengan hadirnya e-book ini dapat memajukan pengetahuan masyarakat Indonesia khususnya dalam bidang Teknologi Informasi (TI).

Surabaya, Desember 2005

Penulis

Daftar Isi

Kata Pengantar.....	iv
Daftar Isi.....	v
Pendahuluan.....	ix
Bab 1 HTML dan PHP	1
Hari ke-1	2
HTML.....	2
Menggunakan HTML.....	2
Manipulasi Font dan Link.....	3
Membuat Tabel.....	5
CSS (Cascading Style Sheet).....	6
Form HTML.....	7
Hari ke-2	12
PHP.....	12
Variabel Pada PHP.....	12
Jangkauan (Scope) Variabel.....	13
Komentar Pada PHP.....	15
Struktur Kontrol(if...else...).....	16
Operator.....	21
Hari ke-3	24
Operator Kombinasi.....	24
Operator Perbandingan.....	24
Operator Logika.....	25
Struktur Kontrol (if...elseif, switch...).....	25
Perulangan (for..., foreach..., while..., do...while).....	28
Hari ke-4	34
File dan Array.....	34
File.....	34
Menyimpan File.....	34
Membuka File.....	34
Menulis File.....	36
Menutup File.....	36
Membaca dari File.....	36
Menghapus File.....	39
Array.....	39
Array dengan Index Numerik.....	40
Looping dalam Array Index Numerik.....	40
Array dengan Index String.....	42
Looping dalam Array Index String.....	43
Array Multidimensi.....	46
Array 2D.....	46
Array 3D.....	48
Fungsi-fungsi pada Array.....	49
Membaca Array dari File.....	51
Hari ke-5	55
Manipulasi String dan REGEX.....	55
Memformat String.....	55
Mengubah Format Huruf pada String.....	58
Format Angka.....	61
Mencari String.....	62
Mengganti String.....	63
Regular Expression (REGEX).....	63

REGEX untuk Validasi Email.....	65
Mencari String dengan REGEX.....	65
Mengganti String dengan REGEX.....	66
Menggunakan Ulang Kode.....	66
Menggunakan require() dan include().....	67
Hari ke-6	70
Proyek Aplikasi Bab 1 – Minimarket Online.....	70
Bab 2 – Fungsi, OOP, dan Teknik PHP Lainnya.....	86
Hari ke-7	87
Fungsi.....	87
Definisi Fungsi.....	87
Memanggil Fungsi.....	87
Penamaan Fungsi.....	88
Mengapa Harus Membuat Fungsi Buatan Sendiri.....	89
Scope(Jangkauan).....	90
Menghentikan Eksekusi Fungsi.....	91
Mengembalikan Nilai dari Fungsi.....	92
Penerapan Teori Fungsi.....	93
OOP (Object Oriented Programming).....	94
Konsep OOP.....	95
Membuat Class.....	95
Variabel \$this.....	96
Properti private, public dan protected.....	96
Memanggil Class.....	97
Inheritance(Pewarisan).....	97
Penerapan Teori Class.....	98
Hari ke-8	103
Teknik PHP Lainnya.....	103
Tanggal dan Waktu.....	103
Session Pada PHP.....	108
Cookie.....	108
PHP Session.....	109
Hari ke-9	114
Teknik PHP Lainnya(2).....	114
Menggunakan \$_GET.....	114
Upload File.....	115
Direktori dan File.....	118
Hari ke-10	122
Proyek Aplikasi Bab 2 – Java Online News.....	122
Bab 3 – MySQL, phpMyAdmin, dan Mengakses MySQL Menggunakan PHP.....	144
Hari ke-11	144
MySQL.....	144
Menggunakan MySQL.....	145
Membuat Database.....	146
Mengaktifkan Database.....	147
Membuat Tabel.....	149
Memasukkan Data ke Database.....	149
Mendapatkan Data dari Database.....	151
Mengupdate Record pada Database.....	153
Meng-Alter Tabel.....	153
Menghapus Record pada Database.....	155

Menghapus Tabel.....	155
Menghapus Database.....	155
Teknik MySQL Lainnya.....	156
User Privilige.....	159
Hari ke-12	162
phpMyAdmin.....	162
Mengenal phpMyAdmin.....	162
Mengenal Berbagai Tab phpMyAdmin.....	163
Mengakses MySQL Menggunakan PHP.....	168
Langkah-langkah Mengakses Database Lewat PHP.....	168
Melakukan Koneksi ke MySQL.....	168
Memilih Database.....	169
Melakukan Query pada Database.....	169
Menampilkan Hasil Query.....	169
Menutup Koneksi.....	170
Fungsi-fungsi Lain.....	170
Penerapan Teori Mengakses MySQL Menggunakan PHP.....	171
Hari ke-13 dan 14	176
Proyek Aplikasi Bab 3 – Java Kids Foundation.....	176
Bab 4 – Masalah Keamanan, Error Handling, dan Contoh-contoh Aplikasi	218
Hari ke-15	219
Masalah Keamanan.....	219
Seberapa Penting Informasi Anda?.....	219
Ancaman Keamanan.....	219
Enkripsi Data.....	222
Secure Login.....	224
Firewall.....	228
Error Handling.....	229
Programming Error.....	229
Syntax Error.....	229
Run Time Error.....	230
Logic Error.....	232
Hari ke-16 dan 17	235
Contoh-contoh Aplikasi.....	235
NgawurPolling.localhost.....	235
Hari ke-18,19 dan 20	277
ForumBareng.localhost.....	277
Hari ke-21, 22, 23, dan 24	340
SuperIklan.localhost.....	340
Bab 5 – Memulai Bisnis di Internet, Web Design dan Membuat Website Reseller	415
Hari ke-25	416
Memulai Bisnis di Internet.....	416
Menciptakan Produk untuk Dijual.....	416
Membuat File E-book (PDF).....	419
Mengotomatiskan Bisnis Anda.....	419
Meta Tags.....	430
Strategi Promosi.....	431
Web Design.....	433

Hari ke-26, 27, 28, dan 29	439
LadangBisnis.localhost(website reseller).....	439
Bab 6 – Upload File, Konfigurasi Website, dan Domain dan Web Hosting.....	536
Hari ke-30	537
Upload File.....	537
Pengertian Upload.....	537
Software FTP.....	537
Memilih Web Hosting Gratisan.....	539
Trik Menyiasati Nama Domain.....	542
Konfigurasi Website.....	547
Mengupload File.....	554
Domain dan Web Hosting.....	557
Penutup.....	559

Pendahuluan

Untuk Siapa E-book Ini?

Buku ini ditujukan bagi anda yang belum pernah mengenal programming atau anda yang sudah mengenal programming namun ingin lebih meningkatkan kemampuan anda. Dan mereka yang ingin menghasilkan profit dari internet.

Untuk memahami konsep e-book ini dengan lebih baik. Anda diharapkan sudah mengenal apa itu internet, email, dan pengetahuan dasar komputer lainnya.

Namun syarat terpenting hanyalah **kemauan** anda untuk mempelajari e-book harus tinggi. Itu saja!.

Apa itu HTML?

HTML(Hyper Text Markup Language) adalah sekumpulan simbol-simbol atau tag-tag yang dituliskan dalam sebuah file yang dimaksudkan untuk menampilkan halaman pada web browser. Tag-tag tadi memberitahu browser bagaimana menampilkan halaman web dengan lengkap kepada pengguna.

Tag-tag HTML selalu diawali dengan <x> dan diakhiri dengan </x> dimana x tag HTML seperti b, i, u dan sebagainya.

Apa Itu PHP ?

PHP adalah bahasa server-side scripting yang didesain khusus untuk web. Pada halaman HTML anda dapat menempelkan(embed) kode PHP. Kode PHP dieksekusi di sisi server bukan di komputer klien. Dan hasil yang ditampilkan adalah kode HTML.

PHP merupakan hasil kerja seorang bernama Rasmus Lerdorf pada 1995. Namun kemudian PHP berkembang dan tidak hanya merupakan proyek pribadi Rasmus. PHP ditulis ulang dan dengan banyak menambahkan fungsi-fungsi baru oleh Zeev Suraski dan Andi Gutmants (disingkat Zend) dan lahirlah PHP 3 pada 1998.

PHP masih terus dikembangkan, pada tahun 2002 PHP hadir dengan versi ke 4. PHP 4 dilengkapi dengan Zend Engine dan mengalami banyak peningkatan performa. Dan yang paling akhir pada tahun 2005 PHP 5 hadir dengan dukungan Zend Engine 2. Banyak fitur baru khususnya pada OOP(Object Oriented Programming) dan XML.

Apa Itu MySQL?

MySQL adalah program *relational database management system (RDMS)* yang bersifat free dan open source. MySQL sangat cepat, gratis, mudah dikonfigurasi, mudah dipelajari dan tersedia source codenya. Itulah beberapa kelebihan utama dari MySQL dibanding RDMS komersial yang ada. MySQL menggunakan SQL (*Structured Query Language*) yang merupakan standar global dalam manajemen database.

Apa Itu Apache?

Apache adalah HTTP web server yang bersifat free dan open source. Oleh karena itu penggunaannya biasanya dibarengkan dengan PHP dan MySQL.

Dengan apache komputer kita seolah-olah menjadi sebuah server yang menangani request dari klien.

Apa Itu phpMyAdmin?

phpMyAdmin adalah kumpulan script PHP yang dijalankan pada browser yang fungsinya untuk administrasi database MySQL. Dengan menggunakan phpMyAdmin kita lebih mudah dalam melakukan manipulasi database karena phpMyAdmin menyediakan user interface yang baik.

Apa Saja Yang Dibutuhkan?

Untuk dapat mempelajari e-book ini minimal software-software yang diperlukan adalah:

- PHP > 4.3.x (Versi Penulis 5.1.2)
- MySQL > 4.x (Versi Penulis 5.0.16)
- Apache 2.x (Windows 2000/XP/2003) (Versi Penulis 2.0.54)
- Apache 1.3.x (Windows 98) (Versi Penulis 1.3.34)
- phpMyAdmin 2.6.x (Versi Penulis 2.6.1)
- PHP DESIGNER 2006 (opsional)
- hMailServer > 4.x (opsional) (Versi Penulis 4.2 build 195)

Semuanya **GRATIS**. Meskipun anda saat ini sudah memiliki PHP versi 4.3.x itu tidak masalah. Karena penulisan pada e-book ini juga mengutamakan kompatibilitas pada versi sebelumnya.

Software yang bersifat opsional artinya software tersebut tidak mutlak harus anda miliki. Namun akan lebih baik jika anda memiliki.

Jika anda mengetik kode menggunakan **notepad** itu sudah cukup. Namun sangat risikan terjadi kesalahan syntax karena tidak ada pewarnaan kode. Dengan PHP DESIGNER kode-kode yang anda ketikkan akan diberi warna secara otomatis sesuai dengan jenisnya. Anda juga dapat mengkonfigurasi sendiri warnanya sesuai keinginan anda.

Apa itu hMailServer?

hMailServer adalah program yang mail server yang bersifat free dan open source untuk sistem operasi Windows. Karena pada aplikasi web biasanya menggunakan email. Maka dengan hMailServer komputer kita dapat menjadi mail server virtual sekaligus klien.

Dengan demikian seolah-olah kita mengirim dan menerima email dari dan ke internet. Padahal proses tersebut hanya terjadi pada komputer kita saja(lokal).

Alamat Download

Berikut ini adalah alamat download dari masing-masing software:

Software	Alamat Download
PHP	http://www.php.net/
MySQL	http://www.mysql.com/
Apache	http://httpd.apache.org/
phpMyAdmin	http://www.phpmyadmin.net/
PHP DESIGNER 2006	http://mpsoftware.dk/
hMailServer	http://www.hmailserver.com/

Jika anda merasa malas mendownload anda dapat memesan software-software diatas kepada saya dalam bentuk CD. Caranya masuk pada member area, buka halaman pesan CD dengan mengklik link pesan cd. Isi dari CD tersebut antara lain:

1. PHP 5.1.2 (Windows + Linux)

2. Apache 2.0.54/1.3.34 (Windows), Apache 2.0.55 (Linux)
3. MySQL 5.0.16 (Windows), MySQL 5.0.18 (Linux)
4. phpMyAdmin 2.6.x
4. Dokumentasi instalasi
5. hMailServer 4.2 build 195
6. PHP DESIGNER 2006
7. E-book "Menjadi Webmaster Dalam 30 Hari"
8. E-book "Mastering Kode HTML"
9. AceFTP 3 Free Edition (Software FTP)

(Isi dari CD dapat berubah sewaktu-waktu sesuai dengan rilis terbaru dari masing-masing vendor)

Instalasi dan konfigurasi

Pada bagian instalasi dan konfigurasi ini kita bagi dalam dua bagian. Yang pertama adalah untuk sistem operasi Microsoft Windows dan yang kedua adalah untuk Linux kernel > 2.4.x.

NB: Pada contoh instalasi di e-book ini menggunakan versi PHP, MySQL, dan Apache yang telah disebutkan sebelumnya. Ketika anda membaca e-book ini mungkin versi rilis yang disediakan berbeda dengan versi yang digunakan pada contoh ini.

Saya sarankan anda menggunakan versi rilis yang up to date. Meskipun demikian langkah-langkah instalasinya tidak jauh berbeda dengan yang digunakan pada ebook ini.

Jika anda ingin menggunakan versi software yang digunakan di e-book ini anda dapat mendownload di alamat:

PHP 5.1.2 : <http://www.php.net/get/php-5.1.2-Win32.zip/from/a/mirror>
Apache 2.0.54/1.3.34 : <http://archive.apache.org/dist/httpd/binaries/win32/>

Pada contoh contoh instalasi ini PHP 5.1.2 mewakili PHP versi 5. Apache 2.0.54 mewakili Apache versi 2.0.x. Sedangkan Apache 1.3.34 mewakili Apache versi 1.3.x.

Untuk dokumentasi dari instalasi versi terbaru silahkan kunjungi:
<http://materi.rahasia-webmaster.com/>

Windows

Instalasi Apache 2.0.54 (Windows 2000/XP/2003)

File yang saya gunakan adalah apache_2.0.54-win32-x86-no_ssl.msi. Untuk instalasi apache sangat mudah karena hanya tinggal klak-klik saja seperti software windows pada umumnya. Ikuti langkah-langkah berikut:

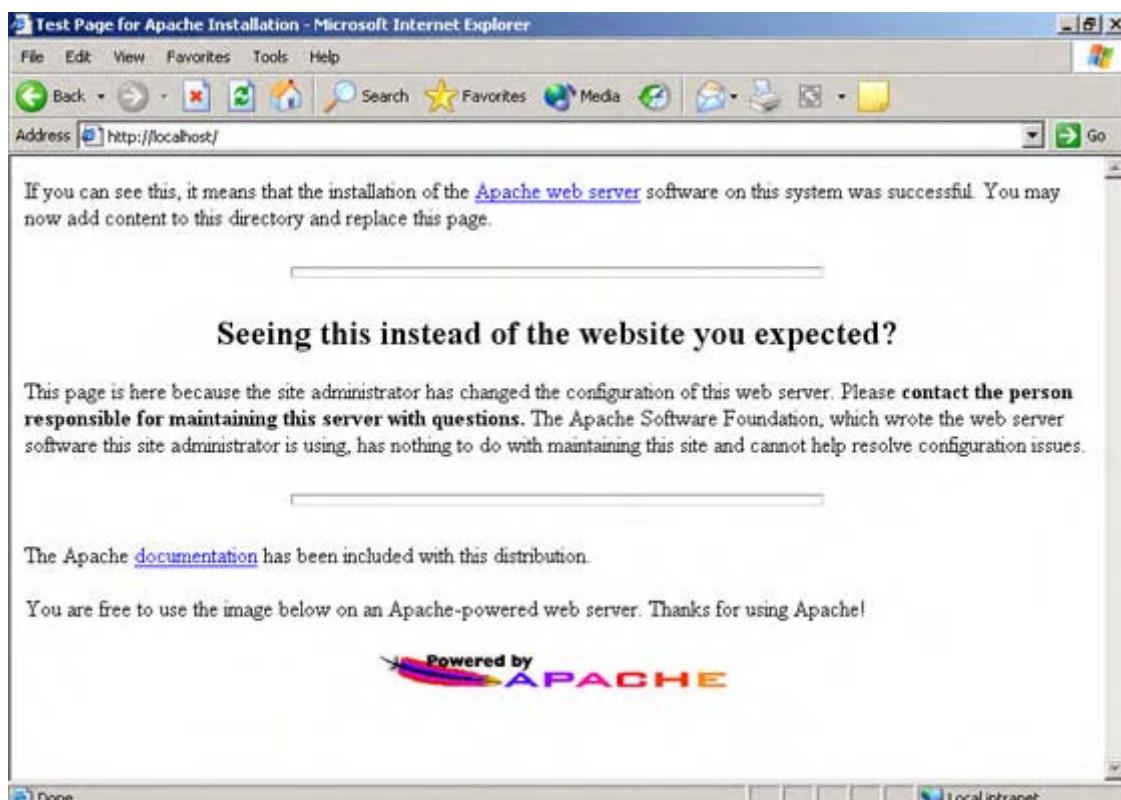
1. Setelah anda download file instalasinya klik dua kali file tersebut.
2. Klik Next > Pilih I Accept Klik Next > klik Next >
3. Pada Network Domain Isikan dan Server Name Isikan domain anda misalnya *localhost*
4. Pada Administrator Email isikan email anda. Ini tidak penting jika anda hanya menggunakan secara personal bukan untuk publik.
5. Klik for All User, On Port Klik Next >
6. Pilih Typical, Klik Next >
7. Klik Change... pada Isian folder name ganti **C:\Program Files\Apache Group** dengan **C:** klik OK(nantinya Apache akan membuat folder C:\Apache2 secara otomatis)
8. Klik Next > klik Install (Proses instalasi dimulai)
9. Klik Finish jika sudah selesai

Setelah proses instalasi selesai akan muncul ikon Apache Monitor (bulu merah dan segitiga hijau). Jika hijau artinya server apache sedang berjalan. Jika merah berarti server apache tidak aktif.

Untuk mengetes apakah Apache berjalan dengan baik atau tidak. Ikuti langkah-langkah berikut.

1. Buka browser anda (Misalnya Internet Explorer)
2. Pada address bar ketikkan `http://localhost/` atau `http://127.0.0.1/`
3. Maka akan muncul halaman seperti ditunjukkan gambar 1.0.1

NB: Jika anda menggunakan Windows XP Service Pack 2. Anda akan menemui window peringatan apakah Apache diperbolehkan menerima koneksi dari network. Klik tombol **Unblock** agar Windows Firewall tidak memblok program apache.exe.



Gambar 1.0.1: Halaman utama apache

Agar file-file PHP kita nantinya dapat diproses oleh Apache maka kita perlu meletakkannya pada direktori **C:\Apache2\htdocs**. Pada browser jika anda mengetik `http://localhost/` atau `http://127.0.0.1/` maka sama saja anda membuka direktori **C:\Apache2\htdocs**.

Instalasi Apache 1.3.34 (Windows 98)

File yang saya gunakan untuk instalasi ini adalah `apache_1.3.34-win32-x86-no_src.exe`. Langkah-langkah untuk instalasi apache 1.3.34 pada windows 98 sama dengan Apache 2.0.54 pada windows 2000/XP/2003.

Pada Apache 1.3.34 nama direktori penginstalan bukan `C:\Apache2` melainkan `C:\Apache`. Untuk mengetesnya buka windows explorer. Masuk pada direktori `C:\Apache` double klik file `Apache.exe` untuk mengeksekusinya.

Atau

1. Klik **Start > Run** ketik **command** [ENTER]
2. ketik **cd ** [ENTER] untuk berpindah ke direktori paling atas
3. ketik **cd C:\Apache** [ENTER] untuk masuk ke direktori **C:\Apache**
4. ketik **apache** [ENTER] untuk mengeksekusi file **Apache.exe**

Setelah file apache.exe dieksekusi maka server apache akan segera dijalankan. Hal itu ditandai dengan munculnya MS-DOS Window yang bertuliskan APACHE. Dan pesan yang ditampilkan kurang lebih bertuliskan seperti berikut,

Apache/1.3.34 (Win32) running...

Jangan tutup window tersebut. Karena jika anda tutup service Apache juga ikut dimatikan. Klik tombol minimize saja untuk meminimize window tersebut.

Untuk mematikan service dari Apache aktifkan dulu window APACHE. Setelah itu tahan tombol **Ctrl** kemudian tekan tombol **C**. Window APACHE secara otomatis akan hilang.

Instalasi MySQL 5.0.16 (Windows 2000/XP/2003)

File yang saya gunakan untuk instalasi ini adalah mysql-essential-5.0.16-win32.msi. Versi essential merupakan instalasi dengan kelengkapan minimal. Namun untuk windows 98 anda harus mendownload versi full. Instalasi MySQL lebih rumit dibanding Apache untuk itu ikuti langkah-langkah berikut dengan seksama:

1. Double klik file instalasi yang telah anda download
2. Klik Next >, Klik Custom, Klik Change ...
3. Pada isian folder name ganti **C:\Program Files\MySQL\MySQL Server 5.0** dengan **C:\MySQL5.0** klik OK
4. Klik Next > , klik Install untuk memulai proses instalasi
5. Setelah itu muncul window Setup Wizard pilih **Skip Sign-Up** lalu klik Next >
6. Muncul Window Baru, beri cek pada **Configure MySQL Server now** lalu klik Next >
7. klik Next > Pilih **Detailed Configuration**, klik Next >, Pilih **Developer Machine**
8. klik Next > Pilih **Multifunctional Database**, klik Next > Klik Next
9. Beri Cek pada **Enable TCP/IP Networking** pastikan port numbernya 3306.
10. Beri Cek pada **Enable Strick Mode**, klik Next >, klik Next >
11. Beri Cek pada **Install As Windows Service**, Service Name pilih **MySQL**
12. Beri Cek pada **Include Bin Direktori in Windows PATH** (agar kita dapat memanggil mysql dari command prompt) klik Next >
13. Beri Cek pada **Modify Security Settings**
14. Pada isian **New root password** ketikkan password yang ingin anda gunakan untuk masuk ke MySQL sebagai root(pemegang hak kuasa)
15. Ulangi pada **Confirm**
16. Jangan beri cek untuk Create An Anonymous Account
17. klik Next > klik Execute > Klik Finish untuk mengakhiri instalasi
18. Restart komputer anda agar semua setting dapat berjalan

NB: Jika anda menggunakan Windows XP SP 2 mungkin akan muncul pesan error ketika anda menekan tombol Execute. Hal ini disebabkan karena Windows Firewall memblok service mysqld-nt.exe yang menggunakan port 3306. Untuk mengatasinya, pergi ke Control panel > **Windows Firewall** > klik Tab **Exceptions** > klik **Add Program...** > Arahkan ke **C:\MySQL5.0\bin** pilih **mysqld-nt.exe** > klik OK > Kembali ke Window pesan error MySQL klik tombol **Retry** untuk mengulangi.

Untuk mengetesnya masuk pada command prompt caranya klik Start > Run > ketik cmd lalu klik OK.

1. Pada command prompt ketik **cd ** [ENTER] agar kita kembali ke direktori root(paling atas).

2. Untuk memastikan direktori C:\MySQL5.0\bin\ ada pada environment windows ketikkan perintah berikut
3. PATH [ENTER] kemudian cari apakah direktori C:\MySQL5.0\bin ada pada daftar.
4. Jika ya lanjut ke no. 6, jika belum klik kanan My Computer > Properties > Advanced > Environment variabels > Double klik **path** pada system variabels. Tambahkan data berikut ;C:\MySQL5.0\BIN (Setiap direktori dibatasi titik koma)
5. Restart komputer anda kemudian lanjutkan ke langkah 6
6. Ketik mysql [ENTER] jika ada error ERROR 1045 (28000): Access denied for user 'ODBC'@'localhost' (using password: NO)
7. Itu berarti benar. Loh error kok malah benar?. Itu berarti user Anonymous tidak dapat mengakses MySQL dan hanya root yang bisa.
8. ketikkan perintah berikut C:\>mysql -u root -p [ENTER] Maka akan muncul pertanyaan ENTER PASSWORD:
9. Masukkan password anda yang anda gunakan pada waktu instalasi tadi lalu tekan ENTER.
10. Jika password benar maka anda akan ke MySQL Monitor yang ditandai dengan prompt mysql> ketik \h untuk bantuan atau \q untuk keluar dari MySQL monitor
11. MySQL Monitor ditunjukkan oleh gambar 1.0.2.

```
D:\>Documents and Settings\Rio Astaman>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2 to server version: 5.0.16-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> -
```

Gambar 1.0.2: Tampilan MySQL Monitor

Pembahasan lengkap mengenai MySQL akan kita pelajari pada Bab 3.

Instalasi MySQL 5.0.16 (Windows 98)

Berdasarkan hasil test, MySQL 5.0.16 versi essential tidak dapat berjalan di Windows 98. Oleh karena itu jika anda pengguna windows 98 anda harus mendownload versi full yang ukuran filenya lebih besar namun lebih komplit.

Langkah-langkah instalasi MySQL 5.0.16 pada windows 98 tidak jauh beda dengan windows 2000/XP. Hanya saja pada Windows 98 langkah instalasinya hanya sampai langkah ke-12.

Karena MySQL tidak dapat berjalan sebagai service di windows 98. Maka untuk menjalankan server MySQL anda harus memanggilnya secara manual. Anda dapat menjalankan server MySQL dengan menjalankan file **mysqld.exe** yang ada di direktori C:\MySQL5.0\Bin lewat Windows Explorer. Jika mau Anda dapat membuat shorcutnya di desktop.

Alternatif lain adalah dengan menggunakan MS-DOS Prompt. Klik **Start > Run** ketik **command** lalu ENTER.

1. Setelah muncul DOS Prompt ketik **cd ** [ENTER] untuk ke direktori paling atas.
2. Masuk pada direktori C:\MySQL5.0\bin caranya ketik cd MySQL5.0\bin [ENTER]

3. Ekeskusi file mysqld.exe dengan mengetik mysqld [ENTER]
4. Untuk mengetes server MySQL telah aktif ketik mysql [ENTER]
5. Jika berhasil anda akan masuk pada MySQL Monitor seperti gambar 1.0.2 sebelumnya.

Khusus Pengguna Windows 98

Jika anda lebih suka mengaktifkan server lewat MS-DOS Prompt daripada windows Explorer. Sebelum mengeksekusi MySQL.exe atau Apache.exe anda tentu harus berpindah-pindah direktori terlebih dahulu.

Hal tersebut tentu kurang efisien. Agar direktori C:\MySQL5.0\bin dan C:\Apache dapat dikenali di seluruh lingkungan DOS. Kita perlu menset sistem PATH windows pada file autoexec.bat. Caranya,

1. pindah ke direktori paling atas dengan perintah cd \ [ENTER]
2. pada C:> ketik perintah berikut edit autoexec.bat [ENTER]
3. Tambahkan baris perintah berikut
SET PATH=C:\WINDOWS;C:\WINDOWS\COMMAND;C:\MySQL5.0\BIN;C:\Apache
4. Tahan tombol **Alt** lalu tekan **F** pilih Save... untuk menyimpan file.
5. Restart komputer anda.

Sekarang anda dapat mengeksekusi semua file pada direktori C:\MySQL5.0\BIN dan C:\Apache tanpa harus masuk pada direktori tersebut.

Jika anda ingin server otomatis berjalan ketika Windows startup kita dapat memasukkannya ke dalam registry windows. Langkah-langkahnya adalah sebagai berikut:

1. Jalankan program notepad klik **Start > Run** ketik notepad [ENTER]
2. Kemudian ketik script berikut

REGEDIT4

```
[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run]
"MySQL5.0 Server"="C:\\MySQL5.0\\bin\\mysqld.exe"
"Apache 1.3.34"="C:\\Apache\\Apache.exe"
```

3. Simpan dengan mengklik menu **File > Save**
4. Tentukan lokasi penyimpanan sesuai keinginan anda
5. Pada Save As Type pastikan pilihannya All Files
6. Pada isian File Name: isikan **server.reg** lalu klik Save
7. Jalankan Windows Explorer arahkan ke tempat dimana anda menyimpan file tersebut
8. Double klik file tersebut untuk menjalankannya
9. Pada dialog pertanyaan klik **Yes** kemudian klik **OK**.

Script diatas bertujuan untuk memasukkan mysqld.exe dan apache.exe ke dalam daftar file-file yang dieksekusi saat windows startup.

Jika anda ingin mengetahui lebih tentang registry silahkan cari sendiri di Internet lewat search engine. Saran saya sebelum mengutak-atik registry windows backuplah data-data penting anda. Karena jika anda salah mensetting registry akibatnya sangat **fatal** yaitu Windows tidak bisa dijalankan.

Instalasi PHP 5.1.2

File yang saya gunakan adalah php-5.1.2-Win32.zip. Kita lebih memilih dalam bentuk ZIP karena fitur-fitur yang didukung PHP lebih banyak daripada versi Installer.

Ada beberapa bagian instalasi PHP yang menuntut anda untuk "turun tangan" secara manual. Turun tangan disini hanya sebatas mengedit beberapa file konfigurasi. Ikuti langkah-langkah instalasi berikut:

- Extract file instalasi menggunakan winzip atau program sejenis
- Misalnya Winzip Double klik file instalasi, klik Extract pada toolbar
- Pastikan Anda memilih All Files/folders in archive
- Pada isian Extract to isikan **C:\php5.1.2** lalu klik Extract
- Masuk pada direktori C:\php5.1.2, rename file **php.ini-dist** menjadi **php.ini**
- Double klik php.ini untuk mengeditnya
- Cari baris dengan tulisan berikut
`;include_path = ".;c:\php\includes"`

Ganti dengan
`include_path = ".;c:\php5.1.2\includes"`

Hilangkan juga titik komanya
- Cari baris dengan tulisan berikut
`extension_dir = "./"`

Ganti dengan
`extension_dir = "c:\php5.1.2\ext"`
- Cari bagian extention lalu hilangkan tanda ; (titik koma) didepan extention-extention berikut:
`;extension=php_mbstring.dll` menjadi `extension=php_mbstring.dll`
`;extension=php_mysql.dll` menjadi `extension=php_mysql.dll`
- Simpan kembali dengan mengklik File > Save

Untuk Windows 98

Setelah kesembilan langkah tersebut selesai anda ikuti. Anda perlu mengkopi file **php.ini** yang ada pada direktori **c:\php5.1.2** ke direktori **c:\windows**.

Sudah selesai? Belum..., masih ada satu file lagi yang akan kita edit. File ini adalah file konfigurasi Apache. Tanpa mengedit file ini script PHP tidak akan bisa diproses.

Konfigurasi Apache 2.0.54 (Windows 2000/XP/2003)

Buka file **httpd.conf** yang ada di direktori C:\Apache2\conf dengan notepad, kemudian ikuti langkah-langkah berikut:

- Cari bagian LoadModule pada bagian bawah tambahkan kode berikut

```
# Module untuk PHP 5
LoadModule php5_module "C:/php5.1.2/php5apache2.dll"
AddType application/x-httdp-php .php

# Untuk pewarnaan source code PHP
AddType application/x-httdp-php-source .phps

# Letak direktori file php.ini
PHPIniDir "C:/php5.1.2"
```

- Cari bagian Directoryindex yang berisi tulisan berikut

`DirectoryIndex index.html index.html.var`

Tambahkan hingga menjadi seperti berikut

DirectoryIndex index.html index.htm index.htm.var index.php

3. Simpan kembali file httpd.conf dengan mengklik File > Save
4. Klik kanan icon Apache Monitor pada pojok kanan layar kanan > Klik **Open Apache Monitor** > Klik **Restart** > Klik **OK**

Konfigurasi Apache 1.3.34 (Windows 98)

Buka file **httpd.conf** yang ada di direktori **C:\Apache\htdocs**, kemudian modifikasi bagian-bagian berikut:

1. Cari bagian LoadModule pada bagian bawah tambahkan kode berikut

```
# Module untuk PHP 5
LoadModule php5_module "C:/php5.1.2/php5apache.dll"
```

2. Pada bagian bawah AddModule tambahkan kode berikut

```
# Module untuk PHP 5
AddModule mod_php5.c
```

3. Cari bagian yang bertuliskan `<IfModule mod_mime.c>`

```
<IfModule mod_mime.c>
    TypesConfig conf/mime.types
</IfModule>
```

Modifikasi hingga menjadi seperti berikut

```
<IfModule mod_mime.c>
    TypesConfig conf/mime.types

    # Script PHP
    AddType application/x-httpd-php .php

    # Untuk pewarnaan Source Code PHP
    AddType application/x-httpd-php-source .phps
</IfModule>
```

4. Cari bagian Directoryindex yang berisi tulisan berikut

DirectoryIndex index.html

Tambahkan hingga menjadi seperti berikut

DirectoryIndex index.html index.htm index.php

5. Simpan kembali file httpd.conf klik File > Save

6. Untuk merestart Apache tutup DOS window Apache server running ...

7. Double klik file shortcut Apache pada desktop jika anda membuatnya pada saat instalasi

8. Jika tidak, masuk pada DOS Prompt ketik C:\>apache.exe [ENTER] untuk mengaktifkan Apache HTTP Server

Lanjutan (Apache 2.0.54 dan Apache 1.3.34)

Sudah selesai? hampir.... Sekarang kita hanya perlu membuat sebuah file untuk mengetes apakah Apache dapat memproses script PHP. Caranya:

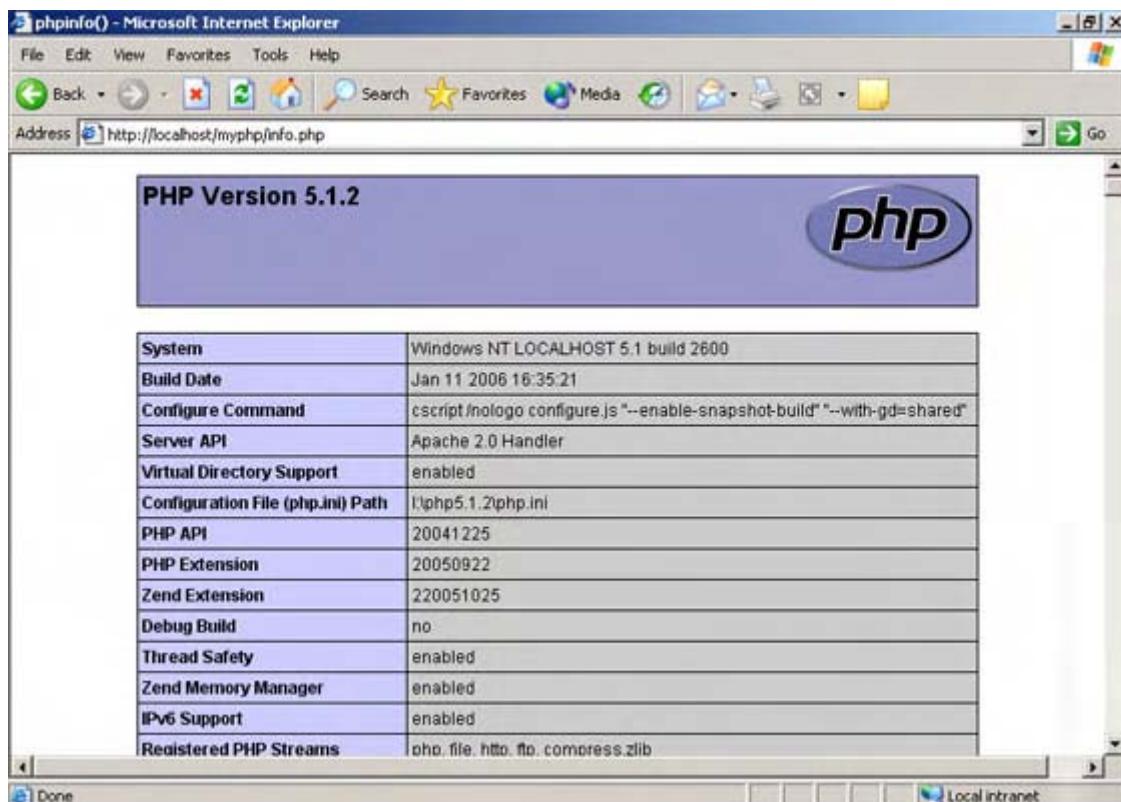
1. Buka notepad atau PHP DESIGNER 2005 jika anda sudah mendownload. Ketik kode berikut:

```
<?php  
phpinfo();  
?>
```

2. Simpan pada direktori C:\Apache2\htdocs\ (Apache 2.0.54) atau C:\Apache\htdocs\ (Apache 1.3.34) dengan nama **info.php**.
3. Jika menggunakan notepad pastikan **Save As**-nya adalah **All Files** agar tidak disimpan sebagai file TXT.
4. Buka browser anda arahkan ke <http://localhost/info.php>. Jika segala sesuatunya benar akan muncul tampilan seperti gambar 1.0.3.

Jika belum bisa coba cek kembali mungkin ada beberapa langkah yang anda lewatkan.

Instalasi yang telah kita lakukan sebelumnya merupakan proses instalasi PHP sebagai Modul Apache. Kita tidak menginstal PHP sebagai CGI karena karena alasan keamanan. Selain itu performa PHP sebagai modul Apache lebih baik daripada sebagai CGI.



Gambar 1.0.3: Script PHP berhasil diproses Apache

Instalasi phpMyAdmin 2.6.1

File instalasai yang saya gunakan adalah phpMyAdmin-2.6.1.zip. Saya asumsikan anda menggunakan winzip untuk mengextract file instalasi tersebut.

1. Buka file zip tersebut menggunakan WinZip
2. Klik Extract pada toolbar, Pastikan Anda memilih All Files/folders in archive

Untuk Apache 2.0.54

3. Pada isian Extract To isikan C:\Apache2\htdocs\phpMyAdmin-2.6.1
4. Edit file config.inc.php pada direktori C:\Apache2\htdocs\phpMyAdmin-2.6.1

Untuk Apache 1.3.34

3. Pada isian Extract To isikan C:\Apache\htdocs\phpMyAdmin-2.6.1
4. Edit file config.inc.php pada direktori C:\Apache\htdocs\phpMyAdmin-2.6.1

5. Cari string berikut

```
$cfg['PmaAbsoluteUri'] = '';
```

Ganti nilainya menjadi

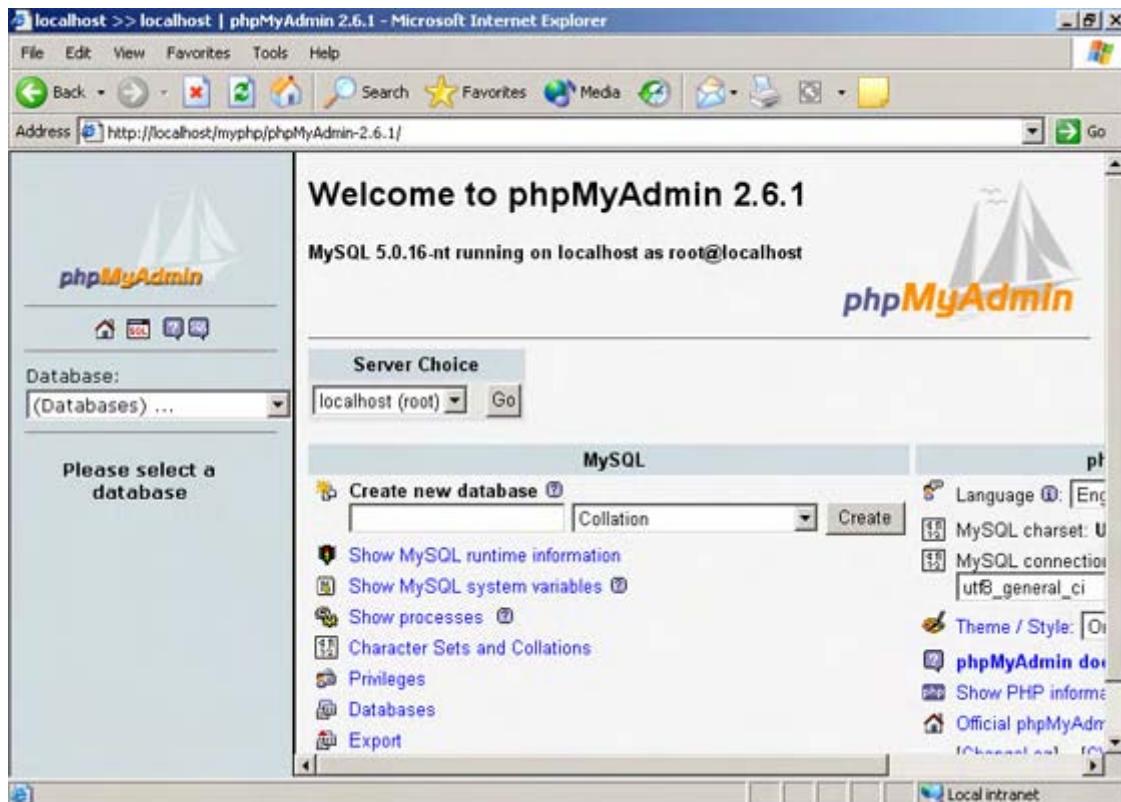
```
$cfg['PmaAbsoluteUri'] = 'http://localhost/phpMyAdmin-2.6.1/';
```

6. Cari string berikut dan set nilainya sehingga seperti dibawah ini:

```
$cfg['Servers'][$i]['host']      = 'localhost';
$cfg['Servers'][$i]['user']      = 'root';
$cfg['Servers'][$i]['password']  = 'GANTI_DENGAN_PASSWORD_ANDA' ;
```

7. Buka browser anda arahkan ke alamat http://localhost/phpMyAdmin-2.6.1/

Jika anda menjalankan sesuai instruksi yang saya berikan tampilan pada browser anda akan seperti gambar 1.0.5



Gambar 5.0.5: Tampilan utama phpMyAdmin

Untuk sementara jangan diutak-atik dulu agar tidak terjadi hal-hal yang diinginkan. Pembahasan tentang MySQL dan phpMyAdmin akan kita kupas pada bab 3.

Instalasi hMailServer 4.2 (Opsional)

Anda tidak harus menginstall hMailServer untuk dapat mengaplikasikan isi e-book ini. Hanya saja dengan hMailServer komputer anda akan lebih lengkap karena memiliki Mail Server. Kita tahu pada aplikasi web sering terjadi proses pengiriman email. Tanpa memiliki mail server kita tidak bisa mengetahui script kita bekerja atau tidak.

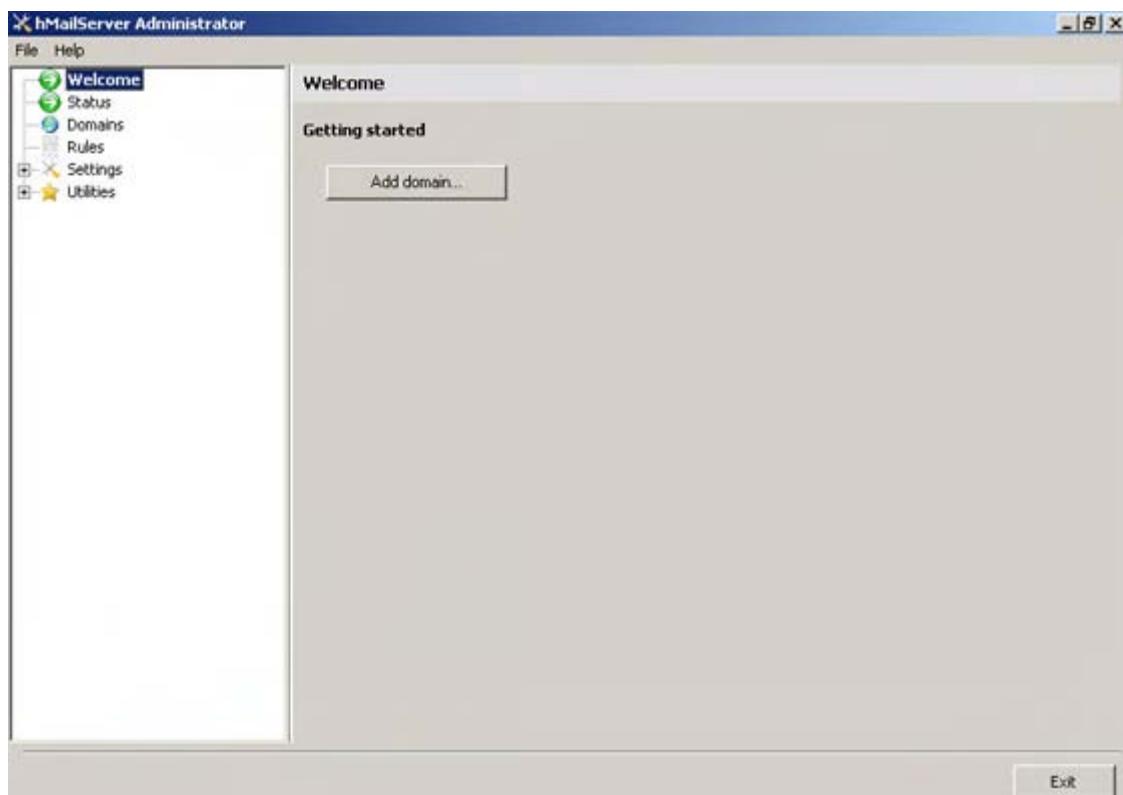
File instalasi yang saya gunakan adalah hMailServer-4.2-Build-195.exe. Untuk instalasi hMailServer ikuti langkah-langkah berikut:

1. Double klik file instalasi tersebut.
2. Klik Next > Pada folder tujuan isikan **C:\Program Files\hMailServer**
3. Klik Next > Pilih **Custom Installation**
4. Klik Next > pilih **Use External database server**
5. Klik Next > klik Next > klik Install
6. Muncul window baru klik Next > pilih **Create a new hMailServer database**
7. Klik Next > pilih MySQL Server, klik Next >
8. Pada database address isikan **localhost**, TCP Port **3306**
9. Username isikan **root**, password isikan password anda pada waktu instalasi MySQL
10. Pada database name isikan **hmailserver** klik Next >, Klik Next >
11. klik Finish, pindah pada window Setup – hMailServer klik **Finish**

Setelah itu akan muncul window hMailServer Administrator – Connect.

1. Jika pada hostname belum ada localhost maka klik **Add** isikan localhost lalu klik Save.
2. Jika anda ingin hmail terkoneksi saat start-up centang juga Automacally connect at start up.
3. Terakhir klik Connect

Tampilan hMailServer saat pertama kali dijalankan ditunjukkan oleh gambar 1.0.6



Gambar 1.0.6: Tampilan hMailServer saat pertama kali dijalankan

Setelah proses instalasi selesai. Kita perlu membuat domain untuk alamat email kita. Karena kita berada di komputer kita sendiri kita bebas menentukan nama domain. Misalnya `alamat@nama-anda.com` dapat anda buat nama domain.

Untuk contoh ini saya menggunakan domain localhost.localdomain. Untuk sementara saya sarankan anda untuk menggunakan domain tersebut. Setelah anda mengerti cara kerjanya anda dapat mengganti sesuka anda. Untuk membuat domain ikuti langkah-langkah berikut:

1. Pada list menu, klik Welcome pada bagian kiri. Kedua klik **Add domain...**
2. Pada Domain Name isikan **localhost.localdomain** klik **Save**
3. Klik **Add Account**, pada Account address isikan **test**
4. Pada Password isikan password isikan **pas123**
5. Pada administration anda dapat memilih antara user, domain, dan server. Jika anda ingin account tersebut dapat digunakan untuk administrasi server pilih server.
6. Jangan lupa beri cek Active kemudian klik Save



Gambar 1.0.7: User test menerima email

Sekarang anda sudah memiliki account email dengan alamat **test@localhost.localdomain**. Untuk mengetes apakah mail server kita berjalan dengan baik kita dapat menggunakan program email client seperti Outlook Express bawaan windows.

1. Jalankan program Outlook Express
2. Klik Menu > Tools > Account
3. Klik tombol Add > Mail...
4. Pada Display Name isikan Nama Anda, pada email Address isikan **test@localhost.localdomain**
5. My Incoming Mail server is a pilih **POP3**
6. Pada incoming mail isikan **localhost**, pada Outgoing mail juga isikan **localhost**
7. Klik Next, Account Name isikan **test@localhost.localdomain**
8. Password isikan **pas123** klik Next > Klik Finish
9. Pada window Internet Accounts pilih **localhost** klik Properties
10. ganti localhost dengan Testing lalu klik OK, Klik Close

Kembali pada hMailServer:

1. Klik tanda + pada Utilities untuk memunculkan tree baru
2. Klik **Server sendout**
3. Pada bagian email, From isikan **AdminTest**, pada From Address isikan **no-reply@localhost.localdomain**
4. Pada subject isikan *Test Test Test Test x999999*
5. Pada body isikan *Jika anda dapat membaca pesan ini, SELAMAT!. Mail server anda telah berjalan dengan sukses!.*
6. Klik Send, muncul dialog box Klik OK.

Kembali pada Outlook Express, klik Send/Recv pada toolbar. Pada inbox nampak 1 email baru, klik inbox tersebut lalu double klik AdminTest untuk memunculkan pesannya seperti gambar 1.0.7.

Linux Kernel > 2.4.x

Sistem operasi yang saya gunakan untuk pengujian adalah Red Hat 9 (Kernel 2.4) dan Ubuntu Breezy (Kernel 2.6). Dan saya rasa langkahnya hampir sama untuk semua distro linux. Selama proses penginstalan anda harus berada pada mode super user (root).

Anda dapat menggunakan perintah su untuk berpindah ke mode super user. Semua paket yang akan kita install bukan rpm melainkan sourcanya. Sourcanya saya letakkan pada direktori /usr/src.

Jika anda sudah menginstall MySQL, Apache dan PHP yang disertakan dalam CD distribusi Red Hat 9. Hapus kesemua paket-paket tersebut karena sudah *Out of Date*. Anda dapat menggunakan Add/Remove Applications yang berbasis GUI untuk melakukannya.

Pada ubuntu breezy anda dapat menggunakan Synaptic untuk menghapus instalasi apache jika anda terlanjur menginstalnya.

Ketiga paket yang saya gunakan antara lain:

- MySQL versi 5.0.18 (mysql-5.0.tar.gz)
- Apache versi 2.0.55 (httpd-2.0.55.tar.gz)
- PHP versi 5.1.2 (php-5.1.2.tar.gz)

Untuk mail server kita dapat menggunakan sendmail (Red Hat) atau postfix (Ubuntu Breezy).

Instalasi MySQL 5.0.18

Versi MySQL yang saya gunakan dalam pengujian adalah MySQL versi 5.0.18. Langkah-langkahnya adalah:

1. Buka terminal window.
2. Jika anda login sebagai user biasa. Pindah ke mode super user dengan menggunakan perintah su.
3. Masuk ke direktori /usr/src lalu extract isi paket:

```
# cd /usr/src
# gzip -d mysql-5.0.tar.gz
# tar -xvf mysql-5.0.tar -C /usr/local
# cd /usr/local
# ln -s mysql-standard-5.0.18-linux-i686-glibc23 mysql
# cd mysql
```

Catatan: sesuaikan dengan versi mysql anda.
4. Jika anda tidak memiliki user biasa (bukan root), maka buatlah

```
# groupadd userx
# useradd userx
```

Catatan: ganti userx sesuai dengan keinginan anda.
5. Install database mysql

```
# ./scripts/mysql_install_db --user=userx
```

6. Buat privilege untuk direktori data kepada userx

```
# chown -R root .
# chown -R userx data
# chgrp -R userx .
```
7. Jalankan server mysql daemon

```
# ./bin/mysql_safe --user=userx & atau # ./bin/mysqld --user=userx &
```

Catatan: mysql_safe biasanya digunakan untuk pengetesan saja.
8. Buat password untuk root (root mysql bukan root sistem)

```
# ./bin/mysqladmin -u root password xxxxxxxx
```

Catatan: ganti xxxxxxx dengan password yang anda inginkan
9. Login ke MySQL

```
# ./bin/mysql -u root -p
```

Enter password: (ketik password anda lalu ENTER)
10. Hapus anonymous user

```
mysql> use mysql;
mysql> DELETE FROM user WHERE User='';
```
11. Jika ingin keluar dari MySQL Monitor gunakan perintah berikut

```
mysql>\q atau quit;
```
12. Untuk menghentikan service mysql ketik perintah berikut

```
# ./bin/mysqladmin shutdown -u root -p
```

Enter Password: (Masukkan password anda)

Instalasi Apache 2.0.55

Aplikasi ke dua yang akan kita instal adalah Web Server Apache. Versi Apache yang saya gunakan adalah Apache versi 2.0.55. Anda harus berada pada mode super user untuk melakukan instalasi ini. Langkah-langkahnya.

1. Masuk pada direktori /usr/src untuk mengextract paket:

```
# cd /usr/src
# gzip -d httpd-2.0.55.tar.gz
# tar -xvf httpd-2.0.55.tar
# ln -s httpd-2.0.55.tar httpd
# cd httpd
```

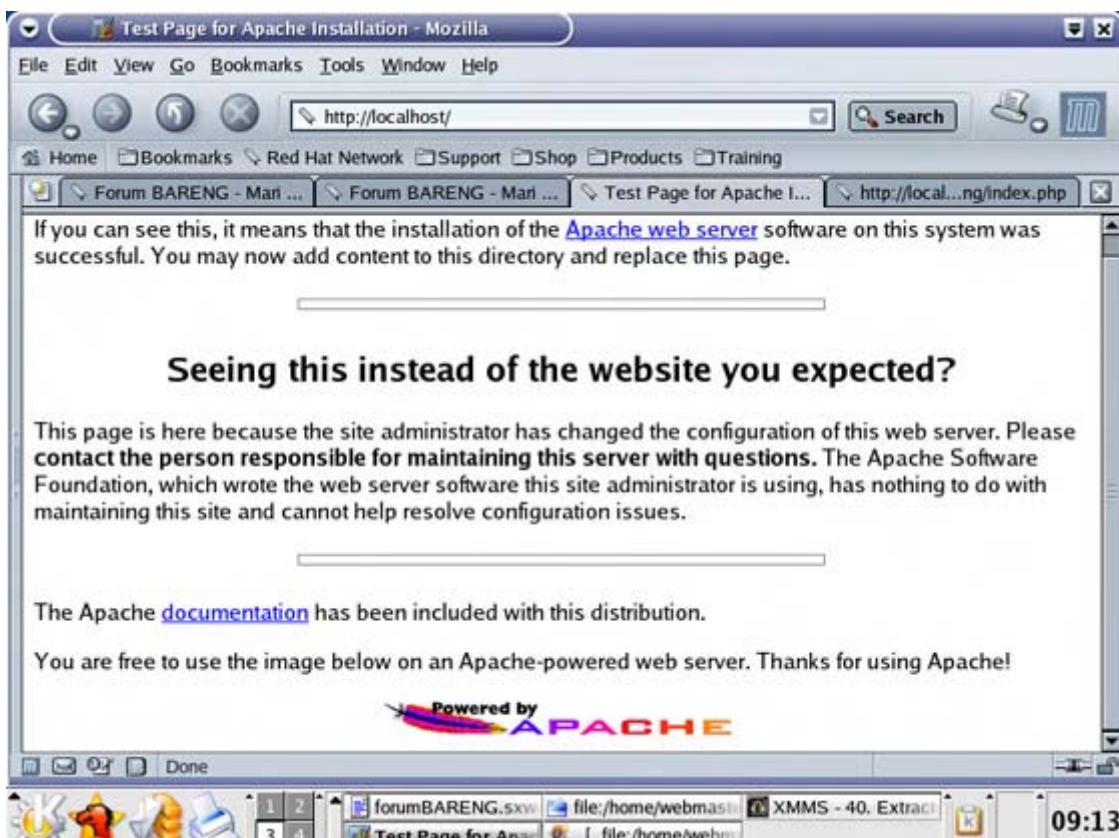
Catatan: sesuaikan dengan versi Apache anda.
2. Install ke direktori /usr/local/apache2

```
# ./configure --prefix=/usr/local/apache2
# make
# make install
```
3. Masuk pada direktori /usr/local/apache2

```
# cd /usr/local/apache2
```
4. Jalankan service httpd

```
# ./bin/apachectl start
```
5. Tes apakah apache sudah terkonfigurasi dengan benar dengan cara:
 - buka browser anda (Mozilla, Firefox atau yang lain) arahkan ke <http://localhost/>
 - Jika berhasil maka akan muncul halaman seperti gambar 1.0.8
6. Buat direktori webmaster pada /home/nama_user

- ```
mkdir /home/nama_user/webmaster
chown -R nama_user /home/nama_user/webmaster
```
7. Buat symbolic links di direktori htdocs yang mengarah ke /home/nama\_user/webmaster  
`# ln -s /home/nama_user/webmaster /usr/local/apache2/htdocs/webmaster`  
 Catatan: jadi jika anda membuka alamat http://localhost/webmaster maka secara otomatis akan diarahkan ke /home/nama\_user/webmaster
8. Untuk menghentikan service httpd gunakan perintah berikut  
`# ./bin/apachectl stop`



Gambar 1.0.8: Konfigurasi Apache berhasil

## Instalasi PHP 5.1.2

Pada saat menginstall PHP kita dihadapkan dua pilihan. Apakah kita akan menginstall PHP sebagai Module dari Apache atau sebagai CGI. Pada sistem Linux dianjurkan untuk menginstall PHP sebagai module. Karena performa PHP sebagai module lebih baik dari pada dipasang sebagai CGI.

### Langkah-langkah penginstalan

1. Masuk pada direktori /usr/src  
`# cd /usr/src`
2. Extract paket PHP  
`# gzip -d php-5.1.2.tar.gz`  
`# tar -xvf php-5.1.2.tar`  
`# ln -s php-5.1.2 php`  
`# cd php`
3. Lakukan konfigurasi  
`# ./configure --prefix=/usr/local/php5 --with-apxs2=/usr/local/apache2/bin/apxs`

```
--with-mysql=/usr/local/mysql
```

Catatan: Jika ada masalah dengan versi libxml anda. Gunakan perintah berikut

```
./configure --prefix=/usr/local/php5 --with-apxs2=/usr/local/apache2/bin/apxs
--with-mysql=/usr/local/mysql --disable-libxml --disable-dom --disable-xml
--disable-simplexml --disable-xmlreader --disable-xmlwriter --without-pear
make
make install
```

4. Setelah itu edit file httpd.conf yang ada pada direktori /usr/local/apache2/conf. Jika anda pada terminal window anda dapat menggunakan gedit atau kedit.

```
gedit /usr/local/apache2/conf/httpd.conf
```

5. Tambahkan kedua baris ini di akhir file httpd.conf

```
AddType application/x-httdp-php .php .phtml
AddType application/x-httdp-php-source .phps
```

6. Pastikan pada bagian Dynamic Shared Object (DSO) Support terdapat

```
LoadModule php5_module modules/libphp5.so
Jika tidak ada, tambahkan.
```

7. Pada bagian DirectoryIndex tambahkan index.php dan index.htm sehingga menjadi seperti berikut

```
DirectoryIndex index.html index.htm.var index.php index.htm
```

8. Copy file php.ini-dist ke /usr/local/lib menjadi php.ini

```
cp /usr/src/php/php.ini /usr/local/lib/php.ini
```

9. Restart server httpd untuk mengetahui perubahannya

```
cd /usr/local/apache2
./bin/apachectl restart
```

10. Buat sebuah file php menggunakan editor kegemaran anda. Dengan isi sebagai berikut

```
<?php
phpinfo();
?>
```

Simpan pada direktori /home/nama\_user/webmaster dengan nama info.php

11. Buka browser anda arahkan ke http://localhost/webmaster/info.php

Gambar 1.1.0 menunjukkan output pemrosesan script PHP oleh modul Apache berjalan sukses.



Gambar 5.1.0: Pemrosesan Script PHP berhasil

# Hal-Hal Yang Perlu Diperhatikan

## Tentang Direktori Kerja

Agar pekerjaan kita lebih terstruktur, ikuti langkah-langkah berikut:

1. Masuk pada direktori C:\Apache2\htdocs (untuk Apache 2.0.54) atau C:\Apache\htdocs (untuk Apache 1.3.34)
2. Buat folder baru, beri nama folder tersebut **webmaster**

Untuk file-file yang kita kerjakan nantinya kita akan menyimpannya pada sub direktori webmaster.

Pada e-book ini pembahasan mengasumsikan anda menginstal Apache 2.0.54. Jadi direktori yang disebutkan pada pembahasan e-book ini adalah C:\Apache2\htdocs\webmaster.

Jadi jika anda menemukan C:\Apache2\htdocs\webmaster pada pembahasan e-book, itu berarti mengarah ke **C:\Apache\htdocs\webmaster** jika anda menggunakan **Apache 1.3.34**.

## Tentang Editor

Pada pembahasan e-book ini, diasumsikan anda menggunakan PHP DESIGNER 2006 untuk mengedit file-file PHP. Jadi jika ada perintah untuk menjalankan PHP DESIGNER 2006 anda dapat menggunakan editor lainnya, misalnya **notepad**.

Ada hal yang perlu anda perhatikan jika anda menggunakan notepad. Ketika anda hendak menyimpan file PHP **pastikan** pilihan **Save as Type**-nya adalah **All Files**. Begitu juga ketika hendak membuka file pastikan Save as Type adalah All Files agar file-file PHP muncul pada daftar. Hal ini untuk menghindari tersimpannya file anda sebagai file teks biasa.

# Bab

1

- H T M L
- P H P

Sebelum melanjutkan ikuti langkah-langkah berikut:

- Buatlah folder baru pada direktori C:\Apache2\htdocs\webmaster\
- Berinama folder tersebut **Bab1**

\*\*\*\*\*

Sebelum melanjutkan buatlah sebuah folder baru di direktori:

- Buat folder baru lagi di direktori C:\Apache2\htdocs\webmaster\Bab1
- Beri nama folder tersebut ***hari-1***

# H T M L

Pada hari yang pertama ini kita akan mempelajari HTML. HTML merupakan format tampilan untuk sebuah website. HTML sangat mudah dipelajari karena kita tidak memerlukan programming skill untuk menguasai HTML. Setelah membaca materi ini diharapkan anda dapat menguasai bagaimana:

- Menggunakan HTML
- Manipulasi Font dan Link
- Membuat Tabel
- Membuat Form HTML

## Menggunakan HTML

Sebuah halaman website yang valid akan selalu diapit tanda <html>...</html>. File-file HTML selalu berakhiran dengan ekstensi \*.htm atau \*.html. Tag-tag HTML selalu diawali dengan tanda <x> dan diakhiri dengan </x> dimana x adalah nama tag seperti b, i, u dan sebagainya. Namun ada juga tag yang tidak diakhiri dengan </x> seperti <br> dan <input>. Dibawah ini adalah tabel tag-tag HTML yang sering digunakan.

**Tabel 1.0** Tag-tag umum pada HTML

| Tag             | Keterangan                                   |
|-----------------|----------------------------------------------|
| <html></html>   | Tag untuk mengapit halaman HTML              |
| <head></head>   | Tag yang berisi informasi umum dari halaman  |
| <title></title> | Judul halaman. Terdapat pada <head>          |
| <body></body>   | Settingan attribut untuk seluruh isi halaman |
| <b></b>         | Untuk menebalkan teks                        |
| <i></i>         | Untuk memiringkan teks                       |
| <u></u>         | Untuk menggaris bawahi teks                  |
| <p></p>         | Untuk membuat paragraf                       |
| <font></font>   | Untuk memanipulasi huruf                     |
| <br>            | Untuk pindah ke baris baru                   |
| <hr>            | Untuk membuat garis horizontal               |
| <a></a>         | Untuk membuat links                          |

Agar anda lebih paham sebaiknya kita langsung menerapkan teori-teori diatas.

### PENERAPAN TEORI MENGGUNAKAN HTML

Pertama Jalankan program PHP DESIGNER 2006. Klik menu **File-New-HTML/XHTML** untuk membuat file HTML baru. Lalu ketik kode berikut ini.

```

<html>
<head>
<title>Website Pertamaku</title>
</head>
<body>

```

```
Halo Dunia...!, Ini website pertamaku
</body>
</html>
```

Simpan di folder hari-1 yang pernah kita buat. Dengan mengklik **File-Save** lalu simpan dengan nama ***pertama.html***. Untuk mengetesnya buka Internet Explorer atau browser favorit anda. Pada address arahkan ke <http://localhost/webmaster/Bab1/hari-1/pertama.html> tampilan yang anda lihat akan seperti gambar 1.0.



Gambar 1.0: output file pertama.html

## Manipulasi Font dan Link

Untuk memanipulasi font di HTML kita gunakan tag `<font></font>`. Tag font tidak hanya sesederhana itu. Tag ini memiliki atribut didalamnya. Dimana atribut inilah yang memiliki peranan penting dalam menentukan tampilan teks. Lihat Tabel 1.1 untuk daftar atribut tag font.

**Tabel 1.1** Daftar attribut dari tag font

Atribut	Keterangan
name	Untuk menentukan jenis huruf yang digunakan
size	Untuk menentukan ukuran huruf
color	Untuk menentukan warna huruf

Contoh jika kita ingin membuat teks yang menggunakan huruf Arial ukurannya 12pt dan warnanya merah. Kita dapat menuliskannya seperti berikut.

```
Teks Disini
```

Untuk size 3 sama dengan 12pt, 4 sama dengan 14pt, 5 sama dengan 18pt, 6 sama dengan 24pt dan 7 sama dengan 36pt. Lalu bagaimana jika saya ingin membuat teks yang hurufnya tebal, miring dan bergaris bawah?. Gampang..., kita cukup apit teks tersebut menggunakan tag `<b></b>`, `<i></i>` dan `<u></u>`. Lihat contoh berikut.

```
<i><u>Tebal, Miring dan Garis Bawah</u></i>
```

Teks diatas akan mempunyai efek tebal, miring dan bergaris bawah. Mudah bukan?. Selain font ada juga `<h1></h1>`, `<h2></h2>`, dan seterusnya yang digunakan untuk mencetak heading. `<h1>` akan mencetak heading dengan ukuran font yang terbesar, `<h2>` akan semakin mengecil dan seterusnya.

## Link

Anda tentu sering mengklik suatu tulisan atau gambar pada website yang menuju ke halaman lainnya. Yang anda klik itu adalah link. Link merupakan teks atau gambar yang membuka

halaman website lainnya. Untuk membuat link kita dapat menggunakan tag <a></a>. Seperti halnya tag font tag a juga memiliki attribut. Tabel 1.2 akan menunjukkan pada anda daftar attribut pada tag a.

**Tabel 1.2** Daftar attribut dari tag a

Attribut	Keterangan
href	Halaman website yang akan dituju
target	Dibuka pada window baru atau tidak
style	Penambahan CSS
class	Nama class yang dipanggil
name	Nama dari link

Contoh jika kita ingin membuka www.google.com pada window yang sama.

```
Google
```

Link diatas akan membuka alamat www.google.com dengan teks Google sebagai teks yang akan diklik.

### PENERAPAN TEORI MANIPULASI FONT DAN LINK

Jalankan PHP DESIGNER 2006 klik menu **File-New-HTML/XHTML**. Lalu ketik kode berikut

```
<html>
<head>
<title>Manipulasi Font dan Link</title>
</head>
<body>
<p>Pengumuman</p>
<p>Silahkan klik link berikut untuk menuju halaman pertama.html</p>
Halaman pertama.html

Halaman pertama.html (new window)
</body>
</html>
```

Simpan dengan nama **font\_link.html**. Jalankan pada browser IE arahkan address ke [http://localhost/webmaster/Bab1/hari-1/font\\_link.html](http://localhost/webmaster/Bab1/hari-1/font_link.html) hasilnya akan seperti gambar 1.1



Gambar 1.2: output file font\_link.html

Karena file *pertama.html* berada satu direktori dengan file *font\_link.html*. Maka isi dari href cukup kita tulis "pertama.html". Untuk link yang ke dua jika kita klik maka akan muncul window baru. Itu dikarenakan pada attribut target kita isi dengan "\_blank".

## Membuat Tabel

Tabel memiliki peranan yang cukup penting dalam pendesainan suatu tampilan. Dengan menggunakan tabel kita dapat dengan mudah menentukan letak suatu objek. Untuk membuat tabel digunakan `<table></table>`. Atribut dari tag table boleh dibilang cukup banyak. Daftar atribut tag table dapat anda lihat pada tabel 1.3.

**Tabel 1.3** Daftar atribut dari tag table

Atribut	Keterangan
width	untuk mengatur lebar tabel (% atau pixel)
height	untuk mengatur tinggi tabel
border	untuk menentukan tebal bingkai
cellpadding	menentukan jarak padding antar cell
cellspacing	menentukan jarak spacing antar cell
name	untuk menentukan nama tabel
bgcolor	untuk mengatur warna background
background	untuk menampilkan gambar sebagai background
align	letak teks secara horizontal (rata kiri, tengah atau kanan)
valign	letak teks secara vertical (rata atas, tengah atau bawah)
style	untuk CSS
bordercolor	untuk mengatur warna bingkai

Jika kita hanya menggunakan tag `<table></table>` saja maka tabel tidak akan pernah terbentuk karena belum memiliki baris dan kolom. Untuk membuat baris kita gunakan tag `<tr></tr>` dan untuk membuat kolom digunakan tag `<td></td>`. Terdapat juga tag `<th></th>` tag ini digunakan untuk membuat header tabel. Ketiga tag tersebut memiliki atribut yang hampir sama. Lihat tabel 1.4 untuk melihat daftar atributnya.

**Tabel 1.4** Daftar atribut tag tr, td, dan th

Atribut	Keterangan
height	untuk mengatur tinggi
bgcolor	untuk mengatur warna background
background	untuk menjadikan gambar sebagai background
align	untuk mengatur letak teks
valign	untuk mengatur letak teks secara vertical
colspan	untuk menghilangkan sejumlah kolom
rowspan	untuk menghilangkan sejumlah baris

Contoh dibawah ini mengilustrasikan kita akan membuat tabel dengan dua baris dan satu kolom.

```
<table border="1">
<tr>
 <td>Baris 1 kolom 1</td></tr>
<tr>
 <td>Baris 2 kolom 1</td></tr>
</table>
```

### PENERAPAN TEORI MEMBUAT TABEL

Seperti biasa jalankan PHP DESIGNER 2006. Klik **File-New-HTML/XHTML** untuk membuat file HTML baru. Kemudian ketik kode dibawah ini

```
<html>
<head>
<title>Membuat Tabel</title>
</head>
```

```

<body>
<h2>Daftar Harga</h2>
<table border="1" style="border-collapse: collapse" width="100%">
<tr bgcolor="#cccccc">
<th>Nama Barang</th><th>Harga Satuan</th><th>Jumlah Barang</th>
<th>Total</th>
</tr>
<tr align="center">
<td>CD R</td><td>Rp. 2.000,-</td><td>5</td><td>Rp. 10.000,-</td>
</tr>
<tr align="center">
<td>CD RW</td><td>Rp. 5.000,-</td><td>4</td><td>Rp. 20.000,-</td>
</tr>
<tr align="center">
<td colspan="3">TOTAL >>> </td><td>Rp. 30.000,-</td>
</tr>
</table>
</body>
</html>

```

Simpan dengan nama **tabel.html**. Jika anda jalankan pada Internet Explorer maka hasilnya akan terlihat seperti gambar 1.2.



Gambar 1.2: output file tabel.html

Coba perhatikan pada tag td yang ada pada baris ke tiga. Disini kita menuliskan attribut colspan="3" yang artinya kita menghilangkan 3 kolom dan menggantinya hanya dengan satu kolom.

## CSS (Cascading Style Sheet)

Dari namanya saja sudah kerennya pasti menarik. Tentu ini menarik tapi pada ebook ini kita hanya bahas secara garis besarnya saja. CSS merupakan suatu teknologi yang digunakan untuk mempermudah pembuatan suatu tampilan website. Kode dari CSS diapit oleh tag `<style></style>`. Tag style diletakkan diantara tag `<head></head>`. Selain dituliskan dalam tag style CSS juga dapat ditulis pada tag yang mendukung attribut `style` seperti `table`.

Setiap tag html seperti `<b>`, `<i>`, `<u>` dan sebagainya dapat dimanipulasi dengan CSS. Didalam CSS tag-tag HTML tersebut disebut *selector* dan atribut didalamnya disebut *declaration*. Declaration ditandai dengan kurung kurawal `{...}`. Contoh berikut akan membuat seluruh teks pada body berjenis huruf verdana dengan ukuran 11 pixel.

```
body{font-family: verdana; font-size: 11px}
```

### PENERAPAN TEORI CSS

Jalankan PHP DESIGNER 2006 klik menu **File-New-HTML/XHTML**. Lalu ketik script berikut ini.

```

<html>
<head>
<title>CSS - Cascading Style Sheet</title>
<style>
 body{font-family: verdana; font-size: 11px}
 b{color: red}
 i{color: green}
</style>
</head>
<body>
<p>Sekarang semua tulisan verdana 11 px</p>
Saya otomatis merah

<i>Saya otomatis hijau</i>
</body>
</html>

```

Simpan dengan nama ***css.html***. Jalankan pada browser IE maka hasilnya akan seperti gambar 1.3.



Gambar 1.3: output file *css.html*

Dapat anda lihat kalau seluruh teks pada body dengan sendirinya bertipe verdana dan berukuran 11 pixel karena adanya CSS. Demikian juga untuk tag b yang otomatis warnanya menjadi merah setelah deklarasikan di CSS.

## Form HTML

Seperti namanya form yang berarti isian atau formulir. Form HTML menyediakan input bagi user untuk mengisikan suatu data guna diproses oleh PHP. Membuat form HTML boleh dibilang sangatlah mudah. Karena tidak membutuhkan keahlian programming.

### Bentuk Dasar Form HTML

Setiap form pada HTML akan diapit oleh tag **<form>** dan **</form>**. Tag **<form>** memiliki beberapa attribut diantaranya:

- **Action** – attribut ini berfungsi untuk memberitahu file mana yang akan digunakan untuk memproses form yang dikirim. Contoh "fileku.php" atau "folderku/fileku.php" atau URL lengkap <http://www.domainku.com/fileku.php>.
- **Method** – attribut ini berfungsi untuk menentukan tipe pengiriman data. Bisa berupa POST atau GET.
- **Name** – attribut ini berguna untuk alamat yang merupakan bagian dari javascript. Nama form tidak dikirim ke server saat form *disubmit* atau dikirim.

Contoh lengkap penulisan form sebagai berikut:

```
<form name="formku" action="proses.php" method="POST">
...
</form>
```

## Macam Input Type

Sebagian besar penulisan form akan selalu digunakan *input type*. Tag *input* dan attributnya yaitu *type* akan menentukan output yang ditampilkan oleh browser. Bisa berupa tombol, check box, password box dan sebagainya. Untuk lebih memperjelas lihatlah contoh di berikut

```
<form action="tidak Ada.php" method="POST">
<input type="text" name="coba" size="30" maxlength="45">
<input type="submit" value="GO! ! ">
</form>
```

Output dari kode diatas akan terlihat seperti gambar 1.3.



Gambar 1.3: Contoh sederhana form HTML

Setiap *type* juga memiliki attribut sendiri-sendiri. Tapi setiap *type* pasti memiliki attribut *name*. Berikut ini beberapa tipe *type* yang dapat anda gunakan:

- **Text** – tipe ini seperti yang terlihat di gambar yaitu berupa textbox. Text memiliki tiga attribut diantaranya:
  - **Size** – yang merupakan ukuran panjang dari textbox.
  - **maxlength** – merupakan karakter maksimal yang dapat ditampung oleh textbox.
  - **Value** – merupakan isi dari textbox yang telah ditetapkan. User dapat menggantinya dengan langsung mengetikkan pada textbox.
- **Password** – hampir sama dengan text hanya saja karakter yang diketikkan akan berupa asterik (\*).
- **Hidden** – input ini tidak dimunculkan di layar browser. Dapat digunakan untuk menyimpan nilai atau variabel yang digunakan untuk halaman berikutnya.
- **Submit** – tipe ini akan menampilkan tombol submit(pengiriman). Nama tombol diambil dari nilai attribut value. Jika tidak ada maka namanya *Submit Query*.
- **Reset** – tipe ini digunakan untuk membersihkan segala isian pada form. Attributnya hampir sama dengan submit.
- **File** – tipe ini digunakan untuk proses peng-upload-an file. Tampilannya hampir sama dengan tipe text tapi disebelahnya ada tombol "browse". Ketika user menekan browse akan muncul kotak dialog untuk memilih file yang diupload.
- **Checkbox** – tipe ini akan memunculkan tampilan checkbox. Biasanya checkbox digunakan untuk memilih lebih dari satu pilihan. Jika pada tag ini terdapat kata "checked" maka checkbox tersebut akan ditandai.
- **Radio** – tipe ini akan menampilkan tampilan radio button. Tipe ini digunakan jika kita harus memilih hanya satu pilihan. Jika ada kata "checked" maka radio tersebut akan ditandai(default).

- **Image** – fungsinya sama dengan submit hanya saja anda dapat menentukan gambar sebagai pengganti tombol.

Kode berikut ini mencontohkan penggunaan macam-macam *type* tersebut kecuali *File* dan *Image*.

```
<form action="tidak Ada.php" method="POST">
<h2>Formulir Pendaftaran</h2>
Nama: <input type="text" name="nama" value="Ketik Nama Disini" size="30" maxlength="50">

Password: <input type="password" name="nama" size="30" maxlength="50">

<input type="checkbox" name="mailing" value="ya"> Jika dicek anda bersedia menjadi anggota mailing list kami.
<input type="hidden" name="sembunyi" value="anda tidak dapat melihatku">
<p>
Status Anda?

<input type="radio" name="status" value="menikah">Sudah Menikah

<input type="radio" name="status" value="belum">Belum Menikah

<input type="radio" name="status" value="duda_janda">Duda/Janda

</p>
<input type="submit" value="DAFTAR"> <input type="reset" value="RESET">
</form>
```

Output dari kode diatas akan seperti gambar 1.4.



Gambar 1.4: Penggunaan macam-macam type

## Select, Multiple Select

Tag select akan menampilkan pilihan dalam kotak *drop-down*. Untuk membuatnya anda harus menuliskan tag `<select>` dengan attributnya yaitu name. Didalam tag select kita memerlukan tag `<option>` untuk membuat pilihan. Pilihan option juga memiliki attribut yaitu value.

Contoh penggunaan tag `<select>`:

```
<form action="tidak ada.php" method="POST">
<select name="status">
 <option value="menikah">Sudah Menikah
 <option value="belum">Belum Menikah
```

```
<option value="duda_janda">Duda/Janda
</select>
</form>
```

Status Anda?

Sudah Menikah
<b>Sudah Menikah</b>
Belum Menikah
Duda/Janda

Gambar 1.5: contoh output select

Jika kita tambahkan kata *multiple* pada tag select. Maka user diperbolehkan memilih lebih dari satu pilihan atau sering disebut *listbox*. Lihat contoh berikut akan menampilkan listbox 3 elemen awal dengan mengisi attribut size. Untuk memilih lebih dari satu gunakan tombol SHIFT atau Ctrl.

```
<form action="tidak ada.php" method="POST">
Makanan yang anda suka?

<select name="makanan" multiple size="3">
<option value="Nasi_goreng">Nasi goreng
<option value="Mie_ayam">Mie Ayam
<option value="Burger">Burger
<option value="Kentang">Kentang
</select>
</form>
```

Makanan yang anda suka?

<b>Nasi goreng</b>
Mie Ayam
Burger

Gambar 1.6: contoh output multiple select

## Textarea

Tag <textarea> akan menciptakan sebuah textbox yang besar yang mempunyai baris dan kolom. Berbeda dengan elemen form lainnya, textarea memiliki tag penutup yaitu </textarea>. Untuk memperjelas lihatlah contoh berikut.

```
<form action="tidak_ada.php" method="POST">
Buatlah deskripsi singkat tentang diri anda

<textarea name="deskripsi" rows="8" cols="50">Tulis Deskripsi Disini</textarea>
</form>
```

Buatlah deskripsi singkat tentang diri anda

Gambar 1.7: contoh output textarea

Untuk pembahasan lebih detail tentang HTML anda dapat membaca e-book "**Mastering Kode HTML**". Anda dapat mendownload "**Mastering Kode HTML**" di alamat berikut:

<http://materi.rahasia-webmaster.com/>

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7



## Hari Ke-2

Karena sudah hari kedua ikuti langkah-langkah berikut:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab1
- Buat folder baru, berinama folder tersebut **hari-2**

# PHP

Untuk hari yang ke dua ini kita akan mempelajari dasar-dasar dari PHP. Mulai dari pengertian tentang variabel, struktur pengambilan keputusan, dan dasar-dasar dari penguasaan tentang bahasa pemrograman. Pada bab ini kita juga akan membuat sebuah proyek kecil untuk mengaplikasikan teori yang telah kita dapat. Contoh-contoh yang saja sajikan dalam e-book ini hampir semuanya merupakan contoh dari aplikasi web sebenarnya. Sehingga anda tahu bahwa sebenarnya web programming itu MUDAH!.

Pada hari kedua ini diharapkan anda dapat mengerti tentang:

- Variabel pada PHP
- Menempelkan(embed) PHP pada HTML
- Komentar pada PHP
- Struktur Kontrol (if...else)
- Operator Aritmatik

## Variabel Pada PHP

Variabel merupakan elemen yang sangat penting dalam bahasa pemrograman. Hampir setiap bahasa pemrograman mengenal apa itu yang namanya variabel. Variabel sendiri merupakan suatu bentuk penyimpanan data sementara pada memori komputer yang akan diolah lebih lanjut.

Contoh variabel pada PHP

```
$nama = 'Rio Astamal';
$website = 'www.rahasia-webmaster.com';
$sangka_1 = 3;
$sangka_2 = 4;
$hasil = $sangka_1 + $sangka_2;
```

Variabel pada PHP memiliki tanda khusus yaitu diawali dengan tanda \$ (dollar). Penggunaan tanda ini sama dengan yang ada di bahasa perl. Seperti terlihat pada contoh diatas variabel dapat berisi string(huruf) atau angka. Sebuah variabel dapat juga berisi variabel lain, contohnya \$hasil = \$sangka\_1 + \$sangka\_2;. Dalam membuat variabel ada beberapa ketentuan yang harus dipenuhi agar variabel itu valid. Dan menurut saya adanya tanda \$ memudahkan kita untuk menemukan kesalahan.

Ketentuan-ketentuan dalam membuat variabel:

- Variabel dapat terdiri dari huruf, angka dan underscore( \_ ) dan tentu saja tanda dollar \$.
- Variabel tidak dapat diawali dengan angka.
- Variabel bersifat *case sensitive* artinya membedakan huruf kecil dan huruf besar. Variabel \$nama tidak sama dengan \$NaMa. Untuk itu hati-hati dalam penulisan nama variabel.

## Tipe Variabel

Variabel juga memiliki banyak tipe. Tipe variabel mengindikasikan jenis data apa yang disimpan didalamnya.

## Macam-macam Tipe Data

PHP mendukung tipe-tipe variabel berikut ini:

**Tabel 1.5** Daftar tipe data pada PHP

Tipe Data	Keterangan
Integer	Digunakan untuk semua angka
String	Digunakan untuk semua huruf, angka, spasi, dan simbol
Double	Digunakan untuk bilangan real
Boolean	Digunakan untuk nilai True atau False
Array	Digunakan untuk menampung beberapa data sekaligus (akan dibahas lebih lanjut)
Object	Digunakan untuk class

## Konstanta

Pada variabel kita diizinkan untuk mengubah nilainya. Ini berbeda dengan konstanta. Nilai dari konstanta hanya boleh diset satu kali saja dan tidak bisa diubah dimanapun didalam script.

```
$var1 = 2000;
$var1 = 4000;
```

Pada contoh diatas nilai awal dari variabel \$var1 adalah 2000. Namun kemudian nilainya diubah menjadi 4000. Inilah perbedaan utama variabel dengan konstanta.

Anda dapat mendeklarasikan konstanta dengan menggunakan fungsi *define*.

```
define('HRG_BENSIN', 2400);
define('HRG_BENSIN', 20000);
```

Perhatikan, berbeda dengan varibel nama konstanta terdiri dari huruf KAPITAL. Ini sama dengan bahasa C yang memudahkan untuk membedakan variabel dengan konstanta. Satu hal yang penting lagi konstanta tidak memiliki tanda \$ seperti yang dimiliki variabel.

Jadi jika kita ingin mencetak nilai konstanta ke layar maka anda harus menuliskan persis seperti nama konstanta tersebut.

```
echo $var1;
echo HRG_BENSIN;
```

Hasil pada layar adalah \$var1 akan menghasilkan output 4000 dan HRG\_BENSIN akan menghasilkan 2400.

## Jangkauan(Scope) Variabel

Jangkauan variabel merupakan tempat di script dimana variabel tersebut masih dapat dipanggil. Bingungkan? Baca lagi dan lanjutkan. Ada beberapa tipe scope variabel diantaranya:

- Variabel built-in bersifat **super-global** artinya dapat dipanggil dimanapun didalam script.
- Variabel **global** dapat dipanggil dari manapun kecuali didalam suatu fungsi.
- Variabel yang digunakan pada fungsi hanya dapat dikenali didalam fungsi itu sendiri.
- Variabel yang digunakan didalam fungsi yang dideklarasikan sebagai global akan mengacu pada variabel global dari nama yang sama.

Nantinya anda akan sering bergaul dengan variabel superglobal seperti `$_GET` dan `$_POST`. Dibawah ini adalah daftar dari variabel ***super-global***. Variabel – variabel berikut ini dapat dipanggil darimana saja didalam script, entah itu didalam atau diluar suatu fungsi.

- `$GLOBALS` – array dari semua variabel global
- `$_SERVER` – array dari variabel lingkungan server
- `$_GET` – array dari variabel yang dikirim melalui method="GET"
- `$_POST` – array dari variabel yang dikirim melalui method="POST"
- `$_COOKIE` – array dari variabel cookie
- `$_FILES` – array dari variabel file yang diupload
- `$_ENV` – array dari lingkungan(environment) variabel
- `$_REQUEST` – array dari semua input variabel dari user
- `$_SESSION` – array dari variabel session

Dari keseluruhan variabel super-global yang ada diatas. Ada beberapa yang akan sangat sering kita gunakan. Seperti `$_POST`, `$_COOKIE`, `$_SESSION` variabel-variabel ini memiliki peran "penting" dalam script kita nantinya.

### PENERAPAN TEORI VARIABEL

Jalankan PHP DESIGNER 2006 klik menu **File > New > PHP** untuk membuat file PHP baru. Kemudian etik script berikut ini agar memori diotak anda menangkap dengan lebih baik. Wis.... kayak spesialis otak saja!?!.

Ehm... sebenarnya begini..., eeh... sebelum menjadi webmaster dulu...saya sempat....?!? Ah.... udah...udah.... banyak omong.

```
<html>
<head>
<title>Variabel</title>
</head>
<body>

<?php
$nama = 'Nama: Mr. Web Janarko
';
$website = "Website: www.wejan.com
";
echo $nama;
echo $website;
?>

</body>
</html>
```

Simpan script diatas dengan nama ***variabel.php*** agar kita tahu bahwa yang sedang kita kerjakan untuk kali ini hanya masalah variabel. Jalankan pada Internet Explorer atau browser lainnya arahkan ke `http://localhost/webmaster/Bab1/hari-2` klik ***variabel.php***. Jika anda melakukannya dengan benar hasilnya akan tampak seperti gambar 1.8.

### PENJELASAN SCRIPT

Perhatikan bagian script yang diantara tanda `<?php` dan `?>`. Script yang beraada diantara tanda tersebut itu merupakan sebuah script PHP. Script PHP selalu diawali dengan tanda `<?php` dan diakhiri dengan tanda `?>`. Untuk memisahkan statemen PHP satu dengan yang lainnya digunakan tanda `";`(titik koma).

Jadi tulisan yang berada dalam tag `<?php ... ?>` akan dianggap kode PHP oleh server. Itulah yang disebut menempelkan (embed) PHP pada HTML.

Mungkin anda bertanya, apa perbedaan menggunakan tanda petik satu(') dan petik dua(")? . Anda tidak dapat meletakkan variabel didalam ('), variabel didalam petik satu tidak akan dianggap dan tetap ditampilkan apa adanya. Anda juga tidak dapat meletakkan karakter spesial seperti `\n` (garis baru), `\t` (tab) dan sebagainya pada petik satu.



Gambar 1.8: output file variabel.php

Jika anda menjalankan perintah berikut.

```
echo '$nama';
```

hasil yang akan ditampilkan pada browser hanya \$nama. Sebaliknya pada tanda petik dua( " ) anda dapat meletakkan variabel didalamnya.

Jika anda menggunakan petik dua pada kode.

```
echo "$nama";
```

output yang akan ditampilkan pada browser adalah isi dari variabel \$nama. Yang tidak lain adalah Nama: Mr. Web Janarko.

Sekarang coba klik kanan dokumen tersebut lalu klik **view source**. Jika anda mencari bagian PHP kodennya jelas tidak ada yang ada hanya Nama: Mr. Web Janarko<br>Website: www.wejan.com<br>. Itulah yang disebut bahasa **server-side scripting**. Kode PHP dieksekusi di server yang dikirim ke klien hanyalah hasil akhirnya saja yaitu berupa kode HTML murni.

Echo merupakan perintah pada PHP yang digunakan untuk mencetak output berupa string ke browser. Perintah yang ekuivalen dengan echo adalah print.

## Komentar Pada PHP

Komentar pada script bertujuan untuk memberitahu pembacanya, baik orang lain ataupun anda sendiri. Biasanya komentar digunakan untuk menjelaskan tujuan dituliskannya script tersebut, siapa penulisnya, kapan ditulis dan sebagainya. Komentar juga berguna bagi anda sendiri ketika suatu waktu anda lupa mengapa anda menulis file ini, fungsi-fungsinya untuk apa dan banyak lainnya.

PHP akan mengabaikan semua text didalam komentar. Jadi tidak akan berpengaruh pada jalannya suatu script.

PHP mengenal tiga jenis gaya komentar. Pertama **model bahasa C** (banyak baris/multi line), biasanya dapat anda letakkan pada bagian atas script. Lihatlah script dibawah ini.

```
<?php
/*
NamaFile: variabel.php
Penulis: Mr. Web Janarko
```

---

```
Fungsi: untuk mengetes variabel pada PHP
*/
?>
```

Pada model ini komentar diawali tanda /\* dan diakhiri tanda \*/.

Yang kedua anda dapat menggunakan **model C++**. Model ini hanya bisa digunakan pada satu baris saja.

```
echo $nama; // mencetak nama
```

atau anda dapat menggunakan **model shell script**.

```
echo $website; # mencetak website
```

Pada kedua model ini, segala text setelah tanda // atau # akan dianggap komentar oleh PHP sampai akhir dari baris tersebut. Coba anda modifikasi file variabel.php yang telah anda buat dan tambahkanlah komentar seperti contoh diatas.

## Struktur Kontrol

Struktur Kontrol merupakan sebuah struktur dalam bahasa pemrograman yang membolehkan kita untuk mengontrol alur dari eksekusi suatu program atau script. Struktur kontrol mencakup struktur kondisi dan struktur pengulangan atau **looping**. Namun karena ini hari pertama saya hanya memberikan struktrur kondisi **if...** dan **if...else...** saja. Kalau kebanyakan nanti malah pusing, mual lalu muntah.

### Statemen if...

Kita dapat menggunakan statemen if untuk membuat sebuah keputusan. Anda harus memberi if sebuah kondisi untuk membuat keputusan. Jika kondisi bernilai *true* maka blok if akan dieksekusi. Bingung? Tenang sebentar lagi kebingungan anda hilang dengan melihat contoh. Statemen if dikelilingi oleh tanda { } (kurung kurawal).

```
$nama = 'Web Janarko';

if ($nama == 'Web Janarko')
{
 echo $nama;
}
```

pada cuplikan script diatas kita menggunakan nilai variabel \$nama sebagai kondisi. Dimana jika nilai variabel \$nama sama dengan Web Janarko maka kondisi bernilai *true* maka blok if akan dieksekusi.

Tanda (=) berbeda dengan (==), tanda == digunakan untuk mengetes suatu nilai dari variabel apakah sama. Sedangkan = digunakan untuk memberi nilai pada variabel.

### Statemen if...else...

Statemen if juga memiliki banyak variasi salah satu diantaranya **if...else....** Anda akan sering menggunakan statemen ini karena dengan statemen ini anda dapat memilih kondisi mana yang ingin anda eksekusi.

Seperti artinya *else* yang dalam bahasa indonesia dapat diartikan *selain itu*. Statemen else mengijinkan kita untuk menentukan aksi alternatif yang diambil jika dalam statemen if bernilai false.

```

$sisa_uang = 6000;
if ($sisa_uang < 5000)
{
 echo "Maaf bung, sisa uang anda tidak cukup, sisa uang anda sekarang sebesar
 Rp. $sisa_uang,-.";
}
else
{
 echo "Silahkan membeli karena sisa uang anda masih Rp. $sisa_uang,-.";
}

```

Saya kira anda sudah bisa menebak alur script tersebut. Pertama kita sudah mendefinisikan variabel \$sisa\_uang = 6000. Pada statemen if yang pertama, jika \$sisa\_uang kurang dari 5000 maka true. Sedangkan sisa uang kita masih 6000 maka nilainya false oleh sebab itu blok statemen else dijalankan. Sekedar mengingatkan karena menggunakan " (petik dua) maka variabel dapat langsung kita masukkan dalam tanda petik tersebut.

### PENERAPAN TEORI IF... DAN IF...ELSE...

Salah satu tugas terpenting dan paling sering dilakukan bahasa server side scripting adalah memproses HTML Form. Pada contoh ini adalah inti awal anda memahami tugas PHP sebagai pemroses HTML Form. Kita akan membuat dua file yang pertama adalah file HTML murni sebagai input data bagi user(HTML Form). Dan file yang kedua adalah file PHP yang bertugas memproses data yang dikirim oleh file HTML.

Untuk itu jalankan PHP DESIGNER 2006 lalu klik menu **File-New-HTML/XHTML**. Ketik script berikut ini agar mata anda tidak jemu membaca e-book.

```

<html>
<head>
<title>MiniMarket Janarko</title>
</head>
<body>

MiniMarket Online - Pembelian<hr>
<p>
<form action="proses_beli.php" method="post">
<table border="0">
<tr bgcolor="#cccccc">
 <td>Barang</td>
 <td>Harga</td>
 <td>Jumlah</td></tr>
<tr>
 <td>Baterei CBA</td>
 <td>Rp. 7500</td>
 <td><input type="text" name="baterei" maxlength="3" size="3"></td></tr>
<tr>
 <td>Baygin</td>
 <td>Rp. 2250</td>
 <td><input type="text" name="baygin" maxlength="3" size="3"></td></tr>
<tr>
 <td>Silver King</td>
 <td>Rp. 1750</td>
 <td><input type="text" name="silver" maxlength="3" size="3"></td></tr>
<tr>
 <td colspan="3"><input type="submit" value="B e l i"></td></tr>
</table>
</form>

</body>
</html>

```

Simpan script diatas dengan nama **minimarket.html** dan jangan lupa tetap letakkan di **C:\Apache2\htdocs\webmaster\Bab1\hari-2\**. Karena kita masih di skop hari kedua. Lihat gambar 1.9 untuk output script HTML diatas.



Gambar 1.9: output file minimarket.html

Sekarang buatlah file PHP-nya, klik **File-New-PHP**. Silahkan ketik script berikut ini atau dilihat-lihat saja jika anda sudah paham.

```
<?php
/*
 NamaFile: proses_beli.php
 Penulis: Mr. Web Janarko
 Fungsi: untuk memproses file minimarket.html
*/

//buat variabel untuk menampung data dari minimarket.html
$baterei = $_POST['baterei'];
$baygin = $_POST['baygin'];
$silver = $_POST['silver'];

buat konstanta harga masing-masing produk
define('HARGA_BTR', 7500);
define('HARGA_BYN', 2250);
define('HARGA_SLR', 1750);

$baterei_ttl = $baterei * HARGA_BTR;
$baygin_ttl = $baygin * HARGA_BYN;
$silver_ttl = $silver * HARGA_SLR;
$total = $baterei_ttl + $baygin_ttl + $silver_ttl;

if ($total < 50000)
{
 $diskon = 0; //diskon 0%
}
else
{
 $diskon = 0.15; //diskon 15%
}
$subtotal = $total - ($total * $diskon);
?>
```

```

<html>
<head>
<title>Kasir Online - MiniMarket Janarko</title>
</head>
<body>
Pembelian Terproses<hr>
<p>Berikut ini faktur pembelian anda</p>

<table border="0">
<tr bgcolor="#cccccc">
<td align="center">Barang</td>
<td align="center">Jumlah Beli</td>
<td align="center">Bayar</td></tr>

<?php
echo '<tr>';
echo '<td >Baterei CBA</td>';
echo '<td align="center">' . $baterei . '</td>';
echo '<td>' . $baterei_ttl . '</td></tr>';
echo '<tr>';
echo '<td>Baygin</td>';
echo '<td align="center">' . $baygin . '</td>';
echo '<td>' . $baygin_ttl . '</td></tr>';
echo '<tr>';
echo '<td>Silver King</td>';
echo '<td align="center">' . $silver . '</td>';
echo '<td>' . $silver_ttl . '</td></tr>';
echo '</table>';
echo "<p>TOTAL: $total
" ;
echo "SUBTOTAL SETELAH DISKON: $subtotal</p>";
?>

</body>
</html>

```

Simpan dengan nama **proses\_beli.php**, letakkan di direktori yang sama dengan minimarket.html. Sekarang buka alamat <http://localhost/webmaster/Bab1/hari-2/minimarket.html>.

Coba isi field-field jumlah lalu klik beli jika script yang anda tulis benar maka hasilnya akan terlihat seperti gambar 1.10

### PENJELASAN SCRIPT

Waduh saya sendiri bingung menjelaskannya? Ha..bercanda!. OK kita mulai dari file minimarket.html dulu. Yang perlu saya tekankan adalah semua komponen dalam tag **<form>...</form>**. Karena data dalam tag inilah yang akan diproses oleh PHP. Perhatikan cuplikan script berikut

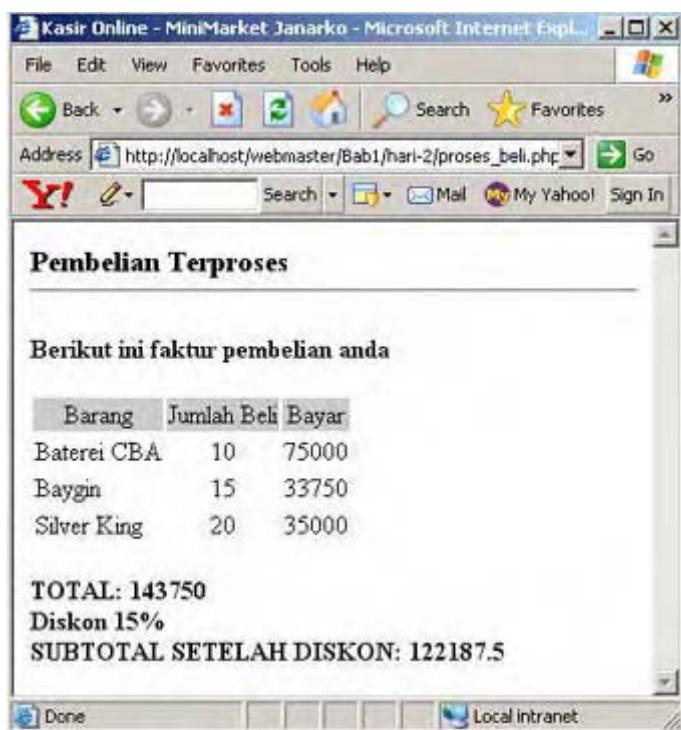
```
<form action="proses_beli.php" method="post">
```

pada script minimarket.html file php yang "diberi tugas" untuk memproses data adalah file proses\_beli.php. Selain **method="post"** anda juga dapat menggunakan **method="get"**. Satu hal lagi yang sangat penting dan mendasar adalah penamaan variabel yang akan diproses.

```
<input type="text" name="baterei" maxlength="3" size="3">
```

arti dari cuplikan script diatas adalah kita akan mempost(mengirim) data dari textbox dengan nama **baterei**. Untuk menampung data yang dikirim tersebut kita harus menyediakannya sebuah tempat yang "nyaman" yaitu di memori komputer dalam hal ini **variabel**.

```
$baterei = $_POST['baterei'];
```



Gambar 1.10: Output file proses\_beli.php

kode diatas artinya menampung data yang dikirim dari metode POST. Isi dari `$_POST[...]` harus sama dengan nama data yang dikirimkan dalam hal ini `baterei`. Sedangkan untuk nama variabel **\$baterei tidak harus sama** dengan nama data yang dikirim. Namun untuk memudahkan pemrograman, saya sarankan agar nama variabel sama dengan nama data yang kirim.

Karena harga barang bersifat tetap dan tidak boleh berubah maka sebaiknya kita gunakan konstanta dibanding variabel. `define('HARGA_BTR', 7500)`, seperti yang pernah dibahas sebelumnya. Untuk membuat konstanta digunakan perintah `define(nama_konstanta, isi_konstana)`. Konstanta `HARGA_BTR` merupakan harga baterei CBA yaitu 7500.

```
$baterei_ttl = $baterei * HARGA_BTR;
.....
$total = $baterei_ttl + $baygin_ttl + $silver_ttl;
```

Saya kira tidak perlu dijelaskan, sudah jelas kan? Oh... perlu dijelaskan? Baik. Variabel `$baterei_ttl` adalah variabel yang menyimpan hasil perkalian (\*) antara isi variabel `$baterei` dengan konstanta `HARGA_BTR`. Yang merupakan jumlah harga baterei CBA sejumlah x unit. `$total` sudah dapat anda tebak kalau variabel tersebut menyimpan semua total harga barang mulai dari batereri, baygin dan silver.

```
if ($total < 50000)
{
 $diskon = 0; //diskon 0%
}
else
{
 $diskon = 0.15; //diskon 15%
}
$subtotal = $total - ($total * $diskon);
```

Pada statemen if diatas jika pengunjung berbelanja kurang dari 50000. Maka ia tidak mendapat diskon sebaliknya jika pengunjung berbelanja lebih dari 50000 maka ia dapat diskon 15% (0.15 dalam desimal). Seperti yang pernah kita pelajari kita dapat mengambil keputusan dari dua kondisi dengan menggunakan statemen `else`. Variabel `$subtotal` merupakan hasil dari total seluruh harga dikurangi perkalian total keseluruhan harga dengan diskonnya. Agar

PHP tidak keliru untuk menafsirkan keinginan kita berilah tanda ( ) didalam perhitungan. Seperti matematika SD saja...!.

```
echo '<td align="center">'.$baterei.'</td>';
echo '<td>'.$baterei_ttl.'</td></tr>';
.....
echo "<p>TOTAL: $total
" ;
echo "SUBTOTAL SETELAH DISKON: $subtotal</p>" ;
```

pada baris yang pertama variabel \$baterei kita masukkan pada kolom ke-2 pada tabel. Nanti secara otomatis nilai dari \$baterei akan diganti dengan angka yang dimasukkan user pada form pembelian. Perlu diperhatikan tanda “.” (titik) digunakan untuk menggabungkan string. Nilai \$baterei\_ttl akan berubah sesuai dengan perkalian dari \$baterei dan HARGA\_BTR. Karena \$total berada dalam tanda petik “ ” (petik dua) maka kita tidak perlu menggunakan tanda “.” Untuk menggabung variabel dengan string.

PHP tidak memperdulikan tata letak kode anda, desainlah kode anda untuk semudah mungkin dapat dibaca dengan nyaman. Terutama pada blok statemen if dan sebagainya. Ini juga akan mempermudah anda dalam mencari suatu kesalahan(bug) pada kode anda.

## Operator

Operator merupakan simbol yang dapat digunakan untuk memanipulasi nilai dan variabel. Pada bagian sebelumnya kita sudah menggunakan beberapa operator diantaranya =, ==, <, ., +, \* dan lainnya. Selanjutnya kita kan membahas operator-operator ini dan operator lainnya lebih detil lagi.

### Operator Aritmatik

Operator aritmatik sudah sangat umum anak SD kelas 1 pun sudah bisa menggunakan operator ini. Operator artimatik tidak ada bedanya dengan operator matematika biasa. Daftar dari operator aritmatik ditunjukkan pada tabel 1.5

**Tabel 1.5 – Operator Aritmatik pada PHP**

Operator	Nama	Contoh
+	Penambahan	\$a + \$b
-	Pengurangan	\$a - \$b
*	Perkalian	\$a * \$b
/	Pembagian	\$a / \$b
%	Modulus	\$a % \$b

Untuk operator aritmatik ini yang perlu sedikit saya jelaskan adalah tentang modulus. Nggak usah... saya ini pakar matematika? Ya.. udah. Hei kami belum tahu!. Karena kita negara demokratis maka suara terbanyak yang kita ambil jadi saya akan menjelaskan tentang modulus.

Modulus merupakan sisa pembagian dari dua bilangan.

```
$a = 11;
$b = 2;
$hasil = $a % $b;
```

nilai dari variabel \$hasil adalah 1. Jika \$a kita ganti dengan 15 dan \$b dengan 4 maka \$hasil = 4;

## Operator String

Saya rasa saya perlu sedikit menjelaskan tentang operator ini. Anda dapat menggunakan tanda penghubung “.” untuk menggabungkan dua string. Seperti yang sudah pernah saya singgung sebelumnya. Kita dapat menggunakan tanda petik satu ( ‘ ) atau petik dua ( “ ). Jika anda ingin menampilkan output yang terdapat tulisan tanda “ maka sebaiknya gunakan tanda ‘ untuk mengapit string anda begitu pula sebaliknya.

```
$a = "Do'aKu ";
$b = 'Selalu Menyertaimu';
$c = $a.$b;
```

Output yang akan ditampilkan oleh variabel \$c adalah ***Do'aKu Selalu Menyertaimu***. Sebagai alternatif anda juga dapat memberi tanda \ (backslash) jika ingin menampilkan tanda “ atau ‘ pada penggunaan tanda yang sama di variabel.

```
$a = 'Do\'aKu ';
$b = "Selalu Menyertaimu";
$c = $a.$b;
```

Output yang dihasilkan akan sama persis dengan yang sebelumnya.

Selain itu anda dapat menggabung string dengan cara demikian.

```
$a = "Do'aKu";
$a .= "Selalu Menyertaimu";
```

Saya kira untuk hari kedua sampai disini saja karena kalau terlalu banyak kepala anda malah pusing. Sekarang sebaiknya istirahat agar besok lebih fresh menerima materi berikutnya OK.

---

## Kalender

27	28	29	30	31	1	2	
3	4	5	6	7	8	9	
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

Karena kita sudah menginjak hari ke lakukan konfigurasi berikut

- C:\Apache2\htdocs\webmaster\Bab1
- Berilah nama folder baru tersebut **hari-3**.

## Operator, Struktur Kontrol, dan Looping

Pada hari yang ke tiga ini kita masih melanjutkan materi yang sebelumnya yaitu operator. Selain itu kita akan membahas statemen selain if...else..., seperti if...elseif..., switch... dan perulangan (looping) while..., for..., foreach... dan do...while.

Pada hari ke tiga ini anda diharapkan mengerti tentang:

- Operator Kombinasi
- Operator Perbandingan
- Operator Logika
- Struktur Kontrol (if...elseif, switch...)
- Perulangan (for..., foreach..., while..., do...while)

### Operator Kombinasi

Operator kombinasi merupakan gabungan dari operator aritmatik biasa dengan variabel itu sendiri. Saya sendiri bingung menjelaskannya, lihatlah contoh berikut ini.

```
$a += 5;
```

Kode diatas artinya sama dengan.

```
$a = $a + 5;
```

Contoh-contoh operator kombinasi dapat anda lihat pada tabel 1.6

**Tabel 1.6** Operator Kombinasi Pada PHP

Operator	Penggunaan	Sama dengan
+=	\$a += \$b	\$a = \$a + \$b
-=	\$a -= \$b	\$a = \$a - \$b
*=	\$a *= \$b	\$a = \$a * \$b
/=	\$a /= \$b	\$a = \$a / \$b
%=	\$a %= \$b	\$a = \$a % \$b
.=	\$a .= \$b	\$a = \$a . \$b

### Operator Perbandingan

Operator perbandingan digunakan untuk membandingkan dua nilai. Penggunaan ekspressi ini juga akan menghasilkan nilai true atau false tergantung dari perbandingan.

**Tabel 1.7** Operator Perbandingan Pada PHP

Operator	Nama	Penggunaan
==	sama dengan	\$a == \$b
===	identik	\$a === \$b
!=	tidak sama dengan	\$a != \$b
<>	tidak sama dengan	\$a <> \$b
<	kurang dari	\$a < \$b

>	lebih dari	\$a > \$b
<=	kurang dari atau sama dengan	\$a <= \$b
>=	lebih dari atau sama dengan	\$a >= \$b

## Operator Logika

Operator logika digunakan untuk mengkombinasikan hasil dari sebuah kondisi logika. Sebagai contoh, kita mungkin menemukan kasus nilai dari variabel \$a berada diantara 0 sampai 10. Untuk mengetes kondisi \$a >= 0 dan \$a <= 10, kita gunakan operator AND.

```
$a >= 0 && $a <= 10
```

PHP mendukung penggunaan operator AND, OR, XOR(eksklusif OR) dan NOT.

**Tabel 1.8** Operator Logika Pada PHP

Operator	Nama	Penggunaan	Hasil
!	NOT	!\$a	True jika \$a adalah false, juga sebaliknya
	OR	\$a    \$b	True jika \$a atau \$b atau keduanya true, selain itu false
&&	AND	\$a && \$b	True jika keduanya \$a dan \$b nilainya true, selain itu false
or	OR	\$a or \$b	sama seperti
and	AND	\$a and \$b	sama seperti &&

## Struktur Kontrol (lanjutan)

### Statemen if...elseif...

Jika statemen if...else... hanya dapat memakai dua kondisi. Statemen if...elseif... dapat memiliki lebih dari dua kondisi alias buanyak. Ini dapat kita gunakan jika kita menghadapi persoalan dimana kita harus membuat keputusan dari banyak kemungkinan.

```
if ($a == 0)
{
 eksekusi kode;
}
elseif ($a > 0)
{
 eksekusi kode;
}
else if ($a < 0)
{
 eksekusi kode;
}
```

anda dapat menulis statemen ini tanpa spasi seperti elseif atau dengan spasi else if. Keduanya sama-sama valid.

Anda tentu menyadari pada script yang pernah kita yaitu **proses\_beli.php**. Pada script tersebut terdapat kondisi dimana kita memberikan diskon pada pembeli. Namun karena hanya menggunakan if...else... maka script tersebut masih banyak kekurangan. Karena jika pembeli yang berbelanja lebih dari 50000 meski sangat besar diskonnya tetap sama yaitu 15%.

Masalah yang akan kita pecahkan adalah.

- Jika user tidak mengisi apa-apa
- Jika berbelanja kurang dari 25000 diskon 0%
- Jika berbelanja antara 25000 – 50000 diskon 5%

- Jika berbelanja antara 50000 – 75000 diskon 10%
- Jika berbelanja lebih dari 75000 diskon 15%

### PENERAPAN TEORI IF...ELSEIF...

Untuk mengaplikasikan statemen ini jalankan PHP DESIGNER 2006. Klik menu **File-Open** arahkan ke c:\Apache2\htdocs\webmaster\Bab1\hari-3\proses\_beli.php untuk membuka file proses\_beli.php

Ubahlah kode script tersebut sehingga menjadi seperti script dibawah ini.

```
<?php
/*
 NamaFile: proses_beli.php
 Penulis: Mr. Web Janarko
 Fungsi: untuk memproses file minimarket.html
*/

//buat variabel untuk menampung data dari minimarket.html
$baterei = $_POST['baterei'];
$baygin = $_POST['baygin'];
$silver = $_POST['silver'];

buat konstanta harga masing-masing produk
define('HARGA_BTR', 7500);
define('HARGA_BYN', 2250);
define('HARGA_SLR', 1750);

$baterei_ttl = $baterei * HARGA_BTR;
$baygin_ttl = $baygin * HARGA_BYN;
$silver_ttl = $silver * HARGA_SLR;
$total = $baterei_ttl + $baygin_ttl + $silver_ttl;

if ($total <= 0)
{
 echo "Anda tidak membeli apa-apa";
 exit;
}

if ($total > 0 && $total < 25000)
{
 $diskon = 0; //diskon 0%
 $pesan_diskon = 'Diskon 0%';
}
elseif ($total >= 25000 && $total <= 49000)
{
 $diskon = 0.05; //diskon 5%
 $pesan_diskon = 'Diskon 5%';
}
elseif ($total >= 50000 && $total <= 74000)
{
 $diskon = 0.1; //diskon 10%
 $pesan_diskon = 'Diskon 10%';
}
elseif ($total >= 75000)
{
 $diskon = 0.15; //diskon 15%
 $pesan_diskon = 'Diskon 15%';
}
$subtotal = $total - ($total * $diskon);
?>

<html>
```

```

<head>
<title>Kasir Online - MiniMarket Janarko</title>
</head>
<body>
Pembelian Terproses<hr>
<p>Berikut ini faktur pembelian anda</p>

<table border="0">
<tr bgcolor="#cccccc">
<td align="center">Barang</td>
<td align="center">Jumlah Beli</td>
<td align="center">Bayar</td></tr>

<?php
echo '<tr>';
echo '<td>Baterei CBA</td>';
echo '<td align="center">' . $baterei . '</td>';
echo '<td>' . $baterei_ttl . '</td></tr>';
echo '<tr>';
echo '<td>Baygin</td>';
echo '<td align="center">' . $baygin . '</td>';
echo '<td>' . $baygin_ttl . '</td></tr>';
echo '<tr>';
echo '<td>Silver King</td>';
echo '<td align="center">' . $silver . '</td>';
echo '<td>' . $silver_ttl . '</td></tr>';
echo '</table>';
echo "<p>TOTAL: $total
";
echo "$pesan_diskon
";
echo "SUBTOTAL SETELAH DISKON: $subtotal</p>";
?>

</body>
</html>

```

Simpan kembali file tersebut lalu jalankan pada browser. Pada bagian bawah ditambahkan tulisan yang menunjukkan diskon yang diperoleh pembeli.

### PENJELASAN SCRIPT

Karena script ini sudah pernah dibahas sebelumnya maka saya hanya menjelaskan beberapa hal yang baru yaitu mengenai if...elseif.... Pada script diatas terdapat empat kondisi yang mungkin terjadi. User tidak mengisi, tidak ada diskon, diskon 5%, diskon 10% dan diskon 15%.

```
if ($total <= 0)
```

Perintah diatas bereri jika **\$total** kurang dari atau sama dengan 0 berarti user tidak mengisi apa-apa. Kita masukkan kondisi kurang dari nol karena mungkin saja user memasukkan angka negatif. Maka tampilkan pesan lalu keluar script dengan perintah **exit**.

Seperti yang tertera pada masalah yang ingin kita pecahkan. Diskon 0% untuk 25000 kebawah dan seterusnya. Kita menggunakan tanda **>=** (lebih besar dari atau sama dengan) dan **<** (kurang dari) untuk menentukan eksekusi kode.

```
elseif ($total >= 25000 && $total < 75000)
```

Pada kode di atas jika kita terjemahkan berarti *selain itu jika harga total lebih besar dari 25000 dan harga total kurang dari 75000*. Itu berarti 25000 masuk pada diskon 10% sedangkan 75000 tidak termasuk. Tanda **&&** artinya adalah "dan", selain menggunakan tanda **&&** kita juga dapat menuliskan nama operatornya yaitu **"and"**. Pada elseif ketiga kita menggunakan **"and"** hal itu sama saja.

```
$pesan_diskon = 'Diskon 10%' ;
```

Variabel **\$pesan\_diskon** berisi string berupa kode HTML yang bertujuan untuk memberitahu pengunjung berapa diskon yang mereka peroleh. Pesan ini akan kita munculkan pada bagian bawah faktur bersama dengan TOTAL biaya.

## Statemen switch()

Sebenarnya prinsip kerja statemen switch hampir sama dengan if. Tapi switch mengijinkan kita untuk mengambil lebih dari dua nilai. Statemen if hanya mengenal *true* atau *false*. Pada switch kita dapat mengambil kondisi dari berbagai macam nilai dan tipe seperti string, integer maupun double. Untuk menangani kondisi yang kita inginkan digunakan statemen **case**.

### PENERAPAN TEORI SWITCH...

Anda dapat mengganti blok statemen if...elseif... pada proses\_beli.php dengan statemen switch... berikut ini. Hasilnya akan sama persis.

```
switch($total)
{
 case $total > 0 && $total < 25000;
 $diskon = 0; //diskon 0%
 $pesan_diskon = 'Diskon 0%';
 break;
 case $total >= 25000 && $total < 50000;
 $diskon = 0.05; //diskon 5%
 $pesan_diskon = 'Diskon 5%';
 break;
 case $total >= 50000 && $total < 75000;
 $diskon = 0.1; //diskon 10%
 $pesan_diskon = 'Diskon 10%';
 break;
 case $total >= 75000;
 $diskon = 0.15;
 $pesan_diskon = 'Diskon 15%';
 break;
}
```

Karena yang ingin kita handel adalah kondisi dari variabel **\$total** maka kita menuliskan statemennya dengan **switch(\$total)**. Ketika suatu kondisi tertentu ditemukan oleh statemen **case** seperti **case \$total == 0;** maka semua kode yang berada pada case tersebut akan dieksekusi sampai adanya statemen **break**. Statemen **break** menandakan akhir dari suatu kondisi **case**.

## Perulangan (looping)

Salah satu kelebihan komputer adalah melakukan suatu pekerjaan berulang-ulang tanpa rasa bosan sedikitpun. Pada bahasan berikut ini kita akan membahas perulangan yang merupakan salah satu hal yang wajib dikuasai programmer. PHP memiliki empat type perulangan diantaranya **for...**, **foreach...**, **while...**, dan **do...while**. Khusus untuk **foreach** tidak dibahas kali ini melainkan pada materi selanjutnya yang berhubungan dengan array.

Sebelum membahas masing-masing perulangan satu per satu. Kita akan membuat sebuah file HTML terlebih dahulu yang nantinya akan kita selesaikan dari sudut pandang looping. Seperti biasa kita jalankan software favorit kita yaitu PHP DESIGNER 2006 klik **File-New-HTML/XHTML**.

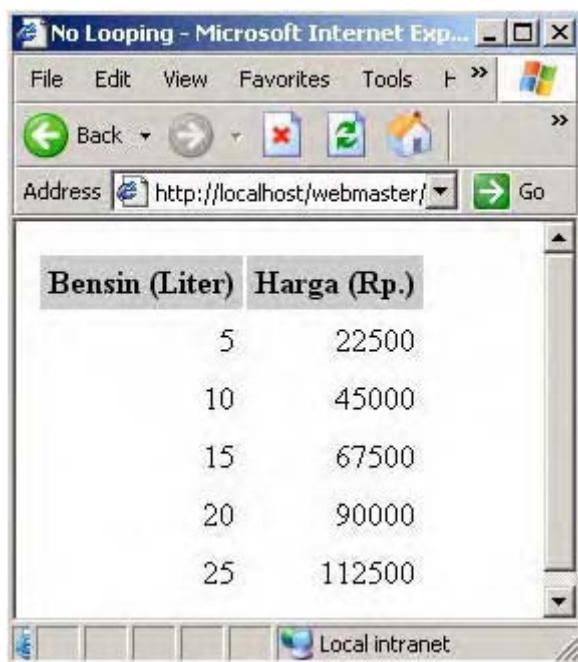
```
<html>
<head>
<title>No Looping</title>
```

```

</head>
<body>
<table border="0" cellpadding="4">
<tr bgcolor="#cccccc">
 <td>Bensin (Liter)</td>
 <td>Harga (Rp.)</td></tr>
<tr>
 <td align="right">5</td>
 <td align="right">22500</td></tr>
<tr>
 <td align="right">10</td>
 <td align="right">45000</td></tr>
<tr>
 <td align="right">15</td>
 <td align="right">67500</td></tr>
<tr>
 <td align="right">20</td>
 <td align="right">90000</td></tr>
<tr>
 <td align="right">25</td>
 <td align="right">112500</td></tr>
</table>
</body>
</html>

```

Simpan dengan nama **bensin.html** pastikan **Save as type**-nya HTML Document sekarang coba jalankan pada browser hasilnya akan seperti gambar 1.11.



Gambar 1.11: Output file bensin.html

Anda sebagai manusia pasti bosan jika harus mengetik dengan cara demikian. Untuk itu telah hadir sebuah terobosan baru --- Lho kok seperti promosi barang?!?. Dengan menggunakan looping kita dapat menghemat waktu dan tenaga kita.

## Perulangan while()

**while** (indonesia: selama) merupakan perulangan paling sederhana dalam PHP. Seperti halnya statemen **if**, **while** bergantung pada suatu kondisi. Perbedaannya adalah jika statemen **if** mengeksekusi blok kode hanya sekali jika kondisi itu true. **while** mengeksekusi blok kode secara berulang-ulang selama kondisi itu true.

Perulangan while dapat anda gunakan ketika anda tidak tahu berapa kali pengulangan itu harus dilakukan untuk membuat kondisi true. Struktur dasar dari perulangan while adalah:

```
while(kondisi) ekspresi;
```

Di bawah ini adalah contoh sederhana penggunaan while.

```
$a = 1;
while ($a <= 5)
{
 echo $a.'
';
 $a++;
}
```

Pada loop while diatas output yang dihasilkan adalah angka 1 sampai dengan 5. Pada awal setiap looping, while akan mengetes kondisi. Jika kondisi false dalam hal ini jika nilai \$a melebihi 5 maka blok kode tidak akan dieksekusi. Sebaliknya jika kondisi masih true dimana nilai \$a masih kurang dari atau sama dengan 5 maka blok kode akan terus dieksekusi.

Arti dari \$a++ adalah pertama nilai dari \$a akan dikembalikan yakni 1, kedua nilai \$a akan ditambah 1. Sekarang nilai \$a menjadi 2, selanjutnya 2 akan ditambah 1 lalu menjadi 3 demikian seterusnya hingga mencapai 5.

### PENERAPAN TEORI WHILE...

Untuk penerapan teori while ini kita akan membuat file yang tampilannya persis seperti file bensin.html tetapi menggunakan looping pada PHP. Jalankan PHP DESIGNER 2006 klik **File-New-PHP** salin kode berikut agar perulangan while nancap di memori otak kita.

```
<html>
<head>
<title>While Loop</title>
</head>
<body>
<table border="0" cellpadding="4">
<tr bgcolor="#cccccc">
 <td>Bensin (Liter)</td>
 <td>Harga (Rp.)</td></tr>

<?php
$liter = 5;
$harga = 4500;
while ($liter <= 25)
{
 echo "<tr>";
 echo "<td align=\"right\">$liter</td>";
 echo '<td align="right">' . $liter * $harga . '</td></tr>';
 $liter += 5;
}
?>

</table>
</body>
</html>
```

Simpan dengan nama **bensin\_while.php**, jika anda menjalankannya pada browser maka hasilnya akan sama persis seperti gambar 1.11.

### PENJELASAN SCRIPT

Pertama kita membuat variabel bernama \$liter yang menentukan nilai awal banyaknya bensin. Selanjutnya kita membuat variabel \$harga yang menampung nilai dari harga bensin

yaitu 4500. Untuk menentukan harga total nantinya kita menggunakan operasi matematik perkalian ( \* ).

```
while ($liter <= 25);
```

Pada statemen while diatas, while mengetes kondisi \$liter apakah kurang dari atau sama dengan 25 jika ya maka true. Berarti blok while akan dieksekusi untuk pertama kali. Karena pada blok while terdapat kode \$liter += 5, yang artinya \$liter = \$liter + 5. Maka pada eksekusi ke dua nilai variabel \$liter tidak lagi 5 melainkan 10. Demikian seterusnya hingga nilai \$liter = 25.

Karena setelah itu kondisi \$liter tidak memenuhi syarat kondisi yang ditentukan while dimana \$liter hanya boleh kurang dari atau sama dengan 25 jadi nilainya false. Maka blok while berhenti dieksekusi. Are You Understand Brother? Good...good....good....

## Perulangan for()

Cara yang kita lakukan pada perulangan while sebelumnya adalah sangat umum dilakukan. Sebelum perulangan kita mengetes kondisi, jika true lakukan eksekusi. Pada akhir dari perulangan kita modifikasi variabel pengetesnya.

Pada for sedikit berbeda, kita menset variabel pengetes lalu kondisi yang diinginkan dan modifikasi variabel pengetes terletak dalam satu baris. Dari sini kita dapat melihat penulisan for akan lebih sederhana dibanding while.

Struktur dasar penulisan for adalah seperti berikut.

```
for (ekspresi1; kondisi; ekspresi2)
 ekspresi3;
```

- **ekspresi1:** dieksekusi satu kali pada saat mulai, disini kita biasanya menentukan nilai awal variabel.
- **kondisi:** akan dites sebelum setiap kali perulangan dilakukan, jika kondisi true maka blok tetap dieksekusi demikian sebaliknya jika false.
- **ekspresi2:** dieksekusi pada pada akhir dari setiap perulangan, disinilah anda merubah nilai variabel pengetes.
- **ekspresi3:** dieksekusi satu kali setiap perulangan, disinilah anda letakkan kode yang ingin anda eksekusi.

## PENERAPAN TEORI FOR...

Saya yakin anda pasti masih bingung, untuk menghapus kebingungan anda obatnya tidak lain adalah satu tablet PHP DESIGNER 2006 dan segelas kode for. Jalankan PHP DESIGNER 2006 klik **File-Open** arahkan ke file **bensin\_while.php** untuk membuka file tersebut.

Ubahlah kode yang terletak pada tag PHP yaitu <?php dan ?> gantilah dengan script berikut.

```
<?php
$harga = 4500;
for ($liter = 5; $liter <= 25; $liter += 5)
{
 echo "<tr>";
 echo "<td align=\"right\">$liter</td>";
 echo '<td align="right">' . $liter * $harga . '</td></tr>';
}
?>
```

**Save As** file ini dengan nama **bensin\_for.php**, agar nanti anda mudah mempelajarinya ulang. Output yang dihasilkan sama persis dengan bensin\_while.php dan bensin.html.

## PENJELASAN SCRIPT

Hampir sama dengan while. Perulangan `for` diatas kita menentukan nilai awal variabel `$liter` yaitu 5. Selanjutnya nilai `$liter` akan dites apakah masih sesuai dengan kondisi dimana `$liter` tidak boleh lebih dari 25.

Untuk itu kondisi dituliskan dengan menggunakan notasi `$liter <= 25`. Pada ekspresi 2 yaitu `$liter += 5` yang artinya setiap kali blok kode selesai dieksekusi maka nilai `$liter` akan selalu ditambah 5. Jadi pada eksekusi ke dua nilai `$liter` adalah 10, pada eksekusi ke tiga nilai `$liter` menjadi 15. Demikian seterusnya hingga nilai `$liter` mencapai 25.

Setelah itu blok kode tidak akan dieksekusi karena kondisi nilai `$liter` menjadi 30. Itu tidak sesuai dengan kondisi yang diinginkan yaitu `$liter <= 25`, sehingga kondisi menjadi false. Maka looping berhenti dilakukan.

## Perulangan `do()...while()`

Perulangan `do...while...` merupakan perulangan yang tergolong berbeda dari `for` dan `while`. Mengapa demikian, ini dikarenakan pengetesan kondisi pada `do...while` dilakukan pada akhir kode. Jadi meskipun kondisi awal nilai variabel false blok kode pada `do...while` setidaknya akan tetap dieksekusi minimal satu kali.

Struktur dasar penulisan looping `do...while` sebagai berikut.

```
do
 ekspresi;
 while (kondisi);
```

## PENERAPAN TEORI DO...WHILE...

Pada PHP DESIGNER 2006 bukalah file **bensin\_for.php** yang pernah anda buat. Gantilah kode yang berada pada tag `<?php` dan `?>` dengan kode berikut ini.

```
<?php
$harga = 4500;
$liter = 5;
do
{
 echo "<tr>";
 echo "<td align=\"right\">$liter</td>";
 echo '<td align="right">' . $liter * $harga . '</td></tr>';
 $liter += 5;
}
while ($liter <= 25);
?>
```

Save As dengan nama **bensin\_do\_while.php**. Karena saya sudah menjelaskan script ini dua kali jadi saya rasa tidak perlu menjelaskannya ulang. Yang perlu anda ingat adalah jika `for` dan `while` akan melakukan pengetesan kondisi pada awal looping, sehingga jika false blok kode tidak akan pernah dieksekusi.

Pada `do...while` meskipun nilai awal false kode minimal akan dieksekusi satu kali. Untuk membuktikannya coba gantilah nilai `$liter` menjadi 30 maka kode akan diekskusi satu kali. Ini tidak berlaku bagi `for` dan `while`.

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

Seperti biasa buatlah folder baru pada direktori

- C:\Apache\htdocs\webmaster\
  - Bab1\ beri nama folder baru tersebut **hari-4**.
- 

## File dan Array

Untuk hari yang ke empat fokus materi yang akan kita pelajari meliputi interaksi PHP dengan file sistem dan array. Pada hari ke empat ini akan cukup sulit jadi mohon kencangkan sabuk pengaman anda. Emangnya mau *take off*.

Pada hari ke empat ini anda diharapkan mengerti tentang:

- Menyimpan File
- Membuka File
- Menulis ke File
- Menutup File
- Membaca dari File
- Menghapus File
- Memahami Array
- Array dengan Index Numerik
- Array dengan Index String

Meskipun dalam konsep diatas kelihatan banyak materi yang harus anda kuasai. Tetapi pada kenyataannya tidak sebanyak itu karena untuk penyimpanan data kita nanti menggunakan **database** bukan file. Namun dalam beberapa situasi anda mungkin memerlukan pengetahuan tentang ini.

### **Menyimpan File**

Sebenarnya ada dua cara pada PHP untuk menyimpan data. Yaitu menggunakan file biasa dan database. Untuk penerapan teori ini nantinya kita akan merubah file minimarket.html agar dapat menyimpan pembelian dari pengunjung.

Penyimpanan data dengan file cukup sederhana tapi mempunyai banyak kekurangan. Terutama jika menyangkut kecepatan akses dan keamanan.

### **Garis Besar Pemrosesan File**

Terdapat tiga langkah untuk menulis ke file, diantaranya:

1. Membuka file. Jika file tidak ada, gunakan exit() untuk keluar dari program
2. Menulis data ke file.
3. Menutup file

Hampir sama dengan menulis, untuk membaca file juga terdapat tiga langkah:

1. Membuka file.
2. Membaca data dari file.
3. Menutup file.

Khusus untuk membaca data dari file, berapa banyak file yang akan kita baca dalam satu waktu. Ini akan kita bahas lebih lanjut nanti.

### **Membuka File**

Untuk membuka file pada PHP kita gunakan fungsi **fopen()**. Pada saat membuka file kita harus menentukan metode apa yang akan kita gunakan pada file ini. Dikenal dengan istilah **File Mode**.

**File Mode** merupakan sinyal yang harus diketahui oleh server agar server dapat menentukan mekanisme kerja yang harus diberikan pada file tersebut. Ada tiga pilihan yang dapat anda gunakan pada saat membuka file.

1. Anda mungkin hanya ingin membaca file saja, menulis file saja, atau keduanya membaca dan menulis file.
2. Pada saat anda ingin menulis file anda mungkin ingin menindih(overwrite) isi dari file atau menambahkannya(append).
3. Jika anda coba menulis ke file pada sistem yang membedakan file binari dan file text biasa, anda harus menentukan ini.

## Menggunakan fopen()

Kita asumsikan kita akan menulis faktur pembelian minimarket online Mr. Web Janarko pada sebuah file. Kita dapat membuka file ini dengan menggunakan perintah.

```
$buka_file = fopen('faktur.txt', 'a'); parameter 1 parameter 2
 ↑ ↓
```

Fungsi **fopen()** minimal memerlukan 2 parameter. Yaitu parameter file yang akan diproses (parameter 1) dan file mode (parameter 2). Setiap parameter dipisahkan oleh tanda " , " koma. Pada parameter anda dapat menggunakan tanda petik satu maupun petik dua.

Jika file anda berada pada subdirektori lagi misalnya "folderX". Maka pada parameter 1 anda harus menuliskan '**folderX/faktur.txt**' Sebaliknya jika file kita berada pada direktori yang diatasnya lagi maka kita gunakan tanda "..." yang artinya pindah satu folder ke atas. Sehingga kita menuliskan parameter 1 menjadi '**../faktur.txt**'

Maksud dari '**a**' pada parameter 2 adalah mode file yang akan kita gunakan adalah **append**. Dimana isi dari file tidak dihapus melainkan ditambah isi baru pada akhir baris. Hal ini perlu dilakukan agar faktur dari pembeli lain tidak ikut terhapus. Daftar lengkap file mode yang tersedia dapat anda lihat pada tabel 1.9.

**Tabel 1.9** File Mode yang tersedia untuk `fopen()`

Mode	Keterangan
r	Read Mode – artinya membuka file untuk dibaca, dimulai dari awal file
r+	Read Mode – artinya membuka file untuk dibaca dan ditulis, dimulai dari awal file
w	Write Mode – artinya membuka file untuk ditulis, dimulai dari awal file. Jika sudah ada tindih file tersebut. Jika file tidak ada, coba dan buat secara otomatis.
w+	Write Mode – artinya membuka file untuk ditulis dan dibaca, dimulai dari awal file. Jika sudah ada tindih file tersebut. Jika file tidak ada buat secara otomatis.
a	Append Mode – artinya membuka file menambahkan(menulis) saja, dimulai dari akhir dari isi file, jika terapat isi. Jika file tidak ada buat secara otomatis.
a+	Append Mode – artinya membuka file untuk menambahkan(menulis) dan dibaca dimulai dari akhir dari isi file, jika terdapat isi. Jika file tidak ada buat secara otomatis.
b	Binary Mode – artinya digunakan untuk menghubungkan satu mode dengan mode lainnya.

Pada contoh kita menggunakan mode '**a**' dimana setiap terjadi transaksi pembelian. Isi dari file faktur tidak dihapus melainkan ditambah pada akhir dari isi. Jika fungsi `fopen()` berhasil membuka file tersebut isi dari file akan disimpan pada variabel `$buka_file`.

## Menulis File

Menulis file pada PHP relatif simpel kita cukup menggunakan fungsi fwrite(). Kita dapat menggunakan fungsi fwrite dengan menuliskan:

```
fwrite($buka_file, $string_untuk_ditulis);
```

Pada pemanggilan fungsi fwrite() diatas parameter pertama yaitu \$buka\_file merupakan file yang dibuka. Parameter kedua merupakan string yang akan ditulis pada file tersebut dapat juga berupa variabel seperti contoh diatas.

Kita tentu harus membuat format file yang ingin kita tulis. Ini nantinya akan mempermudah kita dalam mengedit file tersebut terutama dalam array. Untuk format file yang akan kita tulis ke faktur adalah.

```
$isi_faktur = "$tgl -#- $batereri baterei -#- $baterei_ttl -#- $baygin baygin -#- $baygin_ttl -#- $silver silver king -#- $silver_ttl -#- $subtotal SubTotal\n"
```

Kelihatannya rumit? tidak juga. Jika anda perhatikan setiap elemen diberi tanda -#- ini berguna untuk memisahkan setiap komponen untuk diubah menjadi array. Penjelasan lebih lengkap pada materi selanjutnya.

## Menutup File

Saat anda selesai menggunakan sebuah file anda harus menutupnya. Untuk menutup file anda dapat menggunakan perintah fclose(). Contohnya

```
fclose($buka_file);
```

Fungsi ini akan mengembalikan nilai true jika file berhasil ditutup atau false jika file gagal ditutup.

### PENERAPAN TEORI fopen(), fwrite(), dan fclose()

Untuk penerapan teori ini kita akan merubah file proses\_beli.php dan menambahkan beberapa perintah baru. Untuk itu ikuti langkah-langkah berikut ini.

- Jalankan PHP DESIGNER 2006 klik **File-Open**
- Arahkan ke C:\Apache\htdocs\webmaster\Bab1\hari-4\.
- **Files of type** All files(\*.\*). Pilih file **marketing.html** lalu klik **Open**
- **Save As** file dengan nama yang sama dan letakkan di direktori C:\Apahe2\htdocs\webmaster\Bab1\hari-4\.
- Hal yang sama berlaku untuk file **proses\_beli.php**

Edit file proses\_beli.php, ubahlah kode yang berada pada tag <?php dan ?> dengan menambahkan beberapa baris kode baru seperti berikut.

```
<?php
echo '<tr>';
echo '<td >Baterei CBA</td>';
echo '<td align="center">' . $baterei . '</td>';
echo '<td>' . $baterei_ttl . '</td></tr>';
echo '<tr>';
echo '<td>Baygin</td>';
echo '<td align="center">' . $baygin . '</td>';
echo '<td>' . $baygin_ttl . '</td></tr>';
echo '<tr>';
echo '<td>Silver King</td>';
echo '<td align="center">' . $silver . '</td>';
echo '<td>' . $silver_ttl . '</td></tr>';
echo '</table>';
```

```

echo "<p>TOTAL: $total
";
echo "$pesan_diskon
";
echo "SUBTOTAL SETELAH DISKON: $subtotal</p>";

$tgl = date('H:i, j F');
$isi_faktur = "$tgl -#- $baterei baterei -#- $baterei_ttl -#- $baygin baygin -#-
$baygin_ttl -#- $silver silver king -#- $silver_ttl -#- $subtotal SubTotal\n";

$buka_file = fopen('faktur.txt', 'a');

if (!$buka_file)
{
 echo "<p>Maaf, Pembelian anda tidak dapat diproses. Mohon coba lagi
 nanti</p></body></html>";
 exit;
}

fwrite($buka_file, $isi_faktur);
fclose($buka_file);

echo '<p>Faktur telah tersimpan</p>';
?>

```

Setelah selesai simpan kembali file proses\_beli.php. Sekarang coba jalankan file <http://localhost/webmaster/Bab1/hari-4/marketing.html>.

Coba anda isikan sembarang angka dan klik tombol beli, pada akhir dari baris akan muncul tulisan faktur tersimpan berarti PHP berhasil menulis file *faktur.txt*. Sekarang pada direktori hari-4 terdapat file baru yaitu faktur.txt coba buka file tersebut dengan notepad atau langsung lewat IE.

## PENJELASAN SCRIPT

Pada penambahan kode baru diatas terdapat variabel `$tgl = date(H:i, j F)`. `date()` merupakan fungsi built-in PHP untuk menampilkan format tanggal dan waktu. Kita akan membahasnya pada materi khusus yang membahas **Tanggal dan Waktu**.

Pada variabel `$isi_faktur` kita memisahkan setiap elemen dengan tanda `-#-` (kita akan mengetahuinya nanti pada pembahasan array). Yang perlu anda ketahui adalah tanda `"\n"`. Tanda ini memberitahukan PHP agar berpindah ke baris baru (new line).

Pada contoh setiap transaksi dipisahkan oleh baris baru, kita memilih ini karena dianggap paling mudah. Jadi setiap file proses\_beli.php menulis data ke file faktur.txt ia akan menulis pada baris baru bukan dibelakang huruf paling akhir.

```
$buka_file = fopen('faktur.txt', 'a');
```

arti dari fungsi `fopen` pada kode diatas adalah pertama PHP akan membuka file bernama ***faktur.txt*** jika file itu tidak ada maka buat (ingat file mode '`a`'). File ini diletakkan satu folder bersama file pemroses karena tidak ada tanda khusus berupa masuk ke subdirektori lain atau pindah ke parent direktori `(..)`.

```

if (!$buka_file)
{
 echo "<p>Maaf, Pembelian anda tidak dapat diproses. Mohon coba lagi
 nanti</p></body></html>";
 exit;
}

```

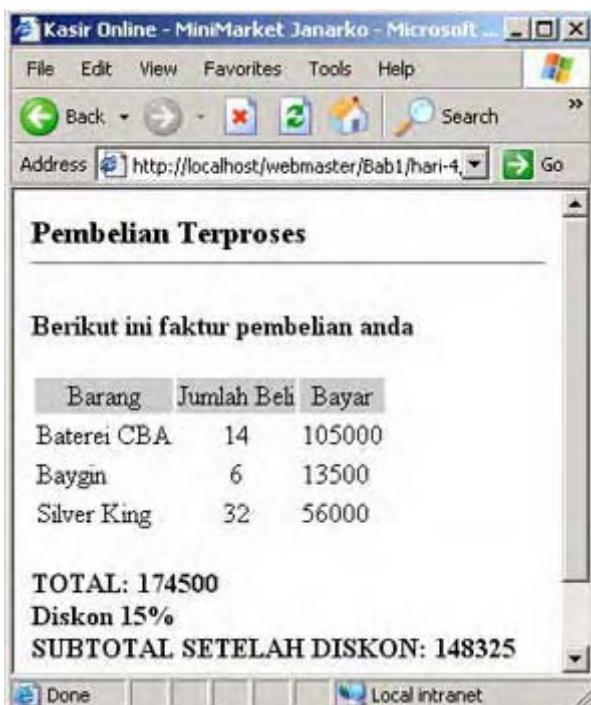
Pada statemen if diatas terdapat tanda `"!"` didepan variabel `$buka_file`. Tanda ini dimaksudkan jika ada kesalahan pada proses `fopen()` sebelumnya. Maka munculkan pesan dan

akhiri jalannya script dengan fungsi `exit();`. Karena script akan berhenti dieksekusi maka kita tutup dengan tag `</body></html>` pada akhir echo.

```
fwrite($buka_file, $isi_faktur);
fclose($buka_file);

echo '<p>Faktur telah tersimpan</p>';
```

Karena pada fungsi `fopen()` sebelumnya file mode yang kita gunakan 'a' maka saat menulis file `fwrite()` akan menambahkan data pada setiap akhir data pada file. `$isi_faktur` merupakan string yang akan kita tulis ke file faktur.txt. Setelah file selesai ditulis maka sebagai pengguna yang baik kita harus menutupnya dengan `fclose($buka_file)` bukan begitu?. Agar user mengetahui sebaiknya kita memunculkan pesan yang menunjukkan bahwa proses penyimpanan faktur telah berhasil.



Gambar 1.12: output proses\_beli.php setelah modifikasi

## Membaca File

Sebenarnya PHP menyediakan banyak fungsi untuk membaca file. Namun yang paling mudah digunakan adalah fungsi `readfile()`. Pada fungsi ini PHP akan membaca seluruh isi file sekaligus. `readfile()` akan mencetak tampilan standar ke browser dan secara otomatis menutup file yang telah dibuka. Struktur dasar penulisan fungsi `readfile()` adalah

```
readfile(string nama_file, [gunakan_included_path]);
```

parameter pertama merupakan string nama file yang akan anda buka. Parameter ke dua bersifat opsional artinya boleh digunakan atau tidak. Jika digunakan PHP akan mencari file pada direktori `included_path` yang ada pada konfigurasi PHP.

### PENERAPAN TEORI `readfile()`

Jalankan PHP DESIGNER 2006 buatlah file PHP baru dengan mengklik **File-New-PHP**. Ketik script dibawah ini.

```
<html>
<head>
<title>Minimarket Online - Lihat Faktur</title>
```

```

</head>
<body>
<h2>Daftar Faktur Pembelian Konsumen</h2>

<?php
 readfile('faktur.txt');
?>

</body>
</html>

```

Simpan script diatas dengan nama ***lihat\_faktur.php***. Saya kira tidak perlu dijelaskan masa cuma tiga baris kode saja dijelaskan?!?. Jalankan pada browser anda output tampilannya akan seperti gambar 1.13.



Gambar 1.13: Daftar transaksi konsumen di minimarket Mr. Web Janarko

## Menghapus File

Jika anda ingin menghapus file anda dapat menggunakan fungsi `unlink()`. Mungkin anda mudah lupa akan fungsi ini karena biasanya untuk menghapus nama fungsinya biasanya ada embel-embel kata "delete". Contoh penggunaan dari fungsi `unlink` adalah

```
unlink('faktur.txt');
```

Jika berhasil akan mengembalikan nilai true, sebaliknya jika false. Kegagalan `unlink` biasanya disebabkan file tidak ada atau permission(hak) atas file dibatasi.

## Array

Dari hari pertama sering kita jumpai kata array. Sebenarnya **apa itu array?**. **Array** adalah sebuah variabel yang menyimpan sekumpulan atau beberapa nilai. Ini berbeda dengan variabel yang hanya menyimpan sebuah nilai. Array juga memiliki banyak elemen. Setiap elemen pada array dapat menyimpan sebuah nilai, seperti text atau angka. Array dapat juga berisi array lainnya, array tipe ini disebut **array multidimensi**. Array merupakan bagian penting dalam sebuah pemrograman.

PHP mendukung penggunaan dua tipe index array yaitu tipe numerik dan tipe string. Maksudnya bukan isi dari array tersebut melainkan bagaimana memanggil array tersebut untuk mendapatkan nilai array tersebut. Waduh...Bingung?!?, sabar sebentar "**apotik contoh**" belum buka nanti kalau buka saya belikan obatnya, OK.

Baterei CBA	Baygin	Silver King
-------------	--------	-------------

Gambar 1.14: Produk Minimarket Mr. Web Janarko dapat diubah menjadi array

Untuk gampangnya kita mengambil contoh produk dari minimarket dan akan kita jadikan array. Pada gambar 1.4 produk terdiri dari tiga barang yaitu baterei CBA, Baygin dan Silver King. Setiap nilai yang ada didalam array disebut elemen array. Setiap elemen mempunyai tanda pengenal berupa **index** yang digunakan untuk mengakses nilai dari elemen itu sendiri.

## Array dengan Index Numerik

Array dengan tipe ini setiap elemen pada array diberi index mulai dari 0,1,2,..., dan seterusnya. "Apotik Contoh" telah buka rupanya untuk itu minumlah obat ini.

```
$produk = array('Baterei CBA', 'Baygin', 'Silver King');
```

Pada kode diatas kita membuat array dengan nama **\$produk** yang berisi tiga nilai yaitu Baterei CBA, Baygin, dan Silver King. **array()** merupakan perintah yang digunakan pada PHP untuk membuat array. Selanjutnnya kita akan bahas bagaimana mengakses array.

## Mengakses Isi Array

Untuk mengakses isi dari variabel kita menggunakan nama variabel tersebut. Jika variabel itu beripe array maka kita gunakan index array tersebut. Jika kita gambarkan pada tabel maka array dari **\$produk** adalah

**Tabel 1.10** Variabel array **\$produk** dan nilainya

Variabel	Nilai
<b>\$produk[0]</b>	'Baterei CBA'
<b>\$produk[1]</b>	'Baygin'
<b>\$produk[2]</b>	'Silver King'

Ingin array dengan index numerik selalu diawali dengan index **0**. Ciri-ciri array lainnya, index array selalu diletakkan dalam tanda [ ] ( kurung siku ). Jadi jika kita ingin menampilkan nilai array pada browser maka kita gunakan perintah.

```
echo "$produk[0] $produk[1] $produk[2]";
```

Tampilan yang dihasilkan dari perintah diatas adalah

Baterei CBA Baygin Silver King → Output 1

Selain dengan menggunakan perintah **array(....)** anda dapat langsung membuat array **\$produk** dengan mengetikkan variabel array dan indexnya secara langsung.

<pre>\$produk[0] = 'Baterei CBA'; \$produk[1] = 'Baygin'; \$produk[2] = 'Silver King';</pre>	Atau <pre>\$produk[ ] = 'Baterei CBA'; \$produk[ ] = 'Baygin'; \$produk[ ] = 'Silver King';</pre>
----------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------

Dengan menggunakan perintah diatas anda akan memproleh hasil yang sama dengan perintah **\$produk = array('Baterei CBA', 'Baygin', 'Silver King')** seperti pada output 1.

## Looping dalam array dengan index numerik

Untuk mengakses nilai dari suatu array kita juga dapat menggunakan looping. Looping sangat berguna jika kita mengakses array dalam jumlah yang sangat besar. Dan sangat melelahkan jika kita mengaksesnya secara manual.

## Looping array dengan for( )

Pada hari ke-2 kita sudah pernah membahasnya. Sekarang kita akan menggunakan for untuk mengakses nilai array. Looping for ini memang cocok digunakan untuk tipe array dengan index numerik. Untuk mendapatkan tampilan yang sama dengan **output 1**. Kita dapat menuliskan for dengan cara

```
for ($i=0; $i<3; $i++)
{
 echo "$produk[$i] ";
}
```

Untuk variabel pengetes \$i kita set nilainya 0 karena array \$produk dimulai dari 0. Setelah itu kondisi yang kita tentukan adalah \$i<3 artinya yang masuk dalam kondisi adalah 0,1,2. Pada \$i++ setiap kali blok kode selesai dieksekusi nilai \$i akan selalu ditambah 1. Blok kode akan berhenti dieksekusi jika nilai \$i lebih dari 2 karena tidak sesuai dengan kondisi yang ditentukan dimana \$i kurang dari 3.

## Looping array dengan while( )

Untuk while juga sama hanya saja penempatan variabel pengetes dan kondisi berbeda letaknya. Kode while dibawah ini juga akan menghasilkan tampilan seperti output 1.

```
$i=0;
while ($i<3)
{
 echo "$produk[$i] ";
 $i++;
}
```

## Looping array dengan foreach( )

Ini adalah satu satunya looping yang belum kita bahas. Mengapa? Karena foreach() merupakan looping yang didesain khusus untuk array. Nah saat inilah saat yang paling tepat untuk mengenal si foreach tersebut lebih jauh. Kali aja orang tuanya setuju, mau ngelamar atau belajar nih...?!?!

Struktur dasar dari perulangan foreach adalah

```
foreach(ekspresi1 as ekspresi2)
 ekspresi3;
```

Dimana:

- ekspresi1, merupakan variabel array yang akan dimanipulasi
- ekspresi2, merupakan variabel baru tempat menyimpan setiap nilai elemen array
- ekspresi3, merupakan tempat anda meletakkan kode yang ingin dieksekusi

untuk mendapatkan tampilan seperti output 1, penulisan foreach untuk variabel \$produk dapat kita ketikkan seperti berikut

```
foreach($produk as $barang);
{
 echo "$barang ";
}
```

Jadi jika kita terjemahkan kedalam bahasa indonesia yang baik dan benar akan berbunyi. *Untuk setiap elemen array pada produk jadikan sebagai variabel barang kemudian tampilkan barang hingga elemen yang terakhir.*

Pada looping ini setiap elemen dari variabel array \$produk akan diubah menjadi variabel \$barang. Isi dari variabel barang tidak lain adalah nilai dari elemen array \$produk. Karena merupakan looping jadi \$variabel akan diulang sampai array terakhir dalam hal ini tiga kali.

## PENERAPAN TEORI ARRAY INDEX NUMERIK

Agar anda lebih memahami penggunaan array dengan tipe index numerik ini jalankan PHP DESIGNER 2006 klik menu **File-New-PHP** untuk membuat file PHP baru. Ketik kode berikut

```
<html>
<head><title>Array index numerik</title></head>
<body>
<h2>Array dengan Index Numerik</h2>

<?php
//Membuat variabel $produk bertipe array dengan tiga nilai

//mencetak array ke browser
$produk = array('Baterei CBA', 'Baygin', 'Silver King');
echo "$produk[0] $produk[1] $produk[2]";

//mencetak array menggunakan looping while...
echo "<hr><p>Looping while</p>";
$i = 0;
while($i<3)
{
 echo "$produk[$i] ";
 $i++;
}

//mencetak array menggunakan looping for...
echo '<hr><p>Looping for</p>';
for ($i=0; $i<3; $i++)
{
 echo "$produk[$i] ";
}

//mencetak array menggunakan looping foreach...
echo '<hr><p>Looping foreach</p>';
foreach($produk as $barang)
{
 echo "$barang ";
}
?>

</body></html>
```

Simpan dengan nama **array\_index\_numerik.php**. Lalu jalankan pada browser semua perintah looping akan mengarah pada hasil yang sama karena memang saya desain demikian agar mudah dimengerti.

## Array dengan Index String

Selain menggunakan angka atau numerik pengaksesan array pada PHP dapat juga menggunakan string (huruf). Jika pada tipe numerik penomoran index array dimulai dari 0, 1 dan seterusnya. Pada array dengan tipe index string kita dapat memberi nama index array sesuai dengan keinginan kita.

```
$produk = array('btr'=>'Baterei CBA', 'byg'=>'Baygin', 'slv'=>'Silver King');
```

Sedikit berbeda dengan cara pembuatan array dengan index numerik. Pada array index numerik kita hanya perlu mendefinisikan isi atau nilai dari array itu saja. Sedangkan pada index

string kita harus menentukan nama index dan nilai index array tersebut. Pada contoh diatas 'btr', 'byg', dan 'slv' merupakan nama index dari array \$produk. 'Baterei CBA', 'Baygin', dan 'Silver King' merupakan nilai dari array \$produk. Tanda "`=>`" digunakan untuk menghubungkan antara nama index dan isinya.

Selain menggunakan cara diatas kita juga dapat membuat array yang sama dengan cara:

```
$produk = array('btr'=>'Baterei CBA');
$produk = array('byg'=>'Baygin');
$produk = array('slv'=>'Silver King');
```

atau yang lebih mudah,

```
$produk['btr'] = 'Baterei CBA';
$produk['byg'] = 'Baygin';
$produk['slv'] = 'Silver King';
```

Untuk mengakses nilai dari array produk kita dapat menuliskan nama array dan nama index array tersebut. Contohnya

```
echo $produk['btr'] $produk['byg'] $produk['slv'];
```

output yang ditampilkan pada browser akan sama persis dengan output 1 pada bahasan sebelumnya.

## Looping dalam Array dengan Index String

Karena index dari array tipe ini bukan merupakan angka maka kita tidak dapat menggunakan looping `for` pada array jenis ini.

### Looping array dengan `foreach()`

`foreach` memiliki struktur yang cukup berbeda jika kita gunakan dalam tipe array ini. Anda juga dapat menggunakan cara penulisan seperti pada tipe numerik sebelumnya dengan hanya menampilkan isi array-nya saja seperti

```
foreach($produk as $barang)
{
 echo $barang.' ';
}
```

Bandingkan dengan perintah perintah berikut ini yang menampilkan isi array berserta nama indexnya sekaligus.

```
foreach($produk as $index_brg => $barang)
{
 echo $index_brg.' >>> '.$barang.'
'; //output 2
}
```

Output yang ditampilkan pada browser akan seperti

btr >>> Baterei CBA	—————>	Output 2
byg >>> Baygin		
slv >>> Silver King		

Pada statemen `foreach` yang ke dua kita menemukan notasi `$index_brg => $barang`. Karena setelah variabel `$index_brg` kita beri tanda "`=>`" maka variabel `$index_brg` akan dianggap sebagai index dari array. Sedangkan `$barang` yang berada disebelah kanan tanda "`=>`" akan dianggap sebagai isi dari array tersebut.

## Looping Array dengan `while()`

Karena index array ini bertipe string kita tidak dapat menggunakan looping while seperti pada array numerik. Untuk itu kita akan menggabungkan looping while dengan fungsi lain seperti `each()` dan `list()` dalam mengidentifikasi index elemen array.

### `each()`

Untuk lebih memahami penggunaan `each()` pada looping while lihatlah contoh berikut:

```
reset($produk);
while($barang = each($produk))
{
 echo $barang['key'].' >>> '.$barang['value'];
}
```

Output tampilan yang dihasilkan kode diatas akan dengan output 2. Looping while diatas kita gabungkan dengan fungsi `each()`. `each()` pada looping while, akan mengembalikan elemen didalam array, merubah dan berhenti saat elemen array terakhir dicapai.

Pada kode diatas perintah `each()` akan mengembalikan empat index array. `key` dan `0` akan merujuk pada nama index array yang aktif. sedangkan `value` dan `1` akan merujuk pada isi dari array yang aktif. Jadi arti `key` dan `value` pada kode diatas bukan nama index yang kita buat melainkan semacam suatu keyword yang digunakan untuk mengakses array.

Sebagai alternatif anda dapat mengganti blok kode pada while diatas dengan kode berikut ini

```
echo $barang[0].' >>> '.$barang[1];
```

`reset()` digunakan untuk mengembalikan array yang aktif kembali ke posisi awal. Tanpa menuliskan `reset()` hasil yang kita inginkan tidak keluar karena array yang aktif adalah index array terakhir.

### `list()`

Ada cara lain yang dapat digunakan dalam mengakses array untuk menghasilkan tampilan yang sama. `list()` digunakan untuk memecah array menjadi index dan nilai dan memasukkan nilainya pada variabel yang telah ditentukan.

```
list($index_brg, $barang) = each($produk);
```

pada kode diatas kita menggunakan `list()` untuk memecah index array ke variabel `$index_brg` dan isi dari array kita simpan pada variabel `$barang`. Untuk menghasilkan output yang sama dengan output 2 kita dapat menggunakan list sebagai berikut

```
reset($produk)
while(list($index_brg, $barang) = each($produk))
{
 echo $index_brg.' >>> '.$barang.'
';
}
```

Karena pada kasus diatas kita letakkan `list()` pada looping maka index dan isi akan dicetak dari awal array sampai array yang terakhir.

**key** dan **value** merupakan keyword khusus yang digunakan untuk mengakses array. **key** untuk mengakses index array sedangkan **value** untuk mengakses isi array.

## PENERAPAN TEORI ARRAY INDEX STRING

Jalankan PHP DESIGNER 2006 lalu buatlah file baru. Agar tangan anda tidak kaku ketiklah script berikut ini

```
<html>
<head><title>Array index String</title></head>
<body>
<h2>Array dengan Index String</h2>

<?php

$produk = array('btr'=>'Baterei CBA', 'byg'=>'Baygin', 'slv'=>'Silver King');

// mencetak nilai array ke browser
echo $produk['btr'].' '.$produk['byg'].' '.$produk['slv'];

// mencetak isi array menggunakan looping foreach...
echo "
<p>Looping foreach tanpa nama index</p>";

foreach($produk as $barang)
{
 echo $barang.' ';
}

// mencetak nama index dan isi dari array menggunakan looping foreach
echo '
<p>Looping foreach dengan nama index</p>';

foreach($produk as $index_brg => $barang)
{
 echo $index_brg.' >>> '.$barang.'
';
}

// mencetak nama index dan isi dari array menggunakan while dan each()
echo "
<p>Looping while dengan each()</p>";

reset($produk);
while ($barang = each($produk))
{
 echo $barang['key'].' >>> '.$barang['value'].'
';
}

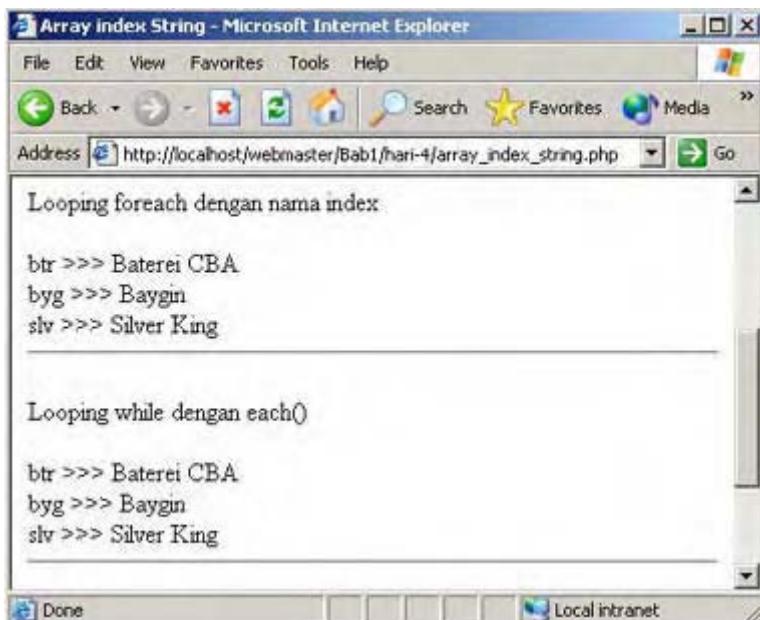
// mencetak nama index dan isi dari array menggunakan while, list dan each()
echo "
<p>Looping while dengan list() dan each()</p>";

reset($produk);

while (list($index_brg, $barang) = each($produk))
{
 echo $index_brg.' >>> '.$barang.'
';
}
?>

</body></html>
```

simpan dengan nama **array\_index\_string.php** lalu jalankan pada browser. Output tampilan script diatas dapat anda lihat pada gambar 1.15.



Gambar 1.15: output file array\_index\_string.php

## Array Multidimensi

Apakah array jenis ini berasal dari dunia lain? Wah anda kebanyakan nonton film fiksi. Array multidimensi dapat kita gambarkan sebagai sebuah matriks yang memiliki baris dan kolom. Oleh karena itu kita dapat membuat array 2-dimensi dan 3-dimensi.

### Array dua-dimensi

Jika kita ingin menyimpan produk minimarket Mr. Web Janarko lebih dari satu bagian kita dapat menggunakan array dua-dimensi. Gambar 1.16 menunjukkan penggunaan array dua-dimensi pada produk minimarket. Dimana setiap baris menunjukkan produk dan setiap kolom menunjukkan attribut produk seperti kode, nama, dan harga.

	Kode	Nama Barang	Harga
p	BTR	Baterei CBA	7500
r	BYG	Baygin	2250
o	SLV	Silver King	1750
d			
u			
k			

Attribut produk →

Gambar 1.16: Kita dapat membuat produk minimarket menjadi array dua-dimensi.

Dengan menggunakan PHP kita dapat menulis array dari gambar 1.16 seperti berikut:

```
$produk = array(array('BTR', 'Baterei CBA', 7500),
 array('BYG', 'Baygin', 2250),
 array('SLV', 'Silver King', 1750));
```

Dengan menggunakan gaya penulisan seperti diatas kita akan lebih mudah mengidentifikasi setiap elemen baris dan kolom dari array tersebut. Untuk mengakses array satu-dimensi seperti yang pernah kita buat, kita perlu menuliskan nama array dan index dari array itu. Pada dua-dimensi juga tidak berbeda jauh hanya saja setiap elemen memerlukan dua index untuk **baris dan kolom**. Baris paling atas indexnya **0** dan kolom paling kiri nilainya **0**.

Untuk menampilkan array \$produk dua-dimensi diatas, kita dapat menuliskannya seperti ini

```
echo '#'.$produk[0][0].'#'.$produk[0][1].'#'.$produk[0][2].'#
';
echo '#'.$produk[1][0].'#'.$produk[1][1].'#'.$produk[1][2].'#
';
echo '#'.$produk[2][0].'#'.$produk[2][1].'#'.$produk[2][2].'#
';
```

output yang ditampilkan pada browser adalah:

```
#BTR#Baterei CBA#7500#
#BYG#Baygin#2250#
#SLV#Silver King#1750#
```

—————→ Output 3

kita dapat menyingkat penulisan kode diatas dengan menggunakan perintah looping. Lihatlah contoh berikut ini

```
for($baris=0; $baris<3; $baris++)
{
 for($kolom=0; $kolom<3; $kolom++)
 {
 echo '#'.$produk[$baris][$kolom];
 }
 echo '#
';
}
```

Tujuan looping for \$baris adalah untuk membuat nilai index baris dan berpindah ke baris baru. Untuk looping for \$kolom kita ingin mendapatkan nilai kolom dan mencetak index dan isi array. Looping for untuk kolom ini akan mencetak satu baris dahulu yang indexnya 0 dan kolomnya 0,1,2 baru akan melanjutkan ke baris selanjutnya. Dan akan diulang terus hingga nilai \$baris mencapai 2.

Dibanding dengan menggunakan baris dan kolom yang susah diingat. Anda mungkin lebih suka dengan menggunakan index tipe string sebagai pengingat letak baris dan kolom. Dengan menggunakan array dengan tipe index string kita tidak perlu mengingat nilai yang diletakkan di [x][y].

```
$produk_2 = array(array('Kode' => 'BTR',
 'Nama Barang' => 'Baterei CBA',
 'Harga' => 7500),
 array('Kode' => 'BYG',
 'Nama Barang' => 'Baygin',
 'Harga' => 2250),
 array('Kode' => 'SLV',
 'Nama Barang' => 'Silver King',
 'Harga' => 1750)
);
```

untuk menghasilkan tampilan seperti output tiga kita dapat mengetik secara manual seperti cara sebelumnya hanya saja index yang ke dua kita ganti dengan nama kolom misalnya untuk menampilkan **BTR** kita menuliskan \$produk[0]['Kode'].

Dengan menggunakan looping for kode yang kita tuliskan akan lebih singkat dan enak dipandang, emang cewek!.

```
for ($baris=0; $baris<3; $baris++)
{
echo '#'.$produk_2[$baris]['Kode'].">#'.$produk_2[$baris]['Nama Barang']. '#'
 .$produk_2[$baris]['Harga']. '#
';
}
```

Hasil yang ditampilkan dari looping for diatas sama dengan output 3. Menggunakan looping for kita bisa mendapatkan index bertipe numerik dari \$produk. Setiap baris dalam \$produk memiliki index bertipe string. Dengan menggunakan each() dan list() kita bisa mendapatkan index string tersebut.

```

reset($produk_2);
for($baris=0; $baris<3; $baris++)
{
 while(list($index_brg, $barang) = each($produk_2[$baris]))
 {
 echo '#'.$barang;
 }
 echo '#
';
}

```

Sekedar mengingatkan `reset()` digunakan untuk mengembalikan posisi array kembali ke index yang awal akibat adanya perintah `each()`. `list()` pada kode diatas akan mengembalikan nilai variabel `$index_brg` sebagai index array numerik. Sedangkan `$barang` akan menyimpan isi dari variabel array. Output kode diatas sama dengan output 3.

Alternatif lain jika anda ingin menampilkan index array-nya, anda bisa merubah perintah echo diatas dengan:

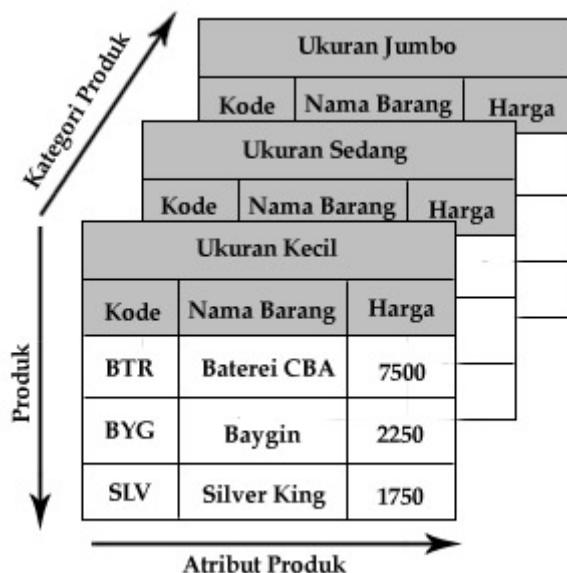
```
echo ' # '.$index_brg.' -> '.$barang.' ';
```

Output pada browser dari kode diatas adalah sebagai berikut.

```
Kode -> BTR # Nama Barang -> Baterei CBA # Harga -> 7500
Kode -> BYG # Nama Barang -> Baygin # Harga -> 2250
Kode -> SLV # Nama Barang -> Silver King # Harga -> 1750
```

## Array tiga-dimensi

Sama halnya dengan array dua-dimensi array tiga dimensi juga memiliki baris dan kolom plus lapisan. Untuk memahai struktur array tiga dimensi lihatlah gambar 1.17 dibawah ini



Gambar 1.17: contoh array 3-dimensi mengizinkan kita untuk menambahkan kategori produk

Dari gambar diatas kita dapat membuat array tiga dimensi dengan menuliskan kode seperti dibawah ini

```
$produk = array(array(array('BTR_KECIL', 'Baterei CBA', 7500),
 array('BYG_KECIL', 'Baygin', 2250),
 array('SLV_KECIL', 'Silver King', 1750)
),
```

```

 array(array('BTR_SEDANG', 'Baterei CBA', 15000),
 array('BYG_SEDANG', 'Baygin', 4500),
 array('SLV_SEDANG', 'Silver King', 3500)
),
 array(array('BTR_JUMBO', 'Baterei CBA', 22500),
 array('BYG_JUMBO', 'Baygin', 6750),
 array('SLV_JUMBO', 'Silver King', 5250)
)
);
 }

```

Karena array yang kita buat diatas hanya memiliki index numerik maka kita dapat menggunakan looping for untuk mengakses nilai dari array.

```

for ($kategori=0; $kategori<3; $kategori++)
{
 echo "<< Kategori $kategori. >>
";
 for ($baris=0; $baris<3; $baris++)
 {
 for ($kolom=0; $kolom<3; $kolom++)
 {
 echo ' # '.$produk[$kategori][$baris][$kolom];
 }
 echo " #
";
 }
 echo '
';
}

```

Output dari kode diatas adalah seperti berikut

```

<< Kategori 0. >>
BTR_KECIL # Baterei CBA # 7500
BYG_KECIL # Baygin # 2250
SLV_KECIL # Silver King # 1750

<< Kategori 1. >>
BTR_SEDANG # Baterei CBA # 15000
BYG_SEDANG # Baygin # 4500
SLV_SEDANG # Silver King # 3500

<< Kategori 2. >>
BTR_JUMBO # Baterei CBA # 22500
BYG_JUMBO # Baygin # 6750
SLV_JUMBO # Silver King # 5250

```

Dari pembuatan array tiga-dimensi diatas dapat anda lihat kalau sebenarnya pembuatan array tidak hanya terbatas pada tiga-dimensi saja. Anda bisa membuat array empat-dimensi, lima-dimensi dan seterusnya. Namun akan lebih mudah memecahkan permasalahan dengan menggunakan array dua atau tiga-dimensi.

## Fungsi-fungsi pada Array

Suatu waktu anda mungkin ingin merubah variabel biasa menjadi variabel array atau sebaliknya. Untuk itu kita dapat menggunakan beberapa fungsi built-in yang disediakan PHP dalam mengatasi permasalahan tersebut.

### **count() dan sizeof()**

Fungsi count() dan sizeof() keduanya sama-sama digunakan untuk menghitung jumlah elemen dalam suatu array. Lihat contoh berikut

---

```
$produk = array('Baterei CBA', 'Baygin', 'Silver King');
echo count($produk);
```

Hasil dari perintah `echo count($produk)` adalah 3. hal yang sama berlaku untuk fungsi `sizeof()`.

### **explode()**

Fungsi `explode()` memecah string menjadi array dengan ketentuan tanda pemisah yang telah ditentukan. Struktur dasar penulisan `explode()` adalah

```
explode(string pemisah, string yang akan dipisah, [limit])
```

Parameter ketiga bersifat opsional, boleh tidak ditulis.

```
$produk = 'Baterei CBA-#-Baygin-#-Silver King';
$pecahkan = explode('-#-', $produk);
echo $pecahkan[0]. '
'. $pecahkan[1]. '
'. $pecahkan[2];
```

dapat anda lihat tanda pemisah yang kita tentukan adalah "-#-". Tanda pemisah tidak harus simbol, **spasi** pun dapat menjadi tanda pemisah. Dengan `explode` isi dari variabel itu dipecah menjadi array. Jadi variabel `$pecahkan` pada kode diatas otomatis menjadi variabel array. Output pada browser dari kode diatas adalah

```
Baterei CBA
Baygin
Silver King
```

### **implode()**

Kalau `explode()` memecah string menjadi array. `implode()` lebih mulia karena dapat menyatukan bangsa dan negara yang tercerai berai. Saya ini ngomong apa ya?. `implode()` akan menyatukan array ke dalam variabel biasa dengan memberikan tanda pemisah yang kita inginkan. Struktur dasar penulisan `implode()` adalah

```
implode(string penghubung, array yang ingin digabung)
```

Misalnya kita mempunyai variabel array `$produk` yang isinya Baterei CBA, Baygin dan Silver King. Untuk menyatukan elemen array tersebut kita gunakan:

```
$produk = array('Baterei CBA', 'Baygin', 'Silver King');
$satukan = implode('-|-', $produk);
echo $satukan;
```

Pada kode diatas `implode` akan menggabungkan elemen pada variabel `$produk` dan menyatukannya ke dalam variabel `$satukan` dengan memberikan tanda pemisah berupa "-|>". Tampilan yang akan keluar pada browser jika dijalankan adalah

```
Baterei CBA-|Baygin-|Silver King
```

### **end(), current(), next() dan prev()**

Keempat fungsi diatas adalah untuk memudahkan kita dalam mengakses elemen array sesuai dengan posisi yang kita inginkan. Fungsi `end()` akan menuju ke posisi elemen array yang terakhir.

`current()` akan menunjuk posisi array yang aktif saat ini. `next()` akan menuju posisi array berikutnya dari posisi array yang aktif. Sedangkan `prev()` akan menuju posisi array sebelumnya dari posisi array yang aktif.

Untuk `next()` dan `prev()` biasanya diletakkan didalam looping. Untuk lebih memperjelas lihat contoh berikut.

```
$produk = array('Baterei CBA', 'Baygin', 'Silver King');
$barang = end($produk);
while ($barang)
{
 echo $barang.' ';
 $barang = prev($produk);
}
```

Output dari kode diatas adalah

Silver King Baygin Baterei CBA

Anda lihat bahwa susunannya terbalik bukan?. Sebelum looping while kita tentukan variabel `$barang` yang isinya adalah `end($produk)`. Ini berarti elemen terakhir dari array `$produk` yaitu Silver King. Selanjutnya dalam blok while kita menuliskan `$barang = prev($produk)` artinya setiap eksekusi elemen array mundur satu elemen hingga menemui elemen terakhir.

## Membaca Array dari File

Sebelumnya kita telah menyimpan pembelian para pengunjung Minimarket Mr. Web Janarko pada file faktur.txt. format yang kita gunakan untuk menyimpan setiap pembelian konsumen adalah sebagai berikut

```
20:38, 16 November -#- 5 baterei -#- 37500 -#- 8 baygin -#- 18000 -#- 10 silver
king -#- 17500 -#- 67500 SubTotal
```

Untuk menjadikan setiap baris pada file menjadi elemen array. kita dapat menggunakan fungsi `file()`. Fungsi ini memiliki satu parameter yaitu nama file yang akan dibuka. Misalnya ada sebuah file bernama `fileku.txt`. Isinya adalah

```
Aku baris 1
Aku baris 2
```

Untuk mengubahnya menjadi array kita gunakan fungsi `file('fileku.txt')`. Dengan asumsi letak `fileku.txt` di direktori yang sama dengan script PHP.

```
$buka_file = file('fileku.txt');
echo $buka_file[0]; //output Aku Baris 1
echo $buka_file[1]; //output Aku Baris 2
```

## PENERAPAN TEORI MEMBACA ARRAY DARI FILE

Kali ini kita akan mempraktekkan teori yang kita pada bahasan-bahasan tentang array sebelumnya. Untuk itu jalankan PHP DESIGNER 2006 lalu ketik kode dibawah ini.

```
<html>
<head><title>Catatan Faktur Pembelian</title></head>
<body>
<h2>Daftar Faktur Pembelian Pengunjung</h2>

<?php
// buka file faktur.txt dengan perintah file() bukan readfile()
// file() akan memecah setiap baris atau pembelian menjadi array
$faktur = file('faktur.txt');

// hitung jumlah array atau pembelian
$jml_pembelian = count($faktur);
```

```

// jika $jml_pembelian == 0 maka tidak ada pembelian
if ($jml_pembelian == 0)
{
 echo "<p>Waduh, Kok sepi tidak ada yang beli</p>";
}

//buat tabel untuk menampilkan faktur
echo '<table border="1" cellpadding=2>
 .<tr bgcolor="#cccccc">
 .<th>Tanggal Beli</th>
 .<th>Baterei CBA</th>
 .<th>Total Rp.</th>
 .<th>Baygin</th>
 .<th>Total Rp.</th>
 .<th>Silver King</th>
 .<th>Total Rp.</th>
 .<th>Sub Total + diskon</th>
 .</tr>';

for($index=0; $index<$jml_pembelian; $index++)
{
 // pecah setiap bagian menjadi array
 $bagian = explode(' -#- ', $faktur[$index]);

 // rubah variabel dari tipe string ke integer dengan intval()
 // supaya angkanya saja yang dicetak
 $bagian[1] = intval($bagian[1]); // x baterei
 $bagian[3] = intval($bagian[3]); // x baygin
 $bagian[5] = intval($bagian[5]); // x silver king
 $bagian[7] = intval($bagian[7]); // x subtotal
 // cetak ke layar
 echo '<tr>';
 for ($jml_bagian=0; $jml_bagian<8; $jml_bagian++)
 {
 echo '<td align="right">'.$bagian[$jml_bagian].'</td>';
 }
 echo '</tr>';
}

echo '</table>';
?>
</body></html>

```

Simpan script diatas dengan nama ***lihat\_faktur2.php***. Jalankan pada browser anda, output script diatas akan terlihat seperti gambar 1.18

Tanggal Beli	Baterei CBA	Total Rp.	Baygin	Total Rp.	Silver King	Total Rp.	Sub Total + diskon
09:46, 31 October	14	105000	6	13500	32	56000	148325
10:24, 31 October	20	150000	10	22500	30	52500	191250
10:24, 31 October	8	60000	10	22500	7	12250	80537

Gambar 1.18: output file *lihat\_faktur2.php*

Pahamilah array dengan baik terutama pada perintah loopingnya karena ini akan sering kita gunakan pada aplikasi berskala besar.

## PENJELASAN SCRIPT

Variabel \$faktur pada script diatas secara otomatis akan berubah menjadi array. Ini dikarenakan fungsi `file('faktur.txt')` akan membaca dan merubah setiap baris dari file menjadi array. Untuk mengetahui berapa banyak baris / pembelian yang ada pada file faktur.txt digunakan fungsi `count($faktur)`. Yang akan menghitung jumlah elemen dalam suatu array. Jumlah elemen array kita masukkan ke dalam variabel `$jml_pembelian`.

Untuk menangkap adanya kemungkinan tidak ada pembeli. Maka kita gunakan statemen if yang menguji jumlah dari pembelian yang tidak lain juga merupakan jumlah baris. Jika jumlahnya sama dengan nol maka tidak ada pembelian untuk kita kita munculkan pesan.

```
for($index=0; $index<$jml_pembelian; $index++)
```

Untuk mencetak setiap pembelian kita menggunakan looping for. Jadi perintah yang ada pada blok for ini akan diulang sebanyak `$jml_pembelian` yang tidak lain adalah banyaknya baris. Untuk mengidentifikasi setiap barang dan harga dari baris pembelian kita perlu memecahnya menjadi array lagi. Ini dimaksudkan agar data tersebut dapat dimaksukkan dalam tabel.

```
$bagian = explode(' #- ', $faktur[$index]);
.....
$bagian[1] = intval($bagian[1]); // x baterei
```

`$bagian` merupakan variabel array karena didalamnya ia menggunakan fungsi `explode` untuk memecah suatu string. Tanda pemisah yang digunakan adalah "  `#-` " dimana setiap text yang berada diantara tanda ini akan diubah menjadi variabel array. String yang akan dipecah dalam hal ini adalah isi dari variabel array `$faktur[index]` array yang sedang aktif dari looping for].

Karena yang ingin kita tampilkan pada kolom **baterei CBA, baygin, dan silver king** hanya angkanya saja. Maka kita menggunakan fungsi `intval()` untuk merubah variabel dari tipe string menjadi tipe integer sehingga hurufnya tidak ikut tercetak.

Isi dari `$bagian[1]` sebelumnya adalah `x baterei` (dimana `x` adalah jumlah baterei). Setelah diubah dengan fungsi `intval($bagian[1])` maka isi dari variabel `$bagian[1]` menjadi `x` saja (hanya berupa angka). Demikian juga untuk `$bagian[3], $bagian[5]` dan `$bagian[7]`.

```
for ($jml_bagian=0; $jml_bagian<8; $jml_bagian++)
```

Looping for diatas dimaksudkan agar kita tidak perlu capek-capek menulis

```
echo '<td align="right">' . $bagian[nomor index array]. '</td>';
```

selama delapan kali berturut-turut. Keep this life as simple as possible, right?. Maklum dulu saya juara I pidato bahasa Inggris ( tapi dalam mimpi... ).

## Kalender

27		28	29	30	31	1	2
3	4	5	6	7	8	9	
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

Sebelum melanjutkan materi pada hari kelima ini,

- Buatlah sebuah folder baru pada direktori C:\Apache2\htdocs\webmaster\Bab1
- Beri nama folder tersebut **hari-5**.

## Manipulasi String dan REGEX

Pada hari yang ke lima ini kita akan mempelajari tentang manipulasi string dan regular expression pada PHP. Setelah melalui hari ke lima ini diharapkan anda dapat menguasai tentang:

- Memformat String
- Menggabung dan Memecah String
- Menggunakan Fungsi-fungsi String PHP
- Menggunakan Regular Expression (REGEX)
- Menggunakan Ulang Kode

Pada materi kali ini anda dapat menggunakan Fungsi String dari PHP untuk memformat dan memanipulasi text. Fungsi ini akan sangat berguna pada banyak hal. Misalnya anda ingin membersihkan atau memformat input dari user sebelum dimasukkan ke dalam database(akan dibahas lebih lanjut pada materi selanjutnya).

### Memformat String

Anda akan sering harus memfilter string terutama dari Form HTML. Karena mungkin saja user keliru mengetikkan karakter dan sebagainya.

#### Menghilangkan Spasi: **trim()**, **ltrim()**, dan **chop()**

Ketika kita menyediakan suatu user input misalnya berupa text box. Text box tersebut kita peruntukkan untuk pengisian alamat email. Lalu user itu secara tidak sengaja menekan spasi tanpa ia ketahui lalu tetap menuliskan alamat emailnya. Jika kita simpan email user dalam database dan kita ingin mengirimkan email ke alamat tersebut maka email tidak akan terkirim karena format email tidak valid.

```
$a = " ada spasi diawal dan diakhir ";
```

pada contoh diatas variabel \$a pada awal dan akhir string terdapat spasi. Jika anda menjalankan perintah

```
echo $a;
```

Maka output pada browser hanya akan menampilkan satu spasi saja yaitu " ada spasi diawal dan diakhir " (tanpa tanda petik). Ini dikarenakan HTML mengabaikan karakter whitespace (banyak spasi).

Hal itu tidak masalah tetapi bagaimana jika kita menyimpan isi variabel \$a ke sebuah file atau memasukkannya ke dalam database. Ini yang menjadi masalah karena isi variabel \$a akan disimpan apa adanya. Untuk menghilangkan spasi diawal dan diakhir string kita gunakan fungsi **trim()**.

```
trim($a);
// isi variabel $a sekarang adalah "ada spasi diawal dan diakhir";
```

Untuk `ltrim()` hampir sama hanya saja `ltrim()` hanya menghilangkan spasi di sebelah kiri string sedangkan `chop()` di sebelah kanan string.

### PENERAPAN TEORI TRIM()

Untuk lebih memahami manipulasi string dengan `trim()`. Buatlah file HTML baru pada PHP DESIGNER 2006 dengan mengklik **File-New-HTML**. Ketik kode HTML dibawah ini

```
<html>
<head><title>Kontak Kami</title></head>
<body>
<h2>Kritik dan Saran Untuk MiniMarket Mr. Web Janarko</h2>

<form action="proses_kontak.php" method="post">
Nama Anda:

<input type="text" name="nama" size="30">

Email Anda:

<input type="text" name="email" size="30">

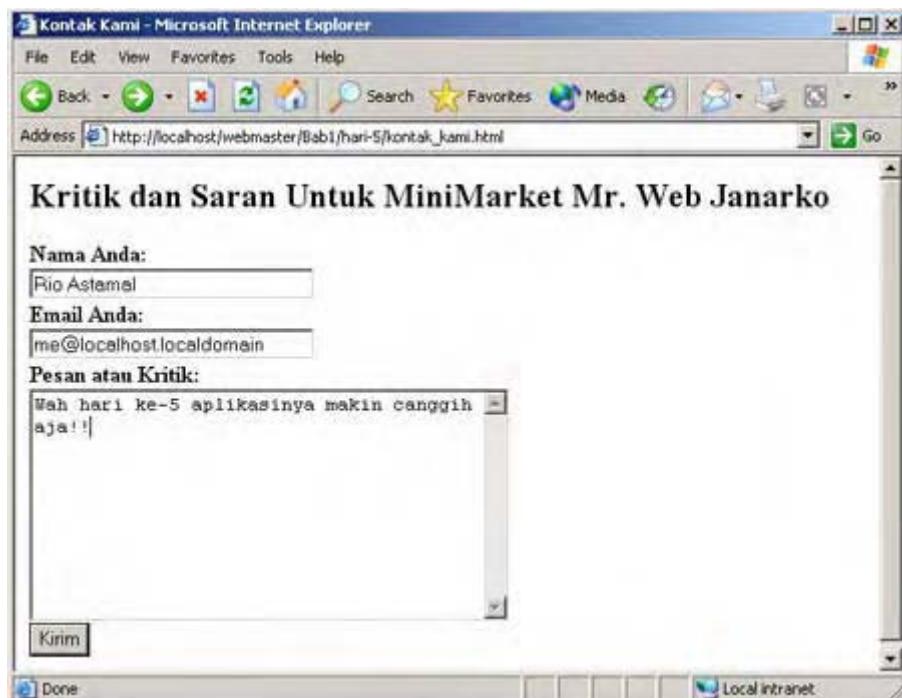
Pesan atau Kritik:

<textarea name="pesan" cols="40" rows="10"></textarea>

<input type="submit" value="Kirim">
</form>

</body>
</html>
```

Simpan kode diatas dengan nama ***kontak\_kami.html***. Sekarang buatlah file PHP baru dengan mengklik menu **File-New-PHP**. File ini akan memproses kritik dan saran dari file ***kontak\_kami.html***.



Gambar 1.19: output file kontak\_kami.html

File ini nantinya akan memproses data yang dikirim dari form kritik dan saran. Seiring dengan bertambahnya materi kita nanti akan menambahkan beberapa kode baru pada script ini.

```
<html>
<head><title>Kritik dan Saran</title></head>
<body>

<?php
```

```

//buat variabel untuk data yang dikirim
$nama = $_POST['nama'];
$email = $_POST['email'];
$pesan = $_POST['pesan'];

//trim() untuk menghilangkan spasi diawal dan diakhir
$nama = trim($nama);
$email = trim($email);
$pesan = trim($pesan);

//buat variabel untuk memudahkan pengiriman email
$subject = "Kritik dan Saran dari Konsumen";
$emailku = 'Web@janarko.com';
$isi_email = 'Nama Konsumen: '.$nama."\n"
 .'Email Konsumen: '.$email."\n"
 .'Pesanan atau Kritik: \n'.$pesan."\n";
$header = 'From: MiniMarket@online.com';

// kirim email dengan fungsi mail()
mail($emailku, $subject, $isi_email, $header);

echo '<h2>Pesan Terkirim</h2>';
?>

</body>
</html>

```

Simpan dengan nama **proses\_kontak.php**. Jalankan pada browser hasilnya akan seperti gambar 1.20.



Gambar 1.20: muncul pesan peringatan tidak memiliki program mail server (seperti hMailServer)

### PENJELASAN SCRIPT

Pertama kita buat variabel untuk menampung data yang dikirim lewat method POST. Selanjutnya kita hilangkan spasi diawal dan diakhir setiap data yang dikirim dengan fungsi trim().

Karena kritik dan saran ini akan dilayangkan ke Mr. Web Janarko selaku pemilik MiniMarket. Maka kita kirim pesan ini ke alamat email Mr. Web tersebut. Untuk mengirim email PHP menyediakan fungsi mail(). Struktur dasar penulisan fungsi mail() adalah sebagai berikut:

```
mail(string alamat, string subject, string isi email, [header], [parameter tambahan])
```

Tiga parameter pertama wajib untuk anda isi. Sedangkan dua yang terakhir sifatnya hanya opsional. Agar penulisan fungsi `mail()` lebih mudah dan tidak membingungkan maka kita buat variabel string untuk masing-masing parameter.

Variabel `$emailku`, `$subject`, `$isi_email` dan `$header` merupakan variabel-variabel yang akan kita masukkan pada fungsi `mail`. Ingat karakter `\n` merupakan garis baru dan hanya bisa berada pada petik dua. Header merupakan informasi tambahan pada email seperti `From` atau `Reply-To`. Kali ini kita hanya menggunakan informasi tambahan `From`. Sehingga penulisan fungsinya menjadi:

```
mail($emailku, $subject, $isi_email, $header);
```

Ketika kita mengklik tombol **Kirim** pada halaman saran dan kritik. Maka akan muncul pesan peringatan seperti ini

**Warning:** mail() [[function.mail](#)]: Failed to connect to mailserver at "localhost" port 25, verify your "SMTP" and "smtp\_port" setting in `php.ini` or use `ini_set()` in  
**I:\Apache2\htdocs\webmaster\Bab1\hari-5\proses\_kontak.php** on line **25**

Ini karena pada komputer kita tidak terdapat aplikasi Mail Server yang menghandel port 25 atau SMPT. Kita hanya berada pada webserver virtual. Pesan ini tidak akan muncul jika kita berada pada webserver yang sebenarnya.

Atau anda sudah menginstal program mail server seperti hMailServer. Untuk instalasi hMailServer lihat pada pendahuluan.

## Mengubah Format Huruf pada String

Anda juga dapat mengubah format dari huruf pada suatu string. Misalnya mengubah huruf kecil menjadi huruf kapital juga sebaliknya. Meskipun ini tidak banyak digunakan pada aplikasi kita nanti namun ada beberapa kasus yang dapat kita tangani dengan merubah format huruf pada string.

Misalnya pada input Nama seseorang mengisikan **nama dia**. Karena nama biasanya pada awal kata selalu diawali huruf kapital. Maka kita menggunakan fungsi

```
ucwords('nama dia'); //hasilnya Nama Dia
```

Berikut ini tabel fungsi-fungsi yang dapat anda gunakan dalam merubah format huruf string

**Tabel 1.11** Fungsi Pengubah String Pada PHP

Fungsi	Penjelasan	Penggunaan	Hasil
<code>strtoupper()</code>	Mengubah string ke huruf KAPITAL	<code>strtoupper(\$subject)</code>	KRITIK DAN SARAN DARI KONSUMEN
<code>strtolower()</code>	Mengubah string ke huruf kecil	<code>strtolower(\$subject)</code>	kritik dan saran dari konsumen
<code>ucfirst()</code>	Mengubah awal awal string ke huruf besar	<code>ucfirst(\$subject)</code>	Kritik dan saran dari konsumen
<code>ucwords()</code>	Mengubah setiap awal kata ke kapital	<code>ucwords(\$subject)</code>	Kritik Dan Saran Dari Konsumen

## **addslashes()** dan **stripslashes()**

User tidak hanya mengisi huruf biasa pada box pesan. Mereka mungkin ingin memasukkan karakter " (petik dua) atau ' (petik satu). Ketika kita akan memasukkan data ke database, karakter-karakter seperti ", ', dan karakter kosong (NULL karakter) akan menjadi masalah security jika kita tidak menanganinya dengan serius.

Kita harus menentukan cara agar karakter-karakter tersebut dapat dimengerti oleh database seperti MySQL. Jika anda ingin menulis tanda " dan ingin memasukkannya didatabase anda dapat menggunakan tanda \" (backslashes petik dua) untuk menggantinya. Dan jika ingin menampilkan tanda \ anda dapat menggantinya dengan tanda \\. Karakter karakter seperti \" dan \\ disebut sebagai **escaping character**.

Daripada menuliskannya seperti diatas kita dapat menggunakan fungsi dari PHP untuk mencetak escaping character. Yaitu dengan menggunakan addslashes( ).

```
$pesan = addslashes($pesan);
```

Fungsi addslashes( ) memerlukan satu parameter yaitu string yang akan di beri escaping character.

Saat anda menggunakan addslashes( ) untuk menyimpan data ke database. String tersebut akan diberi escaping karakter. Saat anda ingin menampilkan string dari database tersebut jangan lupa untuk menghilangkan tanda escaping karakter dengan menggunakan fungsi stripslashes( ).

```
$pesan = stripslashes($pesan);
```

String yang akan ditampilkan setelah penggunaan stripslashes( ) akan menghilangkan tanda \ dan escaping karakter lainnya.

### **strlen() dan empty()**

Script proses\_kontak.php memiliki banyak kekurangan diantaranya jika user tidak mengisi apa-apa script akan tetap memproses data. Ini pada aplikasi sesungguhnya jelas tidak mungkin. Oleh karena itu sebelum memproses data kita perlu mengetahui apakah user telah mengisi field atau belum.

strlen() digunakan untuk mengecek panjang suatu string. strlen() memiliki satu parameter yaitu string yang akan dicek panjangnya.

empty() digunakan untuk mengecek isi dari variabel. Jika isi dari variabel itu kosong maka nilainya true. Ini berguna jika kita ingin mengecek isi dari suatu variabel apakah kosong atau tidak. Seperti contoh berikut:

```
if (empty($nama))
{
 echo "Nama Tidak Boleh Kosong";
 exit;
}
```

### **PENERAPAN TEORI strlen()**

Jalankan PHP DESIGNER 2006 lalu buka file **proses\_kontak.php**. Ubahlah kode yang berada dalam tag <?php ... ?> sehingga menjadi seperti berikut.

```
<?php
//buat variabel untuk data yang dikirim
$nama = $_POST['nama'];
$email = $_POST['email'];
$pesan = $_POST['pesan'];

//cek panjang string
if (strlen($nama) < 3)
{
 echo "Masa, nama kok kurang dari 3 huruf";
 exit;
}
```

```

}

elseif(strlen($email) < 6)
{
 echo "Email anda tidak valid";
 exit;
}
elseif(strlen($pesan) < 10)
{
 echo "Komentar terlalu pendek";
 exit;
}

//trim() untuk menghilangkan spasi diawal dan diakhir
$nama = trim($nama);
$email = trim($email);
$pesan = trim($pesan);

//buat variabel untuk memudahkan pengiriman email
$subject = "Kritik dan Saran dari Konsumen";
$emailku = 'Web@janarko.com';
$isi_email = 'Nama Konsumen: '.$nama."\n"
 .'Email Konsumen: '.$email."\n"
 ."Pesanan atau Kritik: \n".$pesan."\n";
$header = 'From: Web@janarko.com';

//kirim email dengan fungsi mail()
mail($emailku, $subject, $isi_email,$header);

echo '<h2>Pesan Terkirim</h2>';
?>

```

Simpan kembali file tersebut lalu jalankan pada browser. Sekarang coba anda klik tombol **beli** tanpa mengisi apapun. Maka akan muncul pesan

### **Anda belum mengisi nama**

#### **PENJELASAN SCRIPT**

Mengapa kita tidak mengecek menggunakan fungsi `empty()`? Kita gunakan logika kita, jika mengecek menggunakan `empty()` dan selanjunya kita juga menggunakan `strlen()` untuk mengecek. Maka penggunaan `empty()` akan tidak efisien karena dengan `strlen()` kita juga bisa memastikan bahwa variabel itu terisi karakter atau tidak dengan menghitung panjang string pada variabel.

Untuk mengecek panjang string kita gunakan `strlen()`.

```

if (strlen($nama) < 3)
{
 echo "Masa, nama kok kurang dari 3 huruf";
 exit;
}

```

Sama seperti sebelumnya kita gunakan statemen `if...elseif...` untuk menentukan kondisi yang kita inginkan. Pada kode diatas kita tentukan panjang string `$nama` harus lebih dari 2. Untuk `$email` harus lebih dari 5 sedangkan untuk `$pesan` harus lebih dari 9. Jika kondisi-kondisi yang kita tentukan tadi tidak terpenuhi maka tampilkan pesan dan akhiri eksekusi script dengan `exit;`.

### **substr( )**

Dengan menggunakan `substr()` anda dapat mengakses string-string yang berada pada string tersebut (Sub String). Fungsi ini biasanya berguna jika kita ingin memformat suatu string kedalam bentuk lain. Struktur dasar penulisan fungsi `substr()` adalah

```
substr(string awal, angka untuk memulai string, [panjang string])
```

- Parameter pertama, kita harus menentukan string yang akan kita jadikan rujukan(string sumber)
- Parameter ke dua, merupakan angka dari 0 sampai banyaknya string dari string sumber yang ingin anda tampilkan.
- Paramater ke tiga(opsional), angka yang merupakan banyaknya string yang ingin anda tampilkan.

Lihatlah contoh berikut ini

```
$subject = "Kritik dan Saran dari Konsumen";
$format = substr($subject, 2);
echo $format;
```

Pada kode diatas string sumber adalah isi dari variabel `$subject`. Dan 2 merupakan posisi string yang akan dimulai. Output dari kode diatas adalah

`itik dan Saran dari Konsumen`

Huruf K dan r tidak diikutkan karena kita memasukkan angka 2 pada parameter string yang dimulai sehingga dua huruf pertama dilewatkan. Jika kita mengganti isi dari `$format` menjadi seperti berikut

```
$format = substr($subject, -8);
```

Artinya kita akan mencetak isi dari `$subject` mulai dari 8 karakter terakhir. Output dari `$format` diatas adalah

`Konsumen`

Lalu bagaimana jika saya hanya ingin mencetak 6 karakter awal saja?. Anda bertanya pada orang yang tepat. Untuk mencetak hanya sejumlah karakter saja mulai dari awal anda dapat menggunakan parameter yang ketiga.

```
$format = substr($subject, 0, 6);
```

O artinya kita mulai dari karakter yang paling awal yaitu K. 6 merupakan jumlah karakter yang ingin diambil. 6 karakter awal dalam variabel `$subject` berarti *Kritik*.

```
$format = substr($subject, 11, -9);
```

Kode diatas berarti ambil karakter antara karakter ke-11 dan karakter ke-9 dari akhir. Output dari kode diatas adalah ***Saran dari***.

## Format Angka: `number_format()`

Karena website yang kita buat berhubungan dengan "uang". Maka kita harus mempunyai format khusus untuk mata uang tersebut. Kita tidak mungkin melakukan perhitungan matematik jika isi variabel terdapat **Rp. xxx.xxx,-**. Kita hanya memerlukan angkanya saja, untuk penggantian format tulisan kita lakukan setelah diperoleh hasil akhir.

Untuk melakukan format mata uang kita dapat menggunakan fungsi `number_format()`. Karena kita hidup di negara yang sedang krisis yaitu Indonesia. Maka kita gunakan format mata uang rupiah. Struktur dasar penulisan fungsi `number_format()` adalah sebagai berikut.

```
number_format(angka yang dirubah, [angka dibelakang], [string pemisah angka dibelakang], [string pemisah ribuan])
```

- Parameter 1: sifatnya wajib diisi, karena berisi angka yang ingin anda rubah formatnya.
- Parameter 2(opsional): berapa banyak angka yang ingin anda tampilkan dibelakang angka, berupa angka 0.
- Parameter 3(opsional): string yang digunakan sebagai tanda pemisah antara angka yang diubah dengan angka dibelakang.
- Parameter 4(opsional): string pemisah antar bilangan ribuan.

**Catatan:** jika anda mengisi parameter 3 maka anda harus mengisi parameter 4. Nampaknya mereka saling mencintai hingga tak mau pisah.

Saya yakin anda pasti bingung membaca penjelasan diatas. Dan memang obat yang paling mujarab saat ini hanyalah **contoh**.

```
$angka = 1000000; //satu juta
echo number_format($angka, 2); //hasil -> 1,000,000.00
echo number_format($angka, 2, '@', ''); //hasil -> 1000000@00
echo number_format($angka, 2, '@', '.'); //hasil -> 1.000.000@00
echo number_format($angka, 0, '', '.'); //hasil -> 1.000.000
```

Dari contoh diatas rasanya format yang terakhir yang cocok untuk rupiah. Kita hanya perlu menambahkan

```
echo 'Rp '.number_format($angka, 0, '', '.').',-'; //hasil -> Rp. 1.000.000,-
```

Format ini yang akan kita gunakan terus pada aplikasi proyek kita nantinya.

## Mencari String: `strstr()`

Kadang kala anda ingin mencari suatu kata atau string pada string yang lain. Anda dapat menggunakan `strstr()` untuk keperluan tersebut. Struktur dasar penulisan `strstr` adalah

```
strstr(string asal, string yang dicari)
```

Lihatlah contoh berikut untuk memperjelas pemahaman anda tentang `strstr()`

```
$subject = 'Kritik dan Saran dari Konsumen';
if (strstr($subject, 'Kritik'))
{
 echo 'Kritik Konsumen';
}
```

Jika `strstr()` menemukan string yang kita cari ia akan mengembalikan nilai true jika tidak maka false. Hampir sama dengan `strstr()` terdapat fungsi lain untuk mencari yaitu `stristr()` hanya saja fungsi ini *tidak* membedakan huruf kecil dan besar.

Misalnya pada kata 'kritik' jika kita menggunakan fungsi `strstr()` seperti pada kode diatas maka hasilnya akan false karena `strstr()` membedakan huruf kecil dan huruf besar (case sensitive).

## Mengganti String: `str_replace()`

Mencari dan mengganti string merupakan fungsi yang sangat berguna. Dan itulah yang dimiliki `str_replace()`. Fungsi ini dapat mencari string dan menggantinya dengan karakter yang telah ditentukan sebelumnya. Struktur dasar penulisan fungsi ini adalah

```
str_replace(string yang dicari, string pengganti, string sumber, [jumlah yang dicari]);
```

### PENERAPAN TEORI STR\_REPLACE()

Jalankan PHP DESIGNER 2006 buka file **proses\_kontak.php**. Sisipkanlah kode berikut ini sebelum komentar `// trim()` untuk menghilangkan spasi diawal dan pada akhir script tambahkan `.$pesan` dibawah `echo '<b>Pesan Terkirim</b>'`

```
.....
//filter kata-kata kotor
$kotor = array('brengsek', 'fuck', 'sialan', 'bangsat', 'biadab');
foreach ($kotor as $kata)
{
 $pesan = str_replace($kata, '!@#$%', $pesan);
 echo 'Pesan anda tidak sopan
';
}

//trim() digunakan untuk menghilangkan spasi diawal dan diakhiri.
.....
echo '<h2>Pesan Terkirim</h2>'
 . $pesan;
```

### PENJELASAN SCRIPT

Variabel `kotor` merupakan array yang berisi kumpulan kata-kata **kotor** yang akan difilter menggunakan `str_replace()`. Pada kode diatas kata-kata yang difilter sebanyak lima kata yaitu *brengsek, fuck, sialan, bangsat, dan biadab*. Dan gantinya adalah string '`!@#$%`'.

Untuk mengulang pencarian terus menerus kita gunakan looping `foreach()`. Setiap kata pada array `$kotor` kita pecah dan masukkan pada variabel `$kata`. Jadi `$kata` isinya bisa salah satu dari kelima kata kotor tersebut. Selama masih ada kata kotor pada `$pesan` maka `str_replace($kata, '!@#$%', $pesan);` akan dieksekusi.

Untuk membuktikannya coba anda masukkan kata-kata kotor tersebut pada box kritik dan saran lalu tekan **kirim**. Lihatlah pada browser kata-kata kotor yang ada didaftar akan diubah menjadi `!@#$%`. Namun metode ini memiliki kelemahan karena masih membedakan huruf kecil dan huruf besar maka kata 'brengsek' dianggap berbeda dengan 'Brengsek'. Kita akan menutupi kelemahan ini dengan REGEX.

## Regular Expression (REGEX)

Regular Expression atau sering disebut juga REGEX. Merupakan suatu cara untuk mendeskripsikan pola dalam suatu text. Sebagai contoh kata 'kritik' cocok dengan dengan regex 'kritik'. Juga cocok dengan regex 'kri', 'ri', 'k' dan sebagainya.

Dengan regex anda dapat membuat pola yang lebih spesifik sesuai dengan keinginan anda. Pada contoh sebelumnya 'kritik' juga cocok dengan 'kri', 'ri', 'k' dan dapat juga '#k'. Untuk itu jika anda ingin membatasi karakter tertentu saja misalnya hanya karakter dari a sampai z gunakan:

`[a-z]`

Semua karakter yang berada didalam [...] hanya cocok dengan **satu karakter saja**. Anda juga dapat menentukan sekumpulan karakter seperti

[a-zA-Z]

akan cocok dengan segala huruf dari a sampai z atau A sampai Z(KAPITAL). Selain itu anda dapat menentukan sekumpulan karakter yang ingin dicocokkan. Anda dapat juga menentukan karakter yang tidak boleh ikut dari sekumpulan karakter.

[^a-zA-Z]

Artinya cocok dengan segala karakter kecuali a sampai z. Tanda ^ (caping) didalam kurung siku artinya **tidak**, tapi jika diluar kurung siku lain lagi. Kita akan segera mengetahuinya, Sabar OK.

## Pola Diawal atau Diakhir String

Anda dapat meletakkan pola khusus yang harus ada pada awal, diakhir atau bahkan keduanya. Wih..wih...wih... nampaknya bakalan njlimet nih! Jangan pesimis begitu santai saja.

Pola seperti ini sangat berguna terutama untuk mengecek kevalidan suatu string yang berpola aneh seperti email. Simbol ^ (caping) digunakan pada regex untuk menentukan pola yang harus ada pada awal string. Sebaliknya tanda \$ digunakan pada regex untuk menentukan pola yang harus ada pada akhir string

Contohnya, dibawah ini akan cocok dengan **web** pada awal string:

^web

dan ini akan cocok dengan **janarko** pada akhir dari string:

janarko\$

yang terakhir penggunaan ^ diluar kurung siku:

^[a-zA-Z]

pola diatas berarti cocok dengan satu karakter dari a sampai z pada awal string.

## Mencocokkan Karakter Spesial

Jika anda ingin mencocokkan karakter-karakter spesial seperti ., {, \$, " atau ' anda dapat menggunakan tanda \ (backslashes) didepan karakter spesial tersebut. Jika ingin menampilkan tanda \ maka anda harus menuliskannya \\.

Untuk daftar karakter spesial dapat anda lihat pada tabel 1.1.3 dan 1.1.4

**Tabel 1.12** Karakter Spesial REGEX diluar kurung siku

Karakter	Keterangan
\	escape character
^	sesuai dengan awal dari string
\$	sesuai dengan akhir dari string
.	cocok dengan semua karakter kecuali (\n) garis baru
	mulai dari tanda kurung (dibaca OR)
(	mulai dari subpattern
)	akhir dari subpattern
*	ulang 0 atau beberapa kali
+	ulang 1 atau beberapa kali

**Tabel 1.13** Karakter spesial REGEX didalam kurung siku

Karakter	Kerangan
\	Escape Character
^	Tidak diantara
-	menyatakan sekumpulan karakter

## REGEX Untuk Validasi Email

Dari beberapa kelebihan regex yang disebutkan diatas. Kita dapat menggunakan regex untuk proses validasi email pada form. Seperti kita tahu script *proses\_kontak.php* memiliki banyak kelemahan dalam mendeteksi kevalidan email. Dimana script tersebut hanya mengecek panjang string tidak mengecek awal string, tanda @, tanda . dan sebagainya.

Untuk mengecek kevalidan email kita dapat menggunakan pola regex seperti berikut:

```
^[a-zA-Z0-9_\-]+@[a-zA-Z0-9_-]+\.[a-zA-Z0-9_\-\.\.]+$
```

Pola pertama yaitu `^[a-zA-Z0-9_\-]+` artinya "email setidaknya diawali satu karakter yang berupa huruf, angka, \_(underscore), tanda – (minus), atau kombinasi dari semua itu". Kita menggunakan tanda \ didepan – karena minus merupakan karakter spesial.

Pola kedua `@[a-zA-Z0-9\_-]+` artinya "harus ada simbol @ lalu setidaknya ada huruf, angka atau tanda ( - ). Pola kedua ini untuk nama hosting.

Pola ketiga `\.[a-zA-Z0-9_\-\.\.]+$` artinya "harus ada tanda . lalu setidaknya ada huruf, angka, atau titik lain jika diperlukan sampai akhir dari string. Pola yang terakhir ini untuk .com, .co.id dan sebagainya.

Meskipun tidak 100% dapat memfilter email. Tapi pola seperti ini setidaknya akan mempermudah penanganan situasi.

## Mencari String dengan REGEX

Mencari string merupakan pekerjaan tujuan utama dari pembuatan pola regex. PHP memiliki dua fungsi untuk keperluan ini yaitu `ereg()` dan `eregi()`. Perbedaannya jika `ereg()` membedakan huruf kecil dan huruf besar(case sensitive). `eregi()` tidak membedakan huruf kecil dan huruf besar(not case sensitve). Fungsi `eregi()` memiliki struktur dasar seperti berikut.

```
eregi(string pola regex, string sumber, [array])
```

Fungsi akan mencari *pola regex* yang diinginkan pada *string sumber*. Parameter 3 bersifat opsional.

## PENERAPAN TEORI EREGI ()

Jalankan PHP DESIGNER 2006 buka file *proses\_kontak.php* yang pernah kita buat sebelumnya. Tambahkan kode validasi email berikut ini.

```
//cek panjang string
if (strlen($nama) < 3)
{
 echo "Masa, nama kok kurang dari 3 huruf";
 exit;
}
elseif(strlen($email) < 6)
{
 echo "Email anda tidak valid";
 exit;
}
```

```

elseif(strlen($pesan) < 10)
{
 echo "Komentar terlalu pendek";
 exit;
}

```

Tambah kode berikut dibawah dari kode diatas tadi.

```

//cek kevalidan email
if (!eregi('^[a-zA-Z0-9_-]+@[a-zA-Z0-9\-.]+\.[a-zA-Z0-9\-.]+\$', $email))
{
 exit('Email Tidak Valid');
}

```

### PENJELASAN SCRIPT

Saya rasa yang perlu saya jelaskan disini hanyalah penggunaan fungsi `eregi()` saja. Lihatlah cuplikan script berikut

```
if (!eregi('^[a-zA-Z0-9_-]+@[a-zA-Z0-9\-.]+\.[a-zA-Z0-9\-.]+\$', $email))
```

Kita gunakan statemen if untuk menentukan kondisi. Karena didepan `eregi` terdapat tanda ! (seru) yang berarti *jika ada kesalahan atau tidak cocok*. Jadi if akan mengeksekusi blok kode jika memang ada kesalahan atau pola tidak cocok dengan input dari user.

```
 exit('Email Tidak Valid');
```

Perintah `exit` juga dapat diisi parameter berupa string. Mode seperti diatas lebih singkat dibanding menulis dua perintah sekaligus yaitu `echo` dan `exit`.

## Mengganti String dengan REGEX

Anda dapat menggunakan cara yang sama dengan cara sebelumnya untuk mencari string dengan `str_replace()`. Tapi tentu saja dengan regex pola yang kita tentukan lebih spesifik dan mengena. Pada regex kita dapat menggunakan dua fungsi yaitu `ereg_replace()` dan `eregi_replace()`.

Perbedaannya sama seperti sebelumnya `ereg_replace()` membedakan huruf kecil dan huruf besar. Sedangkan `eregi_replace()` tidak. Struktur dasar penulisan `eregi_replace()` adalah:

```
eregi_replace(string pola, string pengganti, string sumber);
```

### PENERAPAN TEORI EREGI\_REPLACE()

Jalankan PHP DESIGNER 2006 buka file `proses_kontak.php`, gantilah blok kode pada foreach menjadi seperti berikut

```
$pesan = str_replace($kata, '!@#$%', $pesan);
```

ganti dengan:

```
$pesan = eregi_replace($kata, '!@#$%', $pesan);
```

Setelah anda mengganti kode tersebut. Kata-kata kotor yang menggunakan huruf besar pun dianggap masuk kategori. Sehingga difilter dan diganti dengan `!@#$%`.

## Menggunakan Ulang Kode

Prinsip kerja programmer adalah kerja seefisien mungkin dengan waktu, tenaga dan biaya seminimal mungkin. Untuk itu daripada menulis kode baru kita lebih baik menggunakan ulang kode yang pernah kita tulis. Anda malas kali ni...?. Ini bukan masalah malas atau tidak.

Dengan menggunakan ulang kode yang pernah kita buat kita akan banyak menghemat waktu dan tenaga kita.

PHP menyediakan dua fungsi yang dapat kita gunakan untuk memanggil ulang kode kita, yaitu fungsi **require()** dan **include()**. Mari kita bedah satu persatu. Maaf dok ruang operasi belum siap!. Pake ruangan **contoh** saja.

## Menggunakan require() dan include()

Untuk memperjelas pemahaman anda kita akan membuat tiga file yaitu file utama.php, require.php, dan include.php. Jalankan PHP DESIGNER 2006 klik **File-New-PHP**. Ketik kode berikut ini,

```
<?php

require('require.php');
echo 'Saya adalah File utama.
';
include('include.php');

?>
```

Simpan dengan nama ***utama.php***. Lalu buatlah kode PHP baru, ketik kode berikut ini

```
<?php
echo 'Saya adalah File require.
';
?>
```

Simpan dengan nama ***require.php***. Sekali lagi buatlah file baru ketik kode berikut ini

```
<?php
echo 'Saya adalah File include.
';
?>
```

Simpan dengan nama ***include.php***. Jangan lupa simpan ketiga file diatas pada folder **hari-5**. Jalankan file utama.php pada browser hasilnya akan seperti gambar 1.21



Gambar 1.21: Output file utama.php

### PENJELASAN SCRIPT

**require()** dan **include()** keduanya memiliki satu parameter yaitu *string nama file*. Keduanya tampaknya sama lantas apa yang membedakan?. Yang membedakan adalah jika ada kesalahan misalnya file tidak ada maka **require()** akan memberikan pesan ***fatal error***. Sedangkan **include()** hanya memberikan pesan ***warning***.

---

Satu lagi `include()` dapat mengembalikan nilai sedangkan `require()` tidak. Pengembalian nilai mungkin berguna pada beberapa situasi seperti berikut.

```
$hasil = include('nama_file.php');
if ($hasil === true) //jika file ada
{
 //eksekusi kode
}
```

Hal seperti diatas tidak dapat anda lakukan jika menggunakan `require()`. Fungsi yang paling penting dari kedua fungsi ini adalah kita dapat memanggil class dan fungsi yang berada pada file lain. Sehingga kita tidak perlu menulis ulang kode yang sama.

---

## Kalender

27	28	29	30	31	1	2	
3	4	5	6	7	8	9	
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

## Proyek Aplikasi Bab 1 – MiniMarket Online

Setelah menguasai teknik-teknik dasar PHP saatnya kita membuat website yang lebih terstruktur dan dinamis meskipun belum ada database. Proyek yang akan kita kerjakan masih sama yaitu website MiniMarket Online milik pengusaha asal kota cyber yaitu Mr. Web Janarko.

Namun kita akan membangun website ini lebih kompleks lagi. Dengan menambahkan beberapa fitur baru.

Fitur-fitur yang terdapat pada website kita nantinya adalah:

- Fasilitas kasir online
- Pencatatan faktur
- Pencatatan pesanan untuk pengiriman
- Buku tamu
- Kontak kami
- Halaman administrator untuk melihat faktur dan kontak

Untuk itu ikuti langkah-langkah berikut

- Buat folder baru pada direktori C:\Apache2\htdocs\webmaster\Bab1.
- Beri nama folder baru tersebut **hari-6**.
- Lalu masuk pada folder C:\Apache2\htdocs\webmaster\Bab1\hari-6
- Buat folder baru lagi, namai folder tersebut **Proyek-Bab1**.
- Masuk pada folder C:\Apache2\htdocs\webmaster\Bab1\Proyek-Bab1
- Buat folder baru, beri nama folder tersebut **admin**

File-file yang akan kita buat untuk membangun website minimarket online ditunjukkan oleh tabel 1.14.

**Tabel 1.14** File-file pembangun website Minimarket online.

<b>Direktori Proyek-Bab1</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.php	untuk menampilkan halaman utama(pembelian)
proses_beli.php	untuk memproses form pembelian
header.php	untuk menampilkan bagian atas website
footer.php	untuk menampilkan bagian bawah website
kontak.php	untuk menampilkan form kontak
proses_kontak.php	untuk memproses form kontak
buku_tamu.php	untuk menampilkan daftar buku tamu
isi_bt.php	untuk menampilkan form buku tamu
proses_bt.php	untuk memproses form buku tamu

<b>Direktori Proyek-Bab1\admin</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.html	untuk menampilkan pesan peringatan
header_admin.php	untuk menampilkan bagian atas halaman admin
lihat_faktur.php	untuk menampilkan daftar faktur pembelian
lihat_kontak.php	untuk menampilkan data kontak yang masuk
faktur.txt	untuk menyimpan daftar faktur pembelian
kontak.php	untuk menyimpan data kontak kami
buku_tamu.txt	untuk menyimpan data buku tamu

Wah...banyak sekali!, iya memang. Itu karena kita masih dalam tahap awal pembelajaran. Setelah kita mengenal database dan teknik-teknik PHP tingkat lanjut jumlah file kita akan lebih

sedikit. Cukup basa-basinya langsung saja jalankan PHP DESIGNER 2006 untuk mengetik kode pertama kita yaitu header.php. Klik **File-New-PHP** lalu ketik kode berikut ini.

### Script dari file **header.php**

---

```
<html>
<head>
<title>MiniMarket Online - Mr. Web Janarko</title>
<style type="text/css">
 .judul{font-family: comic sans ms; font-size: 32; font-weight: bold}
 td{font-family: verdana; font-size: 11}
 a{font-family: comic sans ms; font-size: 16; font-weight: bold; text-decoration: none}
 a:hover{color: #ff6600; font-size: 18}
 h1{font-family: comic sans ms; font-size: 22; font-weight: bold}
 body{font-family: verdana; font-size: 12}
</style>
</head>
<body>

<table border="0" width="100%">
<tr bgcolor="#ff6600">
 <td class="judul">MiniMarket Online</td></tr>
</table>
<table border="0">
<tr bgcolor="#cccccc">
 <td>Home</td>
 <td>Kontak Kami</td>
 <td>Buku Tamu</td>
 <td width="70%"></td></tr>
</table>
```

---

Simpan dengan nama **header.php**. Jangan lupa letakkan pada folder **Proyek-Bab1**. File header.php hanya berisi kode HTML saja tanpa ada kode PHP. Mungkin anda merasa kurang familiar dengan kode-kode yang berada pada tag:

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

Jika anda masih memerlukan penjelasan panjang lebar mengenai CSS. Silahkan download e-book "Mastering Kode HTML". Disana terdapat penjelasan yang lengkap tidak hanya CSS namun juga HTML secara lebih mendetail.

Selanjutnya bualah file baru lagi, ketik kode dibawah ini untuk membuat file footer.php.

### Script dari file **footer.php**

---

```
<p></p>
<table border="0" width="100%">
 <tr bgcolor="#ff6600">
 <td align="right"><h1>(c) Copyright 2005 - Mr. Web Janarko</h1></td>
 </tr>
</table>

</body>
</html>
```

---

Simpan dengan nama **footer.php**. File ini sangat singkat karena fungsinya hanya untuk menampilkan bagian bawah website. Lanjut....!, buat file baru kembali ketik kode berikut ini untuk membuat file index.php.

---

## Script dari file **index.php**

---

```

<?php
 include('header.php');
?>

<center>
<h1>Selamat Datang</h1>
<p>Silahkan pilih barang kesukaan anda lalu klik Beli</p>
<form method="post" action="proses_beli.php">
<table border="0">
 <tr bgcolor="#ff6600">
 <td colspan="3" align="center">PEMBELIAN</td></tr>
 <tr bgcolor="#ff6600">
 <td>Barang</td>
 <td>Harga</td>
 <td>Jumlah</td></tr>
 <tr>
 <td>Baterei CBA</td>
 <td>Rp. 7500</td>
 <td><input type="text" name="baterei" maxlength="3" size="3"></td></tr>
 <tr>
 <td>Baygin</td>
 <td>Rp. 2250</td>
 <td><input type="text" name="baygin" maxlength="3" size="3"></td></tr>
 <tr>
 <td>Silver King</td>
 <td>Rp. 1750</td>
 <td><input type="text" name="silver" maxlength="3" size="3"></td></tr>
 <tr>
 <td colspan="3">Alamat Anda:</td></tr>
 <td colspan="3"><input type="text" name="alamat" size="25"></td></tr>
 <tr>
 <td colspan="3"><input type="submit" value="B e l i"></td></tr>
</table>
</form></center>

<?php
 include('footer.php');
?>

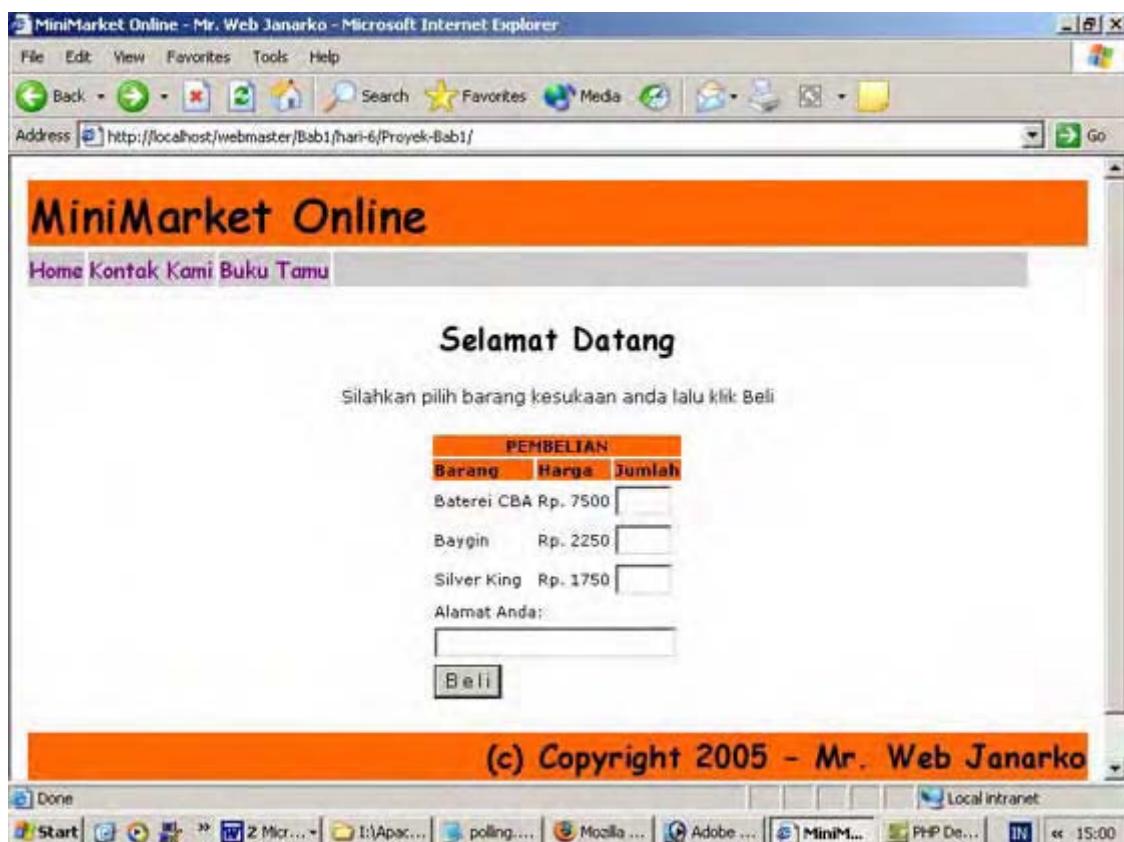
```

---

Simpan script diatas dengan nama **index.php**. Mengapa diberi nama index.php bukan lainnya. Jika pada suatu direktori tidak terdapat file **index.php** atau index.html maka server akan menampilkan daftar file yang ada pada direktori tersebut.

Contohnya seperti direktori **hari-5** atau direktori lainnya. Dimana didalamnya tidak terdapat file **index.html** atau **index.php**. Oleh karena itu yang ditampilkan adalah daftar file yang ada pada direktori tersebut.

Pada file index.php kita memanggil file header.php dengan menggunakan **include('header.php')**. Jadi file index.php nantinya bagian atasnya akan sama dengan tampilan file header.php. Hal sama juga berlaku untuk file footer.php. Output file index.php ditunjukkan oleh gambar 1.22.



Gambar 1.22: output file index.php

Untuk memproses form pembelian yang ada pada halaman utama kita akan membuat file PHP baru bernama proses\_beli.php. Anda juga dapat mengedit file proses\_beli.php yang pernah anda buat sebelumnya hingga menjadi seperti berikut.

#### Script dari file proses\_beli.php

```
<?php
include('header.php');

// buat variabel untuk menampung data dari minimarket.html
$baterei = $_POST['baterei'];
$baygin = $_POST['baygin'];
$silver = $_POST['silver'];
$alamat = $_POST['alamat'];

buat konstanta harga masing-masing produk
define('HARGA_BTR', 7500);
define('HARGA_BYN', 2250);
define('HARGA_SLR', 1750);

// cek alamat
if (strlen($alamat) < 8)
{
 echo "<p>Alamat anda kurang lengkap</p>";
 include('footer.php');
 exit;
}

$baterei_ttl = $baterei * HARGA_BTR;
$baygin_ttl = $baygin * HARGA_BYN;
$silver_ttl = $silver * HARGA_SLR;
$total = $baterei_ttl + $baygin_ttl + $silver_ttl;
```

```

// cek apakah total kurang dari atau sama dengan 0
if ($total <= 0)
{
 echo '<p>Anda tidak membeli apa-apa.</p>';
 include('footer.php');
 exit; //keluar dari program
}

if ($total > 0 && $total < 25000)
{
 $diskon = 0; //diskon 0%
 $pesan_diskon = 'Diskon 0%';
}
elseif ($total >= 25000 && $total <= 49000)
{
 $diskon = 0.05; //diskon 5%
 $pesan_diskon = 'Diskon 5%';
}
elseif ($total >= 50000 && $total <= 74000)
{
 $diskon = 0.1; //diskon 10%
 $pesan_diskon = 'Diskon 10%';
}
elseif ($total >= 75000)
{
 $diskon = 0.15; // diskon 15%
 $pesan_diskon = 'Diskon 15%';
}
$subtotal = $total - ($total * $diskon);
?>

Pembelian Terproses<hr>
<p>Berikut ini faktur pembelian anda</p>

<table border="0">
<tr bgcolor="#cccccc">
 <td align="center">Barang</td>
 <td align="center">Jumlah Beli</td>
 <td align="center">Bayar</td></tr>

<?php
echo '<tr>' .
 '<td>Baterei CBA</td>' .
 '<td align="center">'. $baterei .'</td>' .
 '<td>'. $baterei_ttl .'</td></tr>' .
 '<tr>' .
 '<td>Baygin</td>' .
 '<td align="center">'. $baygin .'</td>' .
 '<td>'. $baygin_ttl .'</td></tr>' .
 '<tr>' .
 '<td>Silver King</td>' .
 '<td align="center">'. $silver .'</td>' .
 '<td>'. $silver_ttl .'</td></tr>' .
 '</table>' .
 "<p>TOTAL: Rp. ". number_format($total, 0, ',', '.') ."-
" .
 "$pesan_diskon
" .
 "SUBTOTAL SETELAH DISKON: Rp. " .
 number_format($subtotal, 0, ',', '.') ."-</p>" ;

$tgl = date('H:i, j F');
$isi_faktur = "$tgl -#- $baterei baterei -#- $baterei_ttl -#- $baygin baygin -#-
$baygin_ttl -#- $silver silver king -#- $silver_ttl -#- $subtotal Subtotal -#-
$alamat\n";

```

```

$buka_file = fopen('admin/faktur.txt', 'a');

if (!$buka_file)
{
 echo "<p>Maaf, Pembelian anda tidak dapat diproses. Mohon coba lagi
 nanti</p><body></html>";
 exit;
}

// tulis ke file dengan fwrite lalu tutup dengan fclose
fwrite($buka_file, $isi_faktur);
fclose($buka_file);

echo '<p>Faktur telah tersimpan</p>'
 . "<p>Barang akan segera kami antar ke alamat $alamat dalam 1 jam.</p>";

include('footer.php');
?>

```

## PENJELASAN SCRIPT

Tidak ada yang banyak berubah dari script yang pernah kita buat pada hari sebelumnya. Yang kita tambahkan adalah fungsi `number_format()` dan `$isi_faktur` kita tambahkan variabel `$alamat` karena pada form terdapat field isian alamat. Kita akan mengirim barang ke alamat tersebut dengan menggunakan alat transportasi baru abad 21 yaitu Web Flash Car(bercanda...). Hal baru yang kita tambahkan adalah fungsi `include()` tentunya.

Selanjutnya kita akan membuat file kontak.php yang menampilkan form kontak kami. Untuk itu buatlah file PHP baru pada PHP DESIGNER 2006. Ketik kode berikut ini

### Script dari file **kontak.php**

---

```

<?php
 include('header.php');
?>

<h1>Kritik dan Saran Untuk MiniMarket Mr. Web Janarko</h1>

<form action="proses_kontak.php" method="post">
Nama Anda:

<input type="text" name="nama" size="30">

Email Anda:

<input type="text" name="email" size="30">

Pesan atau Kritik:

<textarea name="pesan" cols="40" rows="10"></textarea>

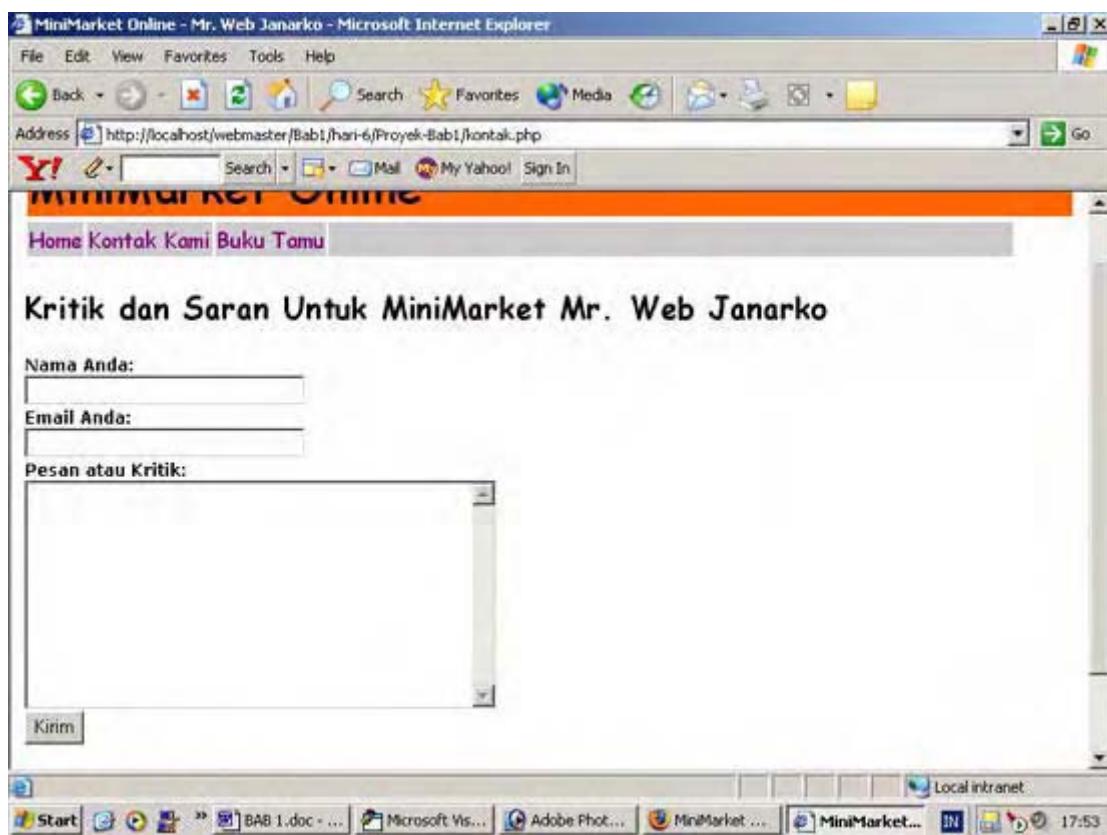
<input type="submit" value="Kirim">
</form>

<?php
 include('footer.php');
?>

```

---

Simpan dengan nama **kontak.php**. Output dari script diatas akan terlihat seperti gambar 1.23.



Gambar 1.2.4: output file kontak.php

Form kontak sudah ada. Selanjutnya kita akan membuat file yang akan memproses data yang dikirim dari form kontak.php. Buatlah file PHP baru dengan mengklik **File-New-PHP** lalu salin kode berikut ini.

#### Script dari file **proses\_kontak.php**

---

```
<?php
 include('header.php');

 // buat variabel untuk data yang dikirim
 $nama = $_POST['nama'];
 $email = $_POST['email'];
 $pesan = $_POST['pesan'];

 // cek panjang string
 if (strlen($nama) < 3)
 {
 echo "Masa, nama kok kurang dari 3 huruf";
 exit;
 }
 if(strlen($pesan) < 10)
 {
 echo "Komentar terlalu pendek";
 exit;
 }
 // cek kevalidan email
 if (!eregi('^[a-zA-Z0-9\-_]+@[a-zA-Z0-9\-_]+\.[a-zA-Z0-9\-_\.]+\$', $email))
 {
 exit('Email Tidak Valid');
 }

 // filter kata-kata kotor
 $kotor = array('brengsek', 'fuck', 'sialan', 'bangsat', 'biadab', 'shit');
 foreach ($kotor as $kata)
```

```

{
 $pesan = eregi_replace($kata, '!@#$%', $pesan);
}
// trim() untuk menghilangkan spasi diawal dan diakhir
$nama = trim($nama);
$email = trim($email);
$pesan = trim($pesan);

// ganti karakter ENTER dengan

$pesan = str_replace("\n", "
", $pesan);

$tgl = date('H:i, d-m-Y');
$isi_kontak = "$tgl -#- $nama -#- $email -#- $pesan\n";

// catat data kontak ke file kontak.txt
$buka_file = fopen('admin/kontak.txt', 'a');

if (!$buka_file)
 exit('Tidak dapat menulis ke file');

fwrite($buka_file, $isi_kontak);
fclose($buka_file);

echo '<h1>Saran atau kritik anda telah tersimpan</h1>';
. '<p>Terima Kasih</p>';

include('footer.php');
?>

```

Simpan dengan nama **proses\_kontak.php**. Kita tidak memasukkan fungsi `mail()` karena kita belum di server sebenarnya. Lagipula jika dengan `mail()` kita juga harus terhubung dengan internet untuk membaca email yang masuk. Lebih baik kita membacanya dari website kita langsung karena akan lebih cepat.

Pada script diatas setiap karakter `\n` pada pesan yang dihasilkan user karena penekanan tombol ENTER akan kita rubah menjadi format html yaitu `<br>`. Untuk bagian script lainnya masih sama dengan sebelum-sebelumnya.

Berikutnya kita akan membuat form buku tamu. Form ini banyak kita temui di berbagai website. Sekarang kita coba untuk membuatnya. Namun biasanya buku tamu yang kompleks menggunakan database. Ini kita akan praktekkan pada bab-bab selanjutnya. Namun untuk sekarang kita akan menggunakan file biasa sebagai media penyimpanan buku tamu. Ketik script dibawah ini lalu simpan dengan nama **buku\_tamu.php**

Script dari file **buku\_tamu.php**

```

<?php
 include('header.php');
?>

<h1>Buku Tamu - MiniMarket Online</h1>
Belum mengisi, isi SEKARANG!<hr>
<p>DAFTAR BUKU TAMU</p>

<?php
$buku_tamu = file('admin/buku_tamu.txt');
// hitung jumlah array
$jml_posting = count($buku_tamu);
// jika $jml_posting == 0 maka tidak ada yang posting
if ($jml_posting == 0)
{

```

```

echo "<p>Tidak ada posting untuk saat ini.</p>";
include('footer.php');
exit;
}

// buat tabel untuk masing-masing posting
for($index=0; $index<$jml_posting; $index++)
{
 //pecah setiap bagian array
 $bagian = explode(' -#- ', $buku_tamu[$index]);

 echo '<table border="0" cellpadding="4" width="70%">
 <tr bgcolor="#ff6600">
 <td>Di-post pada: ' . $bagian[0]
 </bold></td></tr>
 <tr>
 <td>Oleh: ' . $bagian[1] . '</td></tr>
 <tr>
 <td>Email: ' . $bagian[2] . '</td></tr>
 <tr>
 <td><p>Komentar: </p>' . $bagian[3] . '</td></tr>
 </table>
 <p><hr></p>; //untuk memberi jarak antar komentar user
}

include('footer.php');
?>

```

## PENJELASAN SCRIPT

Pada halaman buku\_tamu.php kita akan langsung menampilkan komentar-komentar yang sudah di-post. Juga terdapat link untuk mengisi buku tamu bagi yang belum yaitu link ***isi Sekarang!***. Jika pada file buku\_tamu.txt yang berada di direktori admin isinya kosong maka tidak ada yang posting. Cuplikan kode yang mengontrol kondisi ini adalah

```

if ($jml_posting == 0)
{
 echo "<p>Tidak ada posting untuk saat ini.</p>";
 include('footer.php');
 exit;
}

```

Masing-masing komentar dari user kita letakkan pada tabel sendiri-sendiri. Oleh karena itu perintah untuk menampilkan tabel yaitu echo '<table> ... </table>' kita letakkan pada blok kode for. Sehingga pembuatan tabel akan diulang sebanyak jumlah posting yang ada.

\$bagian merupakan array dari setiap data yang dipisahkan oleh tanda -#- sama seperti sebelum-sebelumnya. \$bagian[0] merupakan array paling kiri yaitu tanggal, \$bagian[1] merupakan nama pengirim dan \$bagian[2] merupakan komentar. Format penulisan ini pada file yang memproses form buku tamu yaitu proses\_bt.php.

Jika user mengklik link ***isi Sekarang*** maka akan muncul form isian buku tamu. Untuk membuat form isian buku tamu buat file PHP baru. Ketik kode berikut ini.

### Script dari file **isi\_bt.php**

```

<?php
 include('header.php');
?>

```

```

<h1>Isi Buku Tamu</h1>
<form action="proses_bt.php" method="post">
Nama:

<input type="text" name="nama">

Email:

<input type="text" name="email">

Komentar:

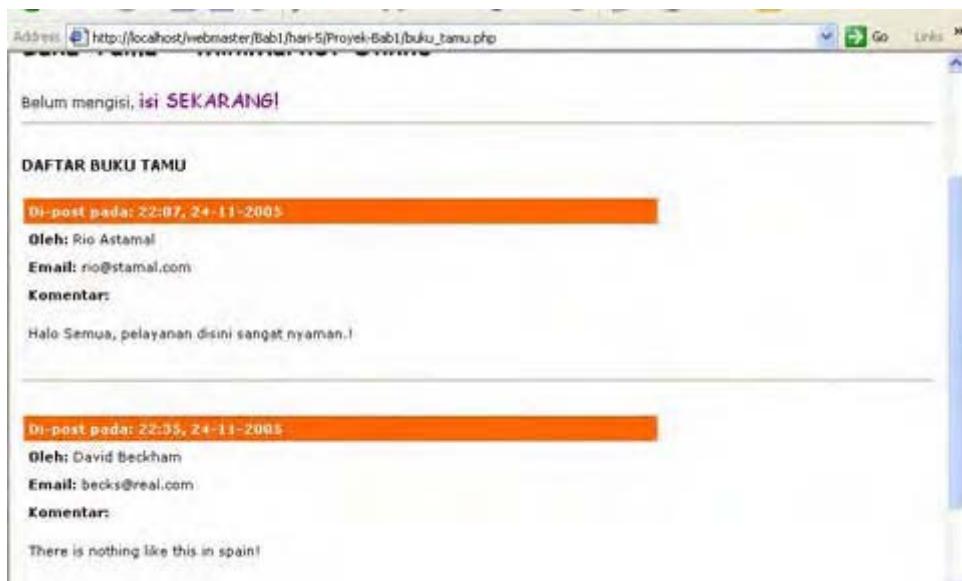
<textarea name="komentar" cols="50" rows="8"></textarea>

<input type="submit" value="KIRIM">
</form>

<?php
 include('footer.php');
?>

```

Simpan dengan nama ***isi\_bt.php***. Tidak ada yang spesial pada script ini jadi saya tidak perlu menjelaskannya OK. Selanjutnya kita akan membuat file untuk memproses form isian buku tamu yaitu file ***proses\_bt.php***



Gambar 1.2.5: output file buku\_tamu.php

### Script dari file ***proses\_bt.php***

```

<?php
include('header.php');

// buat variabel dari data yang dikirim
$nama = $_POST['nama'];
$email = $_POST['email'];
$komentar = $_POST['komentar'];

if (strlen($nama) < 3)
 exit('<p>Nama terlalu pendek</p>');

if (strlen($komentar) < 10)
 exit('<p>Komentar terlalu pendek</p>');

if (!eregi('^[a-zA-Z0-9_-]+@[a-zA-Z0-9\-.]+\.[a-zA-Z0-9\-.]+\$', $email))
 exit('<p>Email tidak valid</p>');

// filter kata-kata kotor
$kotor = array('brengsek', 'fuck', 'sialan', 'bangsat', 'biada', 'shit');

```

```

foreach ($kotor as $kata)
 $komentar = eregi_replace($kata, '!@#$%', $komentar);

// trim() untuk menghilangkan spasi diawal dan diakhir
$nama = trim($nama);
$email = trim($email);
$komentar = trim($komentar);

// ganti karakter ENTER dengan

$komentar = str_replace("\n", "
", $komentar);

$tgl = date('H:i, d-m-Y');
$buku_tamu = "$tgl -#- $nama -#- $email -#- $komentar\n";

// catat data kontak ke file kontak.txt
$buka_file = fopen('admin/buku_tamu.txt', 'a');

if (!$buka_file)
 exit('Tidak dapat menulis ke file');

fwrite($buka_file, $buku_tamu);
fclose($buka_file);

echo '<h1>Komentar anda berhasil di-post</h1>' .
 '<p>Terima Kasih</p>';

include('footer.php');
?>

```

---

## PENJELASAN SCRIPT

Hampir sama dengan dengan proses\_kontak.php file proses\_bt.php juga akan merubah karakter \n akibat tombol ENTER pada komentar dengan format html untuk garis baru yaitu <br>. Hal ini berguna agar tampilan yang keluar sesuai dengan keinginan user.

File buku\_tamu.php kita letakkan pada folder admin. Jadi pada `fopen()` kita menuliskan 'admin/buku\_tamu.php'.

Akhir dari file yang berada pada direktori **Proyek-Bab1**

---

## File-File untuk Administrator

File-file ini hanya bertujuan untuk menampilkan faktur dan kritik atau pesan dari konsumen. Karena fokus kita adalah database maka untuk manajemen administrator yang kompleks akan kita bahas pada bab-bab selanjutnya.

Simpanlah file-file adiministrator ini pada direktori **admin**. Agar daftar file pada direktori admin tidak dapat dilihat secara langsung maka buatlah file **index.html**. File ini hanya bertujuan untuk menampilkan pesan "bohongan". Untuk membuatnya klik **File-New-HTML/XHTML**

Script dari file **index.html**

---

```

<html>
<head>
<title>Akses Ditolak - Admin Page</title>
</head>
<body>
<h1>Akses Ditolak</h1>
</body>

```

---

```
<html>
```

Simpan pada direktori **admin** dengan nama **index.html**. Hanya itu? Ya namanya juga "boongan". Tujuan satu-satunya adalah agar server tidak menampilkan daftar file yang ada di direktori admin. Jika anda mengakses alamat <http://localhost/webmaster/Bab1/hari-5/Proyek-Bab1/admin/> maka yang muncul adalah pesan **Akses Ditolak**.

Untuk bagian atas website file ini kita namakan **header\_admin.php**. File lain yang membutuhkan file ini cukup memanggil dengan include().

#### Script dari file **header\_admin.php**

---

```
<html>
<head>
<title>MiniMarket Online - Mr. Web Janarko</title>
<style type="text/css">
 td{font-family: verdana; font-size: 11}
 th{font-family: verdana; font-size: 11; color: white}
 h1{font-family: comic sans ms; font-size: 22; font-weight: bold}
 body{font-family: verdana; font-size: 12}
</style>
</head>
<body>

<table border="0" width="100%">
<tr bgcolor="#ff6600">
 <td class="judul"><h1>Admin Area - MiniMarket Online</h1></td></tr>
</table>
<table border="0">
<tr bgcolor="#cccccc">
 <td>Lihat Faktur</td>
 <td>Lihat Kontak</td>
 <td>Edit Buku Tamu</td>
 <td width="70%"></td></tr>
</table>
<p></p>
```

---

Simpan pada direktori **admin** dengan nama **header\_admin.php**. Isi dari file hanya berisi kode HTML saja.

Untuk menampilkan faktur pembelian dari konsumen kita akan membuat file PHP baru yang bernama **lihat\_faktur.php**. Script ini hampir sama dengan yang pernah kita buat pada hari ke-3. Seperti biasa buatlah file PHP baru pada PHP DESIGNER 2006 klik **File-New-PHP** lalu ketik listing kode berikut

#### Script dari file **lihat\_faktur.php**

---

```
<?php
include('header_admin.php');

// buka file faktur.txt dengan perintah file() bukan readfile()
// file() akan memecah setiap baris atau pembelian menjadi array
$faktur = file('faktur.txt');
//hitung jumlah array
$jml_pembelian = count($faktur);
//jika $jml_pembelian == 0 maka tidak ada pembelian
if ($jml_pembelian == 0)
{
 echo "<p>Waduh, Kok sepi tidak ada yang beli</p>";
}
```

---

```

// buat tabel untuk menampilkan faktur
echo '<table border="0" cellpadding=2>
 .<tr bgcolor="#ff6600">
 .<th>Tanggal Beli</th>
 .<th>Baterei CBA</th>
 .<th>Total Rp.</th>
 .<th>Baygin</th>
 .<th>Total Rp.</th>
 .<th>Silver King</th>
 .<th>Total Rp.</th>
 .<th>Sub Total + diskon</th>
 .<th>Alamat</th>
 .</tr>';

for($index=0; $index<$jml_pembelian; $index++)
{
 // pecah setiap bagian menjadi array
 $bagian = explode(' -#- ', $faktur[$index]);
 // rubah variabel dari tipe string ke integer dengan intval()
 // supaya angkanya saja yang dicetak
 $bagian[1] = intval($bagian[1]); // x baterei
 $bagian[3] = intval($bagian[3]); // x baygin
 $bagian[5] = intval($bagian[5]); // x silver king
 $bagian[7] = intval($bagian[7]); // x subtotal
 // cetak ke layar
 echo '<tr>';
 for ($jml_bagian=0; $jml_bagian<9; $jml_bagian++)
 {
 echo '<td align="right">'.$bagian[$jml_bagian].'</td>';
 }
 echo '</tr>';
}

echo '</table>';

include('../footer.php');
?>

```

Jangan lupa simpan di direktori **admin** dengan nama ***lihat\_faktur.php***. Untuk mengakses halaman ini arahakan browser anda ke [http://localhost/webmaster/Bab1/hari-5/Proyek-Bab1/admin/lihat\\_faktur.php](http://localhost/webmaster/Bab1/hari-5/Proyek-Bab1/admin/lihat_faktur.php).

### PENJELASAN SCRIPT

Saya akan menjelaskan beberapa bagian script yang mungkin masih mengganjal pada pikiran anda.

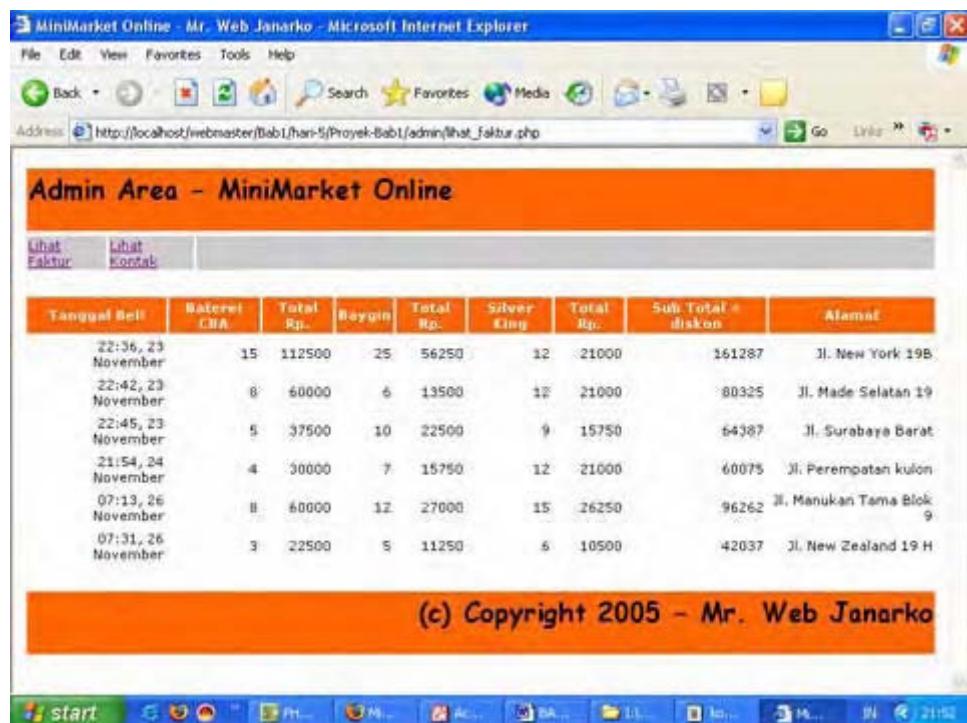
```
$bagian = explode(' -#- ', $faktur[$index]);
```

jangan memusingkan pikiran anda sendiri. **INGAT** kode ini terletak didalam blok looping yaitu for. Sehingga perintah ini akan diulang sebanyak jumlah baris yang ada pada file faktur.txt.

Pada `$faktur[$index]` nilai `$index` pada saat eksekusi akan berubah, pada eksekusi pertama nilai `$index` adalah 0, pada eksekusi ke-2 nilai `$index` adalah 1 dan begitu seterusnya. Dengan demikian isi dari `$bagian` otomatis akan mengikuti `$index` yang aktif. PAHAM? Bagus.... Jika belum baca lagi tentang looping pada array.

```
include('../footer.php');
```

Karena bagian bawah tampilan website kita buat sama maka kita tidak perlu membuat file untuk bagian bawah lagi. Kita cukup panggil file footer.php yang ada pada parent direktori (direktori diatas admin) dengan menggunakan tanda `../` yang artinya pindah satu direktori ke atas.



Gambar 1.2.6: output file lihat\_faktur.php pada admin area

Berikutnya kita akan membuat file lihat\_kontak.php. Seperti namanya file ini akan menampilkan daftar kontak yang masuk ke administrator lewat form kontak kami. Buat file PHP baru lalu ketik kode berikut ini

#### Script dari file **lihat\_kontak.php**

---

```
<?php
include('header_admin.php');

$sumber = file('kontak.txt');
// hitung jumlah kontak yang ada
$jml_kontak = count($sumber);

if ($jml_kontak == 0)
{
 echo "Tidak ada faktur yang tercatat";
}

echo '<table border="0" cellpadding="4">
 <tr bgcolor="#ff6600">
 <th>Tanggal</th>
 <th>Nama</th>
 <th>Email</th>
 <th>Pesan</th>
 </tr>';

for ($index=0; $index<$jml_kontak; $index++)
{
 // pecah setiap bagian
 $bagian = explode(' -#- ', $sumber[$index]);
 // buat baris untuk setiap data kontak
 echo '<tr>';
 for ($jml_bagian=0; $jml_bagian<4; $jml_bagian++)
 {
 echo '<td>'.$bagian[$jml_bagian].'</td>';
 }
 echo '</tr>';
}
```

```

}

echo '</table>'
.'<p></p>';

include('..../footer.php');
?>

```



Gambar 1.26: Output file lihat\_kontak.php pada admin area

Jika pada suatu direktori tidak terdapat file index.htm, index.html atau index.php maka web server akan menampilkan daftar file yang ada pada direktori tersebut. Oleh karena itu untuk alasan keamanan selalu buatlah file index diatas meskipun isinya kosong.

\*\*\*\*\*

## Kalender

27	28	29	30	31	1	2	
3	4	5	6	7	8	9	
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

## Berikutnya

Pada bab berikutnya kita akan mempelajari bagaimana membuat fungsi, mengaplikasikan OOP dan teknik-teknik PHP lainnya.

# Bab 2

- Fungsi
- OOP (Object Oriented Programming)
- Teknik PHP Lainnya

Sebelum melanjutkan ikuti langkah-langkah berikut:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\
- Buat folder baru dan beri nama **Bab2**

\*\*\*\*\*

Seperti biasa karena ada pergantian hari maka ikuti langkah berikut:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab2
  - Buat folder baru berinama folder tersebut **hari-7**
- 

# F U N G S I

Dari pertama pembicaraan kita tentang PHP anda sering mendengar kata fungsi. Sebenarnya apa sih fungsi itu?. Secara garis besar fungsi merupakan kode yang memiliki modul, melakukan beberapa tugas, dan dapat mengembalikan suatu nilai. Kita sebelumnya sering memanggil fungsi-fungsi built-in PHP seperti `fopen()`, `strlen()` dan lain-lain.

Setelah membaca materi fungsi ini anda diharapkan mengerti tentang:

- Definisi Fungsi
- Membuat Fungsi
- Penamaan Fungsi
- Parameter
- Mengembalikan Nilai dari Fungsi
- Scope (Jangkauan)

## Memanggil Fungsi

Kode dibawah ini contoh sederhana pemanggilan fungsi yang tidak memiliki parameter.

```
nama_fungsi();
```

Beberapa fungsi dapat dipanggil dengan cara seperti diatas. Seperti fungsi `phpinfo()` yang merupakan fungsi untuk menampilkan versi PHP dan lingkungan servernya. Fungsi `phpinfo()` tidak memiliki parameter. Jadi untuk memanggilnya kita cukup mengetikkan

```
phpinfo();
```

Namun sebagian besar fungsi memiliki satu atau lebih parameter untuk melakukan tugasnya. Contoh pemanggilan fungsi yang memiliki parameter adalah sebagai berikut

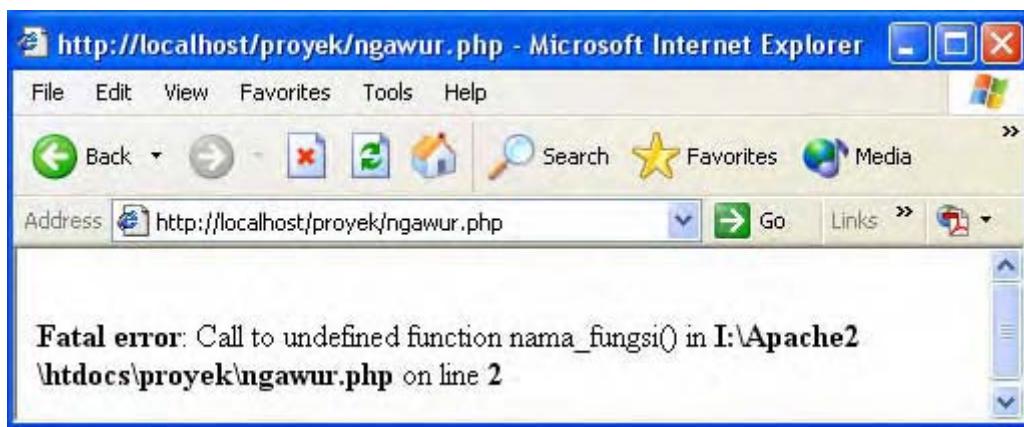
```
nama_fungsi('parameter');
```

Pada contoh diatas parameter yang digunakan adalah tipe string. Selain string parameter juga dapat berupa interger, double atau merupakan variabel bahkan array.

```
nama_fungsi(18); //parameter ini bertipe integer
nama_fungsi(25.765); //parameter ini bertipe double
nama_fungsi($variabel); //parameter ini merupakan variabel
```

## Fungsi Tidak Dikenal

Jika kita memanggil fungsi yang tidak terdefinisi maka akan muncul pesan error seperti gambar 2.0.



Gambar 2.0: fungsi yang tidak dikenal

Pesan error seperti diatas sangatlah membantu kita. Karena kita akan tahu jenis dari kesalahan, file mana yang error dan terletak di baris keberapa. Untuk kasus diatas ada dua hal yang perlu anda selidiki:

1. Apakah nama fungsi sudah tertulis dengan benar?
2. Apakah fungsi itu ada pada versi PHP yang anda gunakan?

Mengecek nama suatu fungsi tidaklah selalu mudah contohnya. Fungsi `addslashes()` menggabungkan dua kata sekaligus. Bandingkan dengan `strip_tags()` yang memisahkan kata dengan tanda underscore. Salah mengetikkan nama fungsi atau fungsi tersebut tidak ada akan memunculkan pesan error seperti gambar 2.0. tadi.

## Penamaan Fungsi

Perlu anda ketahui kalau nama fungsi **tidak** membedakan huruf kecil dan huruf besar. Jadi fungsi `nama_fungsi()` sama dengan `NAMA_FUNGSI()` atau `naMa_Fungsi()` dan sebagainya. Namun pada buku ini kita akan membuat fungsi yang semuanya menggunakan huruf kecil.

Jadi penamaan pada fungsi berbeda dengan variabel. Variabel membedakan huruf kecil dan huruf besar. `$nama` berbeda dengan `$NaMa`. Tapi kalau fungsi `nama()` sama dengan `NaMa()`. Sebaiknya nama fungsi yang anda buat mencerminkan tugas dari fungsi itu sendiri. Misalnya tugasnya mencetak pesan error maka nama fungsinya dapat anda buat `cetakerror()` atau `cetak_error()`.

Berikut ini contoh penamaan fungsi yang boleh(legal) dan tidak diperbolehkan (illegal). Nama-nama fungsi berikut adalah legal:

- `nama()`
- `_nama()`
- `nama3()`
- `nama_tiga()`

Sedangkan nama-nama fungsi berikut ini ilegal:

- `3nama()`
- `nama-tiga()`
- `fopen()`

`fopen` adalah legal **jika** fungsi ini belum ada. Namun karena sudah merupakan fungsi built-in PHP maka akan dianggap ilegal. Dan akan ditangkap oleh pihak kepolisian polda PHP Jaya. Memang pengedar???

## Mengapa Harus Membuat Fungsi Buatan Sendiri?

Fungsi-fungsi built-in yang sudah ada pada PHP sangatlah kompleks. Diantaranya mengijinkan anda berinteraksi dengan filesystem, database, server dan masih banyak lagi. Namun bagaimanapun juga suatu waktu anda ingin melakukan sesuatu yang tidak terpikirkan oleh pembuat program.

Dengan menulis fungsi sendiri anda dapat melakukan sesuatu sesuai keinginan anda. Kode anda mungkin gabungan dari fungsi built-in PHP maupun fungsi buatan anda sendiri. Jika anda menulis suatu blok kode dan blok kode itu sering anda gunakan pada tempat atau file lain. Anda dapat menulis blok kode itu kedalam suatu fungsi.

## Struktur Dasar Fungsi

Untuk membuat suatu fungsi selalu diawali dengan mendeklarasikan keyword **function** lalu diikuti nama fungsi, parameter jika ada, dan blok kode. Contoh sederhana pembuatan fungsi adalah sebagai berikut

```
function fungsiku()
{
 echo "Fungsiku dipanggil";
}
```

Pada kode diatas kita mendeklarasikan sebuah fungsi dengan nama **fungsiku**. Untuk memanggil fungsi ini kita cukup mengetikkan

```
fungsiku();
```

output yang ditampilkan pada browser adalah

Fungsiku dipanggil

Fungsi built-in pada PHP akan dapat diakses diseluruh script anda. Tetapi fungsi yang kita buat sendiri hanya tersedia pada file tempat fungsi itu dibuat. Untuk itu ada baiknya anda membuat sebuah file yang khusus berisi fungsi-fungsi yang sering anda gunakan. Untuk memanggilnya kita dapat menggunakan fungsi **include()** atau **require()**.

## Parameter Fungsi

Jika fungsi sederhana diatas tidak memiliki parameter. Maka sekarang kita akan coba membuat fungsi yang memiliki parameter. Lihatlah contoh berikut

```
function fungsi_2($param1, $param2)
{
 echo "Isi dari parameter 1 adalah ".$param1."
";
 echo "Isi dari parameter 2 adalah ".$param2."
";
}
```

Pada fungsi\_2 diatas kita memiliki dua parameter yaitu **\$param1** dan **\$param2**. Untuk lebih memperjelas pemahaman anda kita akan coba gunakan fungsi tersebut. Untuk menjalankan fungsi tersebut kita dapat menuliskan

```
fungsi_2("1. PHP 5.04", "2. MySQL 4.01");
```

Output yang akan ditampilkan oleh browser dari pemanggilan fungsi diatas adalah

Isi dari parameter 1 adalah 1. PHP 5.04  
Isi dari parameter 2 adalah 2. MySQL 4.01

Dari contoh diatas dapat anda simpulkan kalau variabel `$param1` dan `$param2`, Yang ada pada blok kode `fungsiku_2` akan diganti sesuai dengan input yang diberikan user ketika fungsi itu dipanggil. Pada contoh diatas `$param1` akan diganti dengan “**1. PHP 5.04**” sedangkan `$param2` akan diganti dengan “**2. MySQL 4.01**”.

Parameter juga ada yang bersifat opsional artinya boleh ditulis boleh tidak. Untuk membuat parameter yang bersifat opsional kita harus menentukan nilainya dulu. Untuk lebih jelas lihatlah contoh berikut

```
function peringatan($pesan, $warna = 'red')
{
 echo ''. $pesan. '';
}
```

Pada fungsi diatas parameter pertama **wajib** ditulis sedangkan parameter kedua oleh ditulis atau tidak. Contoh jika parameter kedua tidak ditulis adalah

```
peringatan('PERINGATAN, ANDA GAGAL!!!');
```

Output dari fungsi diatas adalah:

**PERINGATAN, ANDA GAGAL!!!**

Jika kita menulis parameter kedua maka kita harus menulis warna yang ingin ditampilkan. Contoh berikut akan mengganti warna asal yaitu merah dengan biru.

```
peringatan('PERINGATAN, ANDA GAGAL!!!', 'blue');
```

**PERINGATAN ANDA GAGAL!!!**

## Scope (Jangkauan)

Scope variabel mengontrol dimana variabel itu dapat dilihat dan digunakan. Bahasa pemrograman satu dengan lainnya mungkin memiliki peraturan yang berbeda dalam hal scope variabel. Pada PHP ada beberapa ketentuan dari scope variabel diantaranya:

- Variabel yang dideklarasikan didalam fungsi hanya dapat dilihat pada fungsi tersebut saja. Ini disebut *variabel local*.
- Variabel yang dideklarasikan diluar fungsi pada suatu script dapat dilihat diseluruh bagian script *tapi tidak* didalam fungsi. Ini disebut *variabel global*.
- Variabel yang dapat dilihat baik diluar ataupun didalam fungsi disebut *variabel superglobal*.
- Penggunaan `include()` atau `require()` tidak mempengaruhi scope. Jika statemen `include()` berada dalam fungsi maka yang berlaku variabel *lokal*. Jika statemen itu diluar fungsi yang berlaku adalah variabel *global*.
- Keyword `global` dapat digunakan untuk membuat variabel *lokal* yang ada didalam fungsi menjadi variabel *global*.

Contoh-contoh berikut akan menambah pemahaman anda tentang scope variabel pada fungsi.

```
function fungsiku()
{
echo "Isi \$var didalam fungsi: $var
";
$var = "Aku variabel lokal";
echo "Isi \$var didalam fungsi: $var
";
}

$var = "Aku variabel global";
```

```
fungsku();
echo "Isi \$var diluar fungsi: $var
";
```

Output dari kode diatas adalah sebagai berikut:

Isi \$var didalam fungsi:

Isi \$var didalam fungsi: Aku variabel **lokal**

Isi \$var diluar fungsi: Aku variabel **global**

Karena tanda **\$** merupakan karakter spesial agar tidak dicetak isinya maka kita gunakan escape karakter yaitu **\**. Ingat fungsi tidak akan dieksekusi jika tidak dipanggil. Jadi kode yang dieksekusi pertama kali yaitu `$var = "Aku variabel <b>global</b>"`; kemudian fungsi `fungsiku()` dipanggil. Blok kode yang ada didalam fungsiku dieksekusi. Karena pada baris pertama `$var` masih belum didefinisikan maka isinya kosong. Baru pada baris kedua `$var` didefinisikan. Sehingga pada baris ketiga `$var` sudah memiliki nilai.

`$var` yang berada pada fungsi merupakan variabel lokal. Yang isinya Aku variabel **lokal**. `$var` yang berada diluar fungsi tidak terpengaruh jadi tetap dicetak Aku variabel **global**. Jika anda ingin variabel `$var` bertipe global maka kita dapat menambahkan keyword **global** didepannya. Lihat contoh Kode berikut

```
function fungsiku()
{
global $var;
$var = "Aku variabel didalam";
echo "Isi \$var didalam fungsi: $var
";
}

$var = "Aku variabel diluar";
fungsku();
echo "Isi \$var diluar fungsi: $var
";
```

Output dari kode diatas adalah:

Isi \$var didalam fungsi: Aku variabel **didalam**

Isi \$var diluar fungsi: Aku variabel **didalam**

Perhatikan kalau isi dari `$var` yang diluar fungsi tertindih oleh `$var` didalam fungsi karena adanya keyword **global** `$var`.

## Menghentikan Eksekusi Fungsi

Untuk menghentikan eksekusi pada suatu fungsi dapat digunakan keyword **return**. Ketika pada suatu fungsi menemui keyword **return** maka eksekusi akan dihentikan. Pada fungsi berikut akan membandingkan dua angka. Jika pada saat eksekusi angka yang ingin dibandingkan tidak **diset** maka fungsi akan keluar.

```
function fungsiku($a, $b)
{
if (!isset($a) || !isset($b))
{
 echo "Fungsi ini memerlukan dua angka";
 return;
}

if ($a < $b)
 echo "$a lebih kecil dari $b
";
else if ($a == $b)
```

```

echo "$a sama dengan $b
";
else
 echo "$a lebih besar dari $b
";
}

$x = 8;
$y = 10;
fungsku($x, $y);
fungsku($y, $x);
fungsku($z, $y);

```

Output yang akan ditampilkan browser adalah

8 lebih kecil dari 10

10 lebih besar dari 8

Fungsi ini memerlukan dua angka

Pada kode diatas kita gunakan `isset()` untuk mengetes suatu variabel apakah nilainya sudah diset(ditentukan) sebelumnya. Karena kita ingin mengetes apakah jika variabel itu belum diset maka fungsi yang kita gunakan adalah `!isset()`. Yang artinya **belum/tidak** diset. Jadi pada kode diatas dapat kita terjemahkan "*jika a tidak diset atau jika b tidak diset*". Fungsi akan akan mengeksekusi blok kode if tersebut jika kondisinya true artinya ada variabel yang belum diset.

Pada output kode diatas tepatnya baris yang ke tiga. Fungsi keluar dari rutin karena variabel `$z` tidak diset sebelumnya. Jadi fungsi akan keluar.

## Mengembalikan Nilai dari Fungsi

Daripada memberikan perintah echo pada suatu fungsi. Akan lebih baik jika kita hanya mengembalikan nilainya saja. Dengan demikian kita dapat memodifikasi bagaimana cara menampilkan hasilnya. Dari fungsi sebelumnya kita dapat memodifikasi fungsi sebelumnya menjadi seperti berikut.

```

function fungsiku($a, $b)
{
if (!isset($a) || !isset($b))
{
 return false;
}

if ($a < $b)
 return "$a lebih kecil dari $b
";
else if ($a == $b)
 return "$a sama dengan $b
";
else
 return "$a lebih besar dari $b
";
}

$x = 8;
$y = 10;
echo fungsiku($x, $y);
echo fungsiku($y, $x);
echo fungsiku($z, $y);

```

Output yang akan ditampilkan browser adalah

8 lebih kecil dari 10

10 lebih besar dari 8

Karena `$z` tidak ada maka hasilnya tidak ditampilkan karena `false`. Fungsi yang tidak menampilkan hasil biasanya memiliki nilai `true` atau `false` untuk mengindikasikan fungsi tersebut sukses atau tidak.

## PENERAPAN TEORI FUNGSI

Seperti biasa bukalah program favorit kita PHP DESIGNER 2005. Ketik kode berikut ini dengan sersan. Loh kok.... maksudnya serius tapi santai.... Pada kode dibawah ini kita akan coba membuat fungsi-fungsi sederhana untuk menghasilkan tampilan HTML.

```
<?php

//ini untuk bagian keterangan halaman seperti judul dan CSS
function HTML_header($judul)
{
 $header = "<HTML>
 .<HEAD>
 .<TITLE>$judul</TITLE>
 .<STYLE>
 .body{font-family: verdana; font-size: 10px}"
 .</STYLE>
 .</HEAD>
 .<BODY>";

 return $header;
}

//ini untuk bagian bawah halaman sebagai penutup tag
function HTML_footer()
{
 $footer = "</BODY>
 .</HTML>";

 return $footer;
}

//fungsi sederhana yang menampilkan teks berjalan
function animation($input_teks)
{
 $anim = "<MARQUEE>$input_teks</MARQUEE>";
 return $anim;
}
?>
```

Simpan kode diatas dengan nama ***fungsi.php*** dan jangan lupa letakkan pada folder ***hari-7***. Nantinya file *fungsi.php* akan kita panggil menggunakan `include()`.

Sekarang buatlah kembali sebuah file PHP baru klik **File-New-PHP**, ketik kode berikut ini.

```
<?php

include('fungsi.php'); //panggil file fungsi.php

echo HTML_header('Aku Memanggil Fungsi');

$teks = "<h2>Aku memanggil fungsi dari file fungsi.php</h2>";
echo animation($teks);

echo HTML_footer();

?>
```

Simpan dengan nama **panggil\_fungsi.php** lalu jalankan pada browser Internet explorer anda di [http://localhost/webmaster/Bab2/hari-7/panggil\\_fungsi.php](http://localhost/webmaster/Bab2/hari-7/panggil_fungsi.php). Hasilnya akan seperti gambar 2.1



Gambar 2.1: output file panggil\_fungsi.php

### PENJELASAN SCRIPT

Pada file fungsi.php kita memiliki tiga fungsi yaitu `HTML_header()`, `HTML_footer()` dan `animation()`. Masing-masing fungsi tersebut akan mengembalikan nilai. Oleh karena itu untuk menampilkan nilainya kita gunakan perintah `echo()`.

`HTML_header($judul)` merupakan fungsi untuk mencetak header dari halaman. Seperti judul halaman dan style CSS. Fungsi ini memiliki satu parameter yaitu string untuk nama judul. Sedangkan untuk `HTML_footer()` tidak memiliki parameter karena hanya bertugas mencetak tag penutup untuk body dan html.

Fungsi `animation($input_teks)` bertugas untuk menampilkan animasi teks berjalan dengan menggunakan elemen HTML yaitu `marquee`. Sama dengan `HTML_header()` `animation()` juga mempunyai satu paramter yaitu teks yang ingin dianimasikan.

Agar fungsi-fungsi pada file fungsi.php dapat dikenali didalam file panggil\_fungsi.php. Kita gunakan statemen `include('fungsi.php')`. Untuk membuat halaman HTML yang benar tentu harus diawali dengan tag `<html>`.

Oleh karena itu kita panggil fungsi `HTML_header()` terlebih dahulu untuk mencetak header halaman. Pada kode diatas judul halaman dari file fungsi.php *Aku Memanggil Fungsi*.

Lalu pada body halaman kita akan mencetak teks yang bergerak menggunakan fungsi `animation()`. Setelah itu kita gunakan fungsi `HTML_footer()` untuk menutup tag body dan html. Modifikasi lah sesuka hati anda karena seperti kata pepatah "**Pengalaman adalah Guru yang Terbaik**".

## OOP (Object Oriented Programming)

Wah dari namanya saja serem begitu pasti sulit nih. Itu tergantung otak anda, jika otak anda menganggap ini mudah maka ini akan mudah. Tapi kalau anda sudah beranggapan sulit pasti sulit pula otak anda dalam memahaminya. Fokus yang harus anda pahami untuk materi ini diantaranya

- Konsep OOP
- Membuat Class
- Variabel `$this`
- Properti `private`, `public` dan `protected`
- Memanggil Class
- Inheritance(Pewarisan)

PHP 5 memiliki perubahan yang drastis menyangkut OOP. Dimana PHP versi ke-4 masih belum 100% OOP. Sekarang PHP sudah bisa dikatakan bahasa yang mendukung OOP sepenuhnya. Namun disini kita tidak akan membahas kelebihan OOP pada PHP 5. Melainkan kita akan memahami OOP dan menerapkannya pada website kita nantinya. Itulah tujuan utama kita!

## Konsep OOP

Pemrograman berorientasi objek merupakan konsep pemrograman yang menganggap sebuah objek memiliki properti dan metode. Didalam OOP dikenal apa yang dinamakan *class*. Class merupakan sebuah tempat bagi objek yang mendeskripsikan metode dan properti apa yang dimiliki objek ini.

Properti merupakan data dari objek yang berarti sebuah variabel milik dari objek tersebut. Metode merupakan suatu aksi yang dimiliki oleh objek tersebut juga bisa disebut fungsi yang didukung oleh objek itu.

Misalnya kita memiliki sebuah class *orang*. Untuk setiap orang diaplikasi tersebut kita dapat memisahkan suatu kejadian dari class ini yang menunjukkan informasi dari orang yang bersangkutan. Lihat ilustrasi dibawah ini.

class orang
metode:
beriNama(\$nama) panggilNama()
properti:
\$nama

Gambar 2.3: Ilustrasi class orang

Misalnya ada dua orang bernama Jono dan Joni, kita dapat memisahkan kedua aksi atau kejadian dengan memanggil metode *beriNama(\$nama)*. Dengan demikian kita dapat memanipulasi dua orang ini berdasarkan nama mereka yang ada pada variabel *\$nama*. Bingung? Tentu saja karena dokter contoh masih belum datang jadi tunggu saja.

## Membuat Class

Untuk membuat class kita dapat mendeklarasikannya dengan menggunakan keyword **class** dan diikuti nama class. Struktur dasar penulisan class adalah seperti berikut.

```
Class nama_class
{
...
//daftar properti dan metode
...
}
```

Seperti yang kita tahu class merupakan kumpulan dari fungsi atau metode dan properti atau variabel.

```
class nama_class
{
 private $myvar;

 function cetak_string($string);
 {
 $this->myvar = $string;
```

```

 echo $this->myvar;
 }
}

```

Jika pada PHP 3 dan 4 hanya mengenal penggunaan `var` sebagai deklarator variabel dalam suatu class. PHP 5 memiliki tiga deklarator yaitu `private`, `public` dan `protected`. Dari gambar 2.3 kita dapat membuat suatu class dengan nama orang yang memiliki dua metode dan satu properti. Maka kodennya akan seperti berikut:

```

class orang
{
 private $nama;

 function beriNama($strNama)
 {
 $this->nama = $strNama;
 }

 function panggilNama()
 {
 return $this->nama;
 }
}

```

Pada kode diatas kita membuat variabel `$nama` sebagai properti dari class orang dengan menggunakan keyword `private`. Kemudian kita membuat dua fungsi sebagai metode dari class orang yaitu `beriNama($strNama)` dan `panggilNama()`. Metode `beriNama()` akan menset properti dari `$nama` menjadi sesuai dengan input dari user. Sedangkan `panggilNama()` akan mengembalikan nilai dari metode `beriNama()`.

## Variabel `$this`

Saat eksekusi berlangsung pada metode dari suatu objek. Variabel `$this` secara otomatis didefinisikan sendiri oleh PHP. `$this` merujuk pada objek itu sendiri. Contohnya `$this->nama` (jangan gunakan tanda `$` sebelum nama dari properti) akan merujuk pada properti `nama`. Jadi pada kode `$this->nama = $strNama` diatas isi dari properti `nama` sama dengan isi dari parameter metode `beriNama($strNama)`.

## Properti `private`, `public` dan `protected`

Kebanyakan bahasa pemrograman yang mendukung OOP mempunyai tiga keyword untuk identitas akses yaitu `private`, `public` dan `protected`. Pada PHP 3 dan 4 pendeklarasian properti hanya dengan keyword `var`, yang artinya sama dengan `public` di PHP 5.

Pada PHP 5 kita mendeklarasikan properti dengan keyword `private`, `public` atau `protected`. Meskipun demikian, untuk menjaga kompatibilitas dengan versi lama PHP 5 tetap membolehkan penggunaan keyword `var`. Untuk lebih jelasnya lihatlah tabel 2.1.

**Tabel 2.1** Perbedaan properti `private`, `public` dan `protected`

Keyword	Keterangan
<code>private</code>	<code>private</code> hanya dapat diakses dari dalam metode objek saja dan tidak dapat diakses diluar objek. <code>private</code> tidak dapat diakses dari class turunan.
<code>public</code>	<code>public</code> dapat diakses baik dari dalam metode objek saja atau diluar objek. <code>public</code> juga dapat diakses pada class turunan.
<code>protected</code>	<code>protected</code> hampir sama dengan <code>private</code> hanya saja <code>protected</code> dapat diakses dari class turunan.

Hal yang sama juga berlaku untuk metode. Jika anda ingin mendeklarasikan metode yang bersifat private maka anda dapat menulis `private function()`.

## Memanggil Class

Sedikit berbeda dengan memanggil suatu fungsi. Jika kita ingin memanggil class maka kita harus membuat sebuah variabel. Dimana variabel itu akan menyimpan objek dari class yang dipanggil. Struktur dasar pemanggilan class adalah:

```
$a = new nama_class;
```

Sekarang kita mempunyai objek dengan nama `$a` yang merupakan representasi dari class `nama_class`. Setelah itu baru kita dapat menggunakan metode dan properti dari class yang dipanggil. Dengan cara yang sama kita dapat memanggil class orang.

```
$orang1 = new orang;
$orang1->beriNama('Jono');

$orang2 = new orang;
$orang2->beriNama('Joni');

echo $orang1->panggilNama() . "
";
echo $orang2->panggilNama() . "
";
```

Output yang ditampilkan pada browser dari kode diatas adalah:

```
Jono
Joni
```

Pada kode diatas kita membuat dua objek yaitu `$orang1` dan `$orang2`. Untuk memanggil metode dalam objek kita gunakan tanda `->`. `$orang->beriNama('Jono')` artinya pada objek `$orang1` kita memanggil metode beriNama dengan parameter string berisi Jono. Sekarang properti nama sudah memiliki nilai kita tinggal memanggil isi dari objek `$orang1` dengan perintah `$orang1->panggilNama()`.

## Inheritance (Pewarisan)

Pewarisan? Ini mau buat surat wasiat atau bagaimana?. Wah dipikiran anda kok uang saja. Maksud dari inheritance adalah pembuatan class baru berdasarkan class yang sudah ada. Class baru tersebut sering disebut dengan class turunan(child) sedangkan class asal disebut class induk(parent). Sebagai contoh kita akan mewariskan class orang ke class orangbaru.

Untuk membuat class baru yang diturunkan dari class parent kita gunakan keyword extends. Struktur dasar penulisan penurunan class adalah

```
class B extends A
{
...
}
```

Kode diatas berarti turunkan class A pada class B. Jadi metode dan properti dari class A dapat diakses dari objek yang menggunakan class B (jika tidak bersifat private atau protected). Class orang yang pernah kita buat akan kita turunkan ke class orangbaru. Dengan menambahkan metode baru yaitu `beriAlamat()` dan `cariAlamat()`.

Kode dari class orang:

```
class orang
{
```

```

private $nama;

function beriNama($strNama)
{
 $this->nama = $strNama;
}

function panggilNama()
{
 return $this->nama;
}
}

```

Kode dari class orangbaru yang merupakan turunan dari class orang:

```

class orangbaru extends orang
{
 private $alamat;

 function beriAlamat($strAlamat)
 {
 $this->alamat = $strAlamat;
 }

 function cariAlamat()
 {
 return $this->alamat;
 }
}

```

Dibawah ini contoh penggunaan class orang baru.

```

$orang1 = new orangbaru;
$orang1->beriNama('Jono');
$orang1->beriAlamat('Jl. Jalan Santai');

$orang2 = new orangbaru;
$orang2->beriNama('Joni');
$orang2->beriAlamat('Jl. Buntu');

echo $orang1->panggilNama()." - ".$orang1->cariAlamat()."
";
echo $orang2->panggilNama()." - ".$orang2->cariAlamat()."
";

```

Output yang ditampilkan pada browser adalah:

Jono - Jl. Jalan Santai  
 Joni - Jl. Buntu

Jadi \$orang1 dan \$orang2 yang merupakan objek dari class orangbaru. Dapat menggunakan metode beriNama() dan panggilNama() yang ada pada class orang. Ini dikarenakan class orang baru merupakan turunan dari class orang. Tetapi class orang tidak dapat menggunakan metode dan properti dari class orangbaru.

## PENERAPAN TEORI CLASS

Pada contoh kali ini kita akan coba membuat class yang berfungsi sebagai template atau skin suatu halaman. Jadi pada halaman yang ingin kita rubah tampilannya kita cukup memberi tanda khusus seperti tag {namatag}. Untuk lebih jelasnya jalankan PHP DESIGNER 2005 klik **File-New-PHP** untuk membuat file PHP baru. Ketik kode berikut ini

```
class skin
```

```

{
 private $tag = array();
 private $file_skin;
 private $halaman;

 function ganti_tag($namatag, $str_pengganti)
 {
 $this->tag[$namatag] = $str_pengganti;
 }

 function ganti_skin($namafile)
 {
 $this->file_skin = $namafile;
 }

 function ganti_tampilan()
 {
 $this->halaman = file($this->file_skin);
 $this->halaman = implode("", $this->halaman);
 foreach($this->tag as $str_dicari => $str_baru)
 {
 $this->halaman = ereg_replace($str_dicari, $str_baru, $this->halaman);
 }
 echo $this->halaman;
 }
}

```

Simpan dengan nama ***template.php***. Selanjutnya buatlah file PHP baru lalu ketik kode berikut:

```

<html>
<head>
<title>PENERAPAN OOP</title>
</head>
<body>
<center>{JUDUL}</center>
<p>{UTAMA}</p>
</body>
</html>

```

Simpan dengan nama ***file\_skin.php***. Jika anda menjalankan file ini yang muncul hanyalah tulisan {JUDUL} dan {UTAMA}. Kedua tag inilah yang akan kita ganti dengan string lain menggunakan objek dari class *skin*.

Dan yang terakhir buatlah file PHP baru kembali dan ketik listing kode berikut.

```

<?php
include('template.php');

$judul = "<h2>OOP untuk file template</h2>";
$utama = "<p>Tag yang ada pada file file_skin.php akan diubah sesuai dengan
 string yang diinginkan user. kemudian ditampilkan pada file
 utama.php</h2>";

$skin = new skin;
$skin->ganti_skin('file_skin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tampilan();

?>

```

Simpan dengan nama ***utama.php*** dan jangan lupa simpan ketiga file diatas pada fodler hari-7. Untuk mengetehui hasilnya arahkan browser anda ke alamat <http://localhost/webmaster/Bab2/hari-7/utama.php>. Output dari kode diatas akan seperti gambar 2.4.



Gambar 2.4: output file utama.php

### PENJELASAN SCRIPT

Pada file *template.php* kita membuat tiga properti dan salah satunya bertipe array yaitu properti **\$tag**. Mengapa? Karena kita tahu tag yang ada pada file yang kita gunakan sebagai template dalam kasus ini *file\_skin.php* tidak hanya terdiri dari satu tag saja. Tetapi beberapa tag, untuk itulah kita gunakan array. Jadi untuk setiap tag nantinya akan diganti dengan string tertentu.

Pada file *template.php* juga terdapat tiga metode diantaranya metode `ganti_tag()`, `ganti_skin()` dan `ganti_tampilan()`.

```
function ganti_tag($namatag, $str_pengganti)
{
 $this->tag[$namatag] = $str_pengganti;
}
```

Seperti yang anda lihat, metode `ganti_tag` memiliki dua parameter yaitu `$namatag` dan `$str_pengganti`. `$namatag` merupakan tag yang akan diganti, tag ini berada pada file yang memiliki tag yang akan diganti. Dan `$str_pengganti` digunakan untuk mengganti `$namatag`. Untuk dapat mengganti tag pada file yang diinginkan tentu harus kita menentukan tag dari file mana yang akan diganti isinya. Untuk menentukan file yang digunakan kita menggunakan metode `ganti_skin()`.

```
function ganti_skin($namafile)
{
 $this->file_skin = $namafile;
}
```

Metode `ganti_skin()` hanya memiliki satu properti yaitu `$namafile` yang merupakan string dari nama file yang ingin digunakan sebagai template. Jadi isi dari properti `file_skin` merupakan `$namafile`. Setelah menentukan nama file dan tag yang ingin diganti metode yang terakhir adalah tinggal mengganti dan mencetak ke layar. Metode yang melakukan tugas tersebut adalah `ganti_tampilan()`.

```
function ganti_tampilan()
{
 $this->halaman = file($this->file_skin);
 $this->halaman = implode("", $this->halaman);
 foreach($this->tag as $str_dicari => $str_baru)
 {
```

```
 $this->halaman = ereg_replace($str_dicari, $str_baru, $this->halaman);
}
echo $this->halaman;
}
```

Pada metode ini kita menentukan isi dari properti halaman. Isi dari properti halaman merupakan file yang telah kita tentukan dari metode ganti\_skin(). Langkah pertama adalah kita membuka file tersebut dengan pertintah file(\$this->file\_skin).

Sekarang isi dari properti halaman adalah array dari setiap baris file yang dibuka. Agar dapat dibaca secara utuh kita akan menggabung kembali array dari halaman. Untuk itu kita gunakan fungsi implode( " ", \$this->halaman). Artinya kita menggabungkan array halaman tanda diberi tanda pemisah. Sehingga akan tampil apa adanya sesuai yang dikehendaki.

Yang menarik adalah penggunaan looping foreach untuk mengganti setiap tag dengan string yang diinginkan. Ingat tanda => pada foreach berarti disebelah kiri tanda tersebut berarti index array sedangkan disebelah kanan merupakan nilai dari array. Pada blok looping ini kita menggunakan fungsi ereg\_replace() untuk mengganti tag yang dicari dengan string baru. Langkah terakhir adalah mencetak isi dari halaman menggunakan echo \$this->halaman.

---

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

Seperti biasa setiap kita menginjak hari yang baru kita akan membuat direktori baru.

- Masuklah pada direktori C:\Apache2\htdocs\webmaster\Bab2\
- Buat folder baru pada direktori tersebut
- Namai folder itu **hari-8**

## Teknik PHP Lainnya

Pada materi Teknik PHP Lainnya ini kita akan membahas teknik-teknik PHP lainnya. Diharapkan setelah membaca materi ini anda dapat menguasai tentang:

- Tanggal dan Waktu
- Session Pada PHP
- Menggunakan `$_GET`
- Upload File
- Direktori dan File

### Tanggal dan Waktu

Pada bab 1 kita sudah pernah menggunakan fungsi tanggal dan waktu. Yaitu dengan memanggil fungsi `date()`. Fungsi `date()` memiliki satu parameter yaitu bertipe string. Lihatlah contoh berikut

```
echo date('d-m-Y');
```

Output yang ditampilkan pada browser adalah 11-12-2005. Huruf d mengindikasikan tanggal sekarang. m merupakan bulan sekarang dan Y adalah tahun sekarang dalam 4 digit. Tanda “-“ kita gunakan sebagai pemisah antar tanggal, bulan dan tahun. Bandingkan dengan contoh berikut

```
echo date('l, d F Y');
```

Output dari kode diatas adalah

Sunday, 11 December 2005

Fungsi `date()` memiliki banyak sekali format tampilan. Untuk lebih lengkapnya lihatlah tabel 2.2 berikut.

**Tabel 2.2** Format kode untuk fungsi `date()`.

Kode	Keterangan
a	“am” atau “pm”
A	“AM” atau “PM”
d	Tanggal sekarang dalam format 2 digit yaitu “01” sampai “31”
D	Nama hari dalam format 3 huruf misalnya “Sun”, “Mon” dan seterusnya
F	Nama bulan dalam format penuh seperti “January”, “Februari” dan seterusnya
g	Jam dalam format 12 jam tanpa didahului angka nol, yaitu “1” sampai “12”
G	Jam dalam format 24 jam tanpa didahului angka nol, yaitu “0” sampai “23”
h	Jam dalam format 12 jam dengan didahului angka nol, yaitu “01” sampai “12”
H	Jam dalam format 24 jam dengan didahului angka nol, yaitu “00” sampai “23”
i	Menit dalam format yang didahului angka nol, yaitu “00” sampai “59”

I	Menampilkan "1" jika <i>Daylight Saving Time</i> dan "0" jika tidak
j	Tanggal sekarang dalam format yang tidak didahului nol, yaitu "1" sampai "31"
I	Nama hari dalam format utuh seperti "Sunday", "Monday" dan seterusnya
L	Mengembalikan "0" jika tahun sekarang bukan kabisan dan "1" jika sekarang merupakan tahun kabisat
m	Bulan dalam format yang didahului angka nol, yaitu "01" sampai "12"
M	Nama bulan dalam format 3 huruf, misalnya "Jan", "Feb" dan seterusnya
n	Bulan dalam format yang tidak didahului angka nol, yaitu "1" sampai "12"
O	Perbedaan waktu dengan GMT, misalnya "+0700" untuk (Indonesia)
r	Tanggal dalam format RFC 822, misalnya "Sun, 16 Oct 2005 20:10:22 +0700"
s	Detik dalam format yang didahului angka nol, yaitu "00" sampai "59"
S	Akhiran "th", "nd", dan "rd" pada tanggal
t	Jumlah hari dalam bulan sekarang, misalnya "28", "29", "30" atau "31"
T	Pengaturan timezone pada mesin, misalnya "MDT"
w	Kode hari dalam angka, misalnya "0" untuk Sunday, "1" untuk Monday dan seterusnya
y	Tahun sekarang dalam format 2 digit, misalnya "05"
Y	Tahun sekarang dalam format 4 digit, misalnya "2005"
z	Hari ke sekian dalam satu tahun, yaitu "1" sampai "365"

Wih banyak sekali, mana hafal? Buat apa dihafal kalau lupa tinggal lihat saja. Namun ada beberapa yang penting seperti d, m, Y. Karena kita tinggal di negeri kita Indonesia tercinta. Maka seyogyanya kita menggunakan nama hari dan bulan sesuai negara kita.

## PENERAPAN TEORI TANGGAL DAN WAKTU

Pada penerapan teori kali ini. Kita akan mencoba membuat fungsi untuk menampilkan **deadline** hari tertentu. Anda tentunya ingat ketika mengunjungi website rahasia-webmaster.com. Deadline pada rahasia-webmaster.com adalah dua hari setelah user mengunjunginya. Kita akan coba membuat fungsi untuk menampilkan deadline tersebut.

Untuk itu jalankan PHP DESIGNER 2005 buat file PHP baru dengan mengklik **File-New-PHP** ketik listing kode berikut ini

```
<?php

function show_deadline($pilihan)
{
// buat ketentuan hari, tanggal, bulan dan tahun sekarang
$tanggal = date('j'); // tanggal format 1 sampai 31
$hari = date('w'); // kode hari 0=minggu 1=senin dan seterusnya
$bulan = date('n') - 1; // dikurangi 1 agar index awal = 0
$tahun = date('Y'); // tahun format 4 digit
$jml_hari = date('t'); // jumlah hari dalam bulan sekarang

//buat variabel $nama_hari bertipe array
$nama_hari = array("Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jumat",
"Sabtu");

//buat variabel $nama_bulan bertipe array
$nama_bulan = array("Januari", "Februari", "Maret", "April", "Mei", "Juni",
"Juli", "Agustus", "September", "Oktober", "November", "Desember");

//tentukan jumlah penambahan dari tanggal sekarang
$tglplus = 2; // kita tetapkan deadline-nya 2 hari
$blnplus = 0;
$thnplus = 0;

$hari_ini = $nama_hari[$hari];
```

```

$bulan_ini = $nama_bulan[$bulan];
$hari_nanti = $nama_hari[$hari + $tglplus];

if ($hari_nanti == null && $hari_ini == "Jumat")
 $hari_nanti = "Minggu";
else if ($hari_nanti == null && $hari_ini == "Sabtu")
 $hari_nanti = "Senin";

$tgl_nanti = $tanggal + $tglplus;

if ($jml_hari == 31)
{
 if($tgl_nanti > 31)
 {
 $tgl_nanti = $tgl_nanti - 31;
 $blnplus = 1;
 }
}
else if ($jml_hari == 30)
{
 if ($tgl_nanti > 30)
 {
 $tgl_nanti = $tgl_nanti - 30;
 $blnplus = 1;
 }
}
else if ($jml_hari == 28)
{
 if ($tgl_nanti > 28)
 {
 $tgl_nanti = $tgl_nanti - 28;
 $blnplus = 1;
 }
}
else
{
 //tahun kabisat
 if ($tgl_nanti > 29){
 $tgl_nanti = $tgl_nanti - 29;
 $blnplus = 1;
 }
}
//buat variabel bln_nanti untuk menampilkan Bulan
$bln_nanti = $nama_bulan[$bulan + $blnplus];

// jika index bulan takterdefinisi maka index array bulan berarti 12 //maka kita
// anggap bulan Januari
if ($bln_nanti == null)
{
 $bln_nanti = "Januari";
 $thnplus = 1;
}
//buat variabel thn_nanti untuk menampilkan Tahun
$thn_nanti = $tahun + $thnplus;

// jika pilihan = 1 maka tanggal deadline yang ditampilkan
// jika pilihan = 0 maka tanggal hari ini yang ditampilkan
if ($pilihan == 1)
{
 $hari_deadline = $hari_nanti.', '. $tgl_nanti.' '. $bln_nanti.' '. $thn_nanti;
 return $hari_deadline;
}
else if ($pilihan == 0)

```

```

{
 $hari_biasa = $hari_ini . ' ' . $stanggal . ' ' . $bulan_ini . ' ' . $tahun;
 return $hari_biasa;
}
else
 exit('Nilai paramater pertama dari show_deadline() harus 0 atau 1');

}

//ini akan menampilkan tanggal hari ini
echo show_deadline(0)." - hari ini
";
//ini akan menampilkan tanggal deadline
echo show_deadline(1)." - deadline";

?>

```

Simpan dengan nama **tanggal.php** lalu jalankan pada browser. Output yang akan ditampilkan pada browser akan seperti berikut jika hari ini merupakan hari Minggu tanggal 16 Oktober 2005. Output di browser anda tergantung dari tanggal hari ini.

Minggu, 16 Oktober 2005 - hari ini  
 Senin, 18 Oktober 2005 - deadline

## PENJELASAN SCRIPT

Pada kode diatas kita membuat sebuah fungsi yaitu `show_deadline()`. Fungsi ini memiliki satu paramater. Yaitu `$pilihan` dimana 0 untuk menampilkan tanggal hari ini. Dan 1 untuk menampilkan deadline dua hari ke depan. Selanjutnya kita membuat variabel untuk hari, tanggal, bulan, tahun dan jumlah hari bulan sekarang.

Kesemuanya bertipe integer atau angka. Ini dimaksudkan agar kita mudah untuk mengolahnya dengan array. Anda dapat melihat kembali tabel 2.2 jika masih kurang jelas.

Karena standar hari komputer mengikuti orang-orang bule. Dimana biasanya hari pertama yang disebut adalah Monday atau Minggu. Maka mau tidak mau kita juga harus mengikutinya. Jadi kita akan membuat variabel `$nama_hari` yang bertipe array dengan tujuh elemen yaitu Minggu, Senin dan seterusnya. Jadi isi dari `$nama_hari[0]` adalah Minggu, `$nama_hari[1]` adalah Senin dan seterusnya hingga elemen ke-7 yaitu `$nama_hari[6]` yang berisi Sabtu. Hal yang sama berlaku untuk array `$nama_bulan`.

Hal berikutnya yang kita lakukan adalah menentukan penambahan hari, bulan dan tahun dari waktu sekarang. Variabel yang kita buat untuk mengindikasikan penambahan adalah `$tglplus`, `$blnplus`, dan `$thnplus`. Khusus untuk `$tglplus` penambahannya sudah kita tetapkan yaitu 2.

```

$hari_ini = $nama_hari[$hari];
$bulan_ini = $nama_bulan[$bulan];
$hari_nanti = $nama_hari[$hari + $tglplus];

```

Yang unik dari kode diatas adalah cara kita mendapatkan index array dari variabel `$nama_hari`. Seperti yang ditetapkan sebelumnya kalau isi dari `$hari` adalah `date('w')`. Jadi `$hari` akan mengembalikan nilai 0 sampai 6 sesuai dengan hari sekarang. Sehingga isi dari `$hari_ini` akan merujuk pada string nama hari sekarang. Begitu pula dengan `$bulan_ini`, hanya saja karena nilai yang dikembalikan `$nama_bulan` adalah 1 sampai 12. Kita harus menguranginya dengan angka 1 agar sesuai dengan index array yaitu 0 sampai 11. `$hari_nanti` akan menampilkan hari deadline dengan cara menambahkan index array dari `$nama_hari` dengan isi dari `$tglplus` yang sudah kita tetapkan yaitu 2.

```

if ($hari_nanti == null && $hari_ini == "Jumat")
 $hari_nanti = "Minggu";

```

```
else if ($hari_nanti == null && $hari_ini == "Sabtu")
 $hari_nanti = "Senin";
```

Kode diatas berguna untuk menghandel index array yang tidak terdefinisi(null). Contohnya jika hari ini Jumat maka index \$hari\_nanti adalah  $5 + 2 = 7$ . Karena index \$nama\_hari hanya sampai 6 maka \$hari\_nanti menjadi tidak terdefinisi. Untuk itu kita menentukan isi dari \$hari\_nanti menjadi dua hari kedepan yaitu Minggu. Begitu pula jika hari ini adalah sabtu maka haka index \$hari\_nanti  $6 + 2 = 8$ . Karena index 8 tidak terdefinisi maka kita tetapkan hari \$hari\_nanti menjadi dua hari ke depan yaitu Senin.

```
if ($jml_hari == 31)
{
 if($tgl_nanti > 31)
 {
 $tgl_nanti = $tgl_nanti - 31;
 $blnplus = 1;
 }
}
```

Untuk apa kita membuat blok kode if diatas. Tanpa kode blok if diatas sebenarnya fungsi ini sudah berjalan dengan normal. Tapi..., itu hanya untuk tanggal 1 awal bulan sampai 2 tanggal terakhir dari bulan sekarang misalnya 28 atau 29. Jika kita tidak menambahkan blok if diatas maka jika maksimal tanggal pada bulan sekarang 31 dan tanggal sekarang 30. Maka deadlineanya menjadi 32, 32? Mana ada tanggal 32 bung!!.. Untuk itu kita perlu mengetahui jumlah hari dalam bulan sekarang agar kita dapat menentukan pengurangnya.

Kita ambil contoh misalnya bulan sekarang Desember yang jumlah harinya 31. Maka blok if pada kode `if ($jml_hari == 31)` dijalankan. Jika tanggal deadline yang isinya pada \$tgl\_nanti lebih dari 31 maka kita kurangi dengan 31. Misalnya \$tgl\_nanti isinya 32 maka kita kurangi 31 sekarang \$tgl\_nanti menjadi 1. Berarti ada pergantian bulan maka \$blnplus kita set menjadi 1. Agar \$bln\_nanti menjadi satu bulan kedepan.

```
$bln_nanti = $nama_bulan[$bulan + $blnplus];
```

Isi dari \$bln\_nanti merupakan isi dari array \$nama\_bulan. \$bulan akan mengembalikan nilai dari 0 hingga 11 setelah kita kurangi satu pada awal kode diatas. Isi dari \$blnplus awalnya adalah 0. Namun akan menjadi 1 jika isi dari \$tgl\_nanti lebih dari jumlah tanggal sekarang.

```
if ($bln_nanti == null)
{
 $bln_nanti = "Januari";
 $thnplus = 1;
}

$thn_nanti = $tahun + $thnplus;
```

Pada blok if diatas jika nilai \$bln\_nanti tak terdefinisi(null) maka index arraynya adalah 12. Karena pada \$nama\_bulan index 12 tidak ada maka kita tetapkan isi \$bln\_nanti menjadi Januari. Otomatis tahun akan berubah jadi kita tetapkan isi dari \$thnplus menjadi 1 agar tahun bertambah 1.

```
if ($pilihan == 1)
{
 $hari_deadline = $hari_nanti.'.'.$tgl_nanti.'.'.$bln_nanti.'.'.$thn_nanti;
 return $hari_deadline;
}
```

Selanjutnya kita tinggal menentukan nilai balik dari fungsi `show_deadline()`. Jika \$pilihan yang diisikan user pada parameter adalah 1. Maka yang ditampilkan adalah deadline. Untuk itu kita buat variabel \$hari\_deadline dengan isi yang merupakan variabel-variabel yang isinya

adalah 2 hari kedepan. Kita gunakan return \$hari\_deadline untuk mengembalikan nilainya. Sehingga dapat ditampilkan ketika dipanggil dengan echo. Jika \$pilihan sama dengan 0 maka yang ditampilkan adalah tanggal hari ini. Jika tidak 0 atau 1 maka keluar dari program.

## Session Pada PHP

Pada dasarnya session digunakan untuk melacak user selama ia berada pada suatu website. Biasanya session diimplementasikan pada sistem login, shopping chart dan lain-lain. Untuk session kita dapat menggunakan **cookie** atau **PHP Session**. Kita akan membahasnya satu persatu. Kita mulai dari cookie terlebih dahulu.

### Cookie

Cookie adalah sekumpulan informasi kecil yang diletakkan pada komputer klien. Cookie biasanya menyimpan informasi pada suatu sesi tertentu. Anda dapat menggunakan fungsi **setcookie()** untuk membuat cookie. Struktur dasar dari penulisan **setcookie()** adalah

```
setcookie(string nama cookie, [string nilai], [angka kadaluwarsa], [string path],
[string domain], [string secure]);
```

setcookie memiliki enam parameter namun yang wajib diisi hanya satu yaitu nama dari cookie. Contoh berikut mengilustrasikan cookie dengan nama *mycookie* dan isinya adalah *Halo Cookie*.

```
setcookie('mycookie', 'Halo Cookie');
```

Untuk mengakses isi dari cookie *mycookie* kita menggunakan **\$\_COOKIE['mycookie']**.

```
echo "Isi dari cookie mycookie adalah ". $_COOKIE['mycookie'];
```

Untuk menghapus cookie yang telah anda buat kita harus membuat cookie dengan nama yang sama tapi waktunya kita set pada waktu lampau. Misalnya

```
setcookie('mycookie', '', time() - 3600);
```

Arti dari **time() - 3600** arinya adalah 3600 detik yang lalu dalam jam berarti 1 jam yang lalu. Karena lampau maka otomatis *mycookie* dihapus.

### PENERAPAN TEORI COOKIE

Jalankan PHP DESIGNER 2005 lalu buat file PHP baru dengan mengklik **File-New-PHP**. Lalu ketik kode berikut

```
<?php

ob_start();
echo "COOKIE
";
setcookie('mycookie', 'Halo Cookie');
echo "Isi dari mycookie adalah ".$_COOKIE['mycookie']."
";

echo 'Ke halaman cookie2.php';

?>
```

Simpan dengan nama ***cookie1.php***. Jalankan pada browser hasilnya akan seperti berikut.

### COOKIE

Isi dari *mycookie* adalah Halo Cookie

[Ke halaman cookie2.php](#)

Selanjutnya buat kembali file PHP baru dan ketik kode berikut.

```
<?php

echo "COOKIE dari halaman cookie1
";
echo "Isi dari mycookie adalah ". $_COOKIE['mycookie']. "
";
echo 'Kembali ke cookie1.php';

?>
```

Simpan dengan nama ***cookie2.php*** lalu jalankan pada browser. Hasilnya akan seperti berikut.

### **COOKIE dari halaman cookie1**

Isi dari mycookie adalah Halo Cookie

[Kembali ke cookie1.php](#)

### **PENJELASAN SCRIPT**

Fungsi ob\_start() pada baris awal mengaktifkan output buffering yang dibutuhkan untuk menciptakan cookie. Tanpa adanya ob\_start() maka akan muncul pesan seperti berikut:

**Warning:** Cannot modify header information - headers already sent by (output started at I:\Apache2\htdocs\webmaster\Bab2\hari-8\cookie1.php:4) in  
I:\Apache2\htdocs\webmaster\Bab2\hari-8\cookie1.php on line 5

Dapat anda lihat cookie mycookie tidak hanya dapat dilihat dari halaman cookie1.php tetapi juga cookie2.php.

## **PHP Session**

Selain menggunakan cookie kita juga dapat menggunakan PHP Session untuk menyimpan informasi pada sesi tertentu. Langkah-langkah yang perlu kita lakukan untuk berinteraksi dengan PHP Session antara lain:

- Memulai Session dengan perintah `session_start()`
- Mendaftarkan variabel session dengan `$_SESSION['nama_session']`
- Menggunakan variabel session tersebut
- Menghapus dan menghancurkan variabel session dengan `unset()` dan `session_destroy()`

Fungsi `session_start()` akan mengecek apakah sudah ada session yang aktif. Jika belum, maka akan dibuat session baru. Tapi jika sudah ada maka `session_start()` akan membaca session tersebut agar dapat kita gunakan.

### **PENERAPAN TEORI SESSION**

Pada penerapan teori session ini kita akan membuat empat file yaitu ***login.php***, ***proses\_login.php***, ***member.php*** dan ***logout.php***. Keempat file itu kita gunakan untuk membangun sebuah member area sederhana. Buat file PHP baru klik **File-New-PHP** untuk membuat file PHP baru. Ketik kode berikut.

```
<html>
<head>
<title>Sistem login Sederhana</title>
</head>
<body>

<h2>Silahkan Login</h2>
<form action="proses_login.php" method="POST">
Username: <input type="text" name="username" size="16">

Password: <input type="password" name="password" size="16">

<input type="submit" value="Login">
```

```

</form>
Member Area

</body></html>

```

Simpan dengan nama ***login.php***. Jalankan pada browser hasilnya akan seperti gambar 2.5. Selanjutnya buat file PHP baru lagi dan ketik kode berikut.

```

<?php

session_start();
$username = $_POST['username'];
$password = $_POST['password'];

if ($username == 'test' && $password == 'pas123')
{
 $_SESSION['login'] = $username;
 echo 'Anda login sebagai '.$_SESSION['login'].'
';
 echo 'Silahkan klik Member Area untuk melihat member area.' ;
}
else
{
 echo 'Login gagal, username atau password salah
';
 echo 'Klik Login untuk mencoba lagi.' ;
}

?>

```

Simpan kode diatas dengan nama ***proses\_login.php***.



Gambar 2.5: output file login.php

Selanjutnya kita akan membuat file ***member.php***. Ketik kode berikut ini untuk membuatnya

```

<?php

session_start();
$sesiku = $_SESSION['login'];

if (isset($sesiku))
{
 echo "<h2>Member Area</h2>";
 echo $sesiku." Selamat datang di member area
";
 echo 'Logout';
}
else

```

```
{
 echo 'Anda belum login Silahkan Login dulu.' ;
}

?>
```

Dan yang terakhir ketik kode berikut ini

```
<?php

session_start();
$sesiku = $_SESSION['login'];

if(isset($sesiku))
{
 unset($sesiku);
 session_destroy();
 echo 'Anda telah logout. Silahkan Login kembali.' ;
}
else
{
 echo 'Anda belum login. Silahkan Login dulu.' ;
}

?>
```

Simpan dengan nama ***logout.php***. Coba jalankan file ***login.php*** lalu isikan username dengan *test* dan password dengan *pas123*. Lalu klik tombol login, maka anda akan dibawa ke member area.

## PENJELASAN SCRIPT

Kita mulai pembahasan kita dari file yang pertama yaitu *login.php*. Tugas utama dari file ini hanya untuk menampilkan form login. Tidak ada yang spesial didalamnya. Disini kita membuat dua input yaitu text dan password. Masing-masing namanya adalah *username* dan *password*. Jadi kita harus ingat dua nama ini jika kita akan memprosesnya.

Berikutnya adalah file *proses\_login.php*. Pada file ini kita akan memproses form login dari file *login.php*. Karena pada file ini kita akan menggunakan session maka pada awal script kita panggil fungsi *session\_start()*.

Selanjutnya kita buat variabel *\$username* dan *\$password* yang isinya masing-masing dari *login.php* yang kita dapat lewat metode POST. Setelah mendapatkan isi dari *\$username* dan *\$password* kita akan mencocokkannya dengan menggunakan statemen if.

```
if ($username == 'test' && $password == 'pas123')
```

Pada kode diatas jika isi dari *\$username* sama dengan *test* dan isi dari *\$password* sama dengan *pas123* Maka login berhasil, karena user berhasil login maka kita buat session dengan nama *login* yang berisikan username dari user yang berhasil login. Caranya dengan menggunakan *\$\_SESSION['login'] = \$username*.

Jika username tidak sama dengan *test* dan password tidak sama dengan *pas123*. Maka login gagal, untuk itu kita tidak akan membuatkan sesi untuk user ini. Blok kode pada ***else*** akan dijalankan, yang memberitahu bahwa proses login gagal. Kita tidak perlu membuatkan sesi untuk user ini karena dia gagal melakukan login.

Pada file *member.php* kita membuat variabel *\$sesiku* yang isinya adalah session login. Kita gunakan fungsi *isset()* untuk mengecek apakah *\$sesiku* memiliki nilai. Jika ya maka user ini user yang berhasil login. Maka tampilkan member area. Jika *\$sesiku* tidak memiliki nilai maka

user tidak berhasil login maka jangan tampilkan member area. Untuk itu blok kode pada else yang akan dieksekusi.

File yang terakhir yaitu *logout.php* bertugas menghapus session login dengan perintah `unset()`. Lalu menghancurkan session menggunakan `session_destroy()`.

---

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

Seperti biasa setiap pergantian hari baru kita akan buat folder baru. Untuk itu ikuti langkah-langkah berikut:

- Masuk ke direktori C:\Apache2\htdocs\webmaster\Bab2
- Buat folder baru pada direktori tersebut
- Beri nama folder itu hari-9

## Teknik PHP Lainnya (2)

### Menggunakan \$\_GET

Penggunaan `$_GET` yang paling sering digunakan adalah jika kita mengambil data dari URL. Misal `fileku.php?ref=test`. Pada URL tersebut data yang akan kita ambil adalah `test` dan berada pada variabel `ref`. Untuk membuat lebih dari satu variabel pada URL kita dapat menggunakan tanda & untuk memisahkan antar data.

Contoh `fileku.php?proses=hapus&user=test`. Tidak usah berlama-lama kita langsung saja mempraktekkan teori ini.

#### PENERAPAN TEORI \$\_GET

Jalankan PHP DESIGNER 2005. Klik menu **File-New-PHP** lalu ketik kode berikut.

```
<?php

// buat variabel untuk menampung data dari URL
$action = $_GET['action'];
// tentukan aksi yang dilakukan jika action kosong
if ($action == '')
 $action = 'login';

if ($action == 'login')
{
 echo '<h2>Form Login</h2>
 . '<form>
 . 'Username: <input type="text">

 . 'Password: <input type="password">

 . '<input type="submit" value="Login">

 . '</form>
 . '<p>Berita Terbaru</p>';
}
else if ($action == 'news')
{
 echo '<h2>Berita Terbaru</h2>
 . 'Penjualan Komputer Menurun

 . 'Melemahnya rupiah terhadap Dollar AS berdapat pada...
 . '<p>Login</p>';
}

?>
```

Simpan dengan nama **pilihan.php**. Jalankan pada browser maka yang muncul adalah halaman form login. Jika anda mengklik link **Berita Terbaru** maka yang muncul adalah halaman berita. Menarik bukan? satu file tapi memiliki dua tampilan.

#### PENJELASAN SCRIPT

Pada kode diatas pertama yang harus kita lakukan adalah menambahkan isi dari data dari URL. Dimana isi dari data yang akan kita ambil terletak pada *action*. Oleh karena itu kita gunakan `$_GET['action']`. Lalu isi dari `$_GET['action']` kita tampung pada variabel `$action`.

Karena pada saat memanggil file ini yang URL yang dipanggil hanyalah <http://localhost/webmaster/Bab2/hari-9/pilihan.php>.

Maka yang muncul dibrowser hanyalah tampilan putih bersih tidak ada apa-apa. Untuk itu kita gunakan statemen `if ($action == '')`. Jadi jika isi dari action kosong maka yang dijadikan nilai mula-mula adalah login.

Selanjutnya kita tentukan jika action berisi *login* maka blok kode pada statemen `if ($action == 'login')` dijalankan. Pada tampilan login kita memberi link berita terbaru yang linknya di <http://localhost/webmaster/Bab2/hari-9/pilihan.php?action=news>.

Karena isi dari action adalah *news* maka blok pada `else if ($action == 'news')` akan dijalankan.

## Upload File

Salah satu fungsi yang cukup berguna pada PHP adalah mendukung penggunaan HTTP Upload. Upload merupakan kebalikan dari download. Jika download adalah proses pengiriman file dari server ke komputer kita. Upload kebalikannya yaitu pengiriman file dari komputer kita ke server.

Untuk menampilkan form upload tidak jauh berbeda dengan form biasa. Hanya saja didalam tag `<form>` kita menuliskan attribut `enctype = "multipart/form-data"`. Untuk menampung file dari form upload kita gunakan variabel array superglobal yaitu `$_FILES`.

### PENERAPAN TEORI UPLOAD FILE

Jalankan PHP DESIGNER 2005 klik menu **File-New-HTML/XHTML** untuk membuat file HTML baru.

```
<html>
<head>
<title>File Upload</title>
</head>
<body>
<h2>File Upload...</h2>
<form enctype="multipart/form-data" action="upload.php" method="post">
 <input type="hidden" name="MAX_FILE_SIZE" value="1000000">
 Upload file ini: <input type="file" name="userfile" size="50">
 <input type="submit" value="Upload File...">
</form>
</body>
</html>
```

Simpan kode HTML diatas dengan nama ***upload.html***. Output dari kode diatas kan terlihat seperti gambar 2.6.



Gambar 2.6: output file upload.html

Selanjutnya kita akan membuat file PHP baru untuk memproses form upload. Untuk itu klik menu **File-New-PHP** pad PHP DESIGNER 2005. Lalu ketik kode berikut.

```

<html>
<head>
<title>Meng-upload...</title>
</head>
<body>
<h2>Mengupload File...</h2>

<?php

// $userfile merupakan tempat file di server
$userfile = $_FILES['userfile']['tmp_name'];
// $nama_file merupakan nama file asli
$nama_file = $_FILES['userfile']['name'];
// $ukuran_file merupakan ukurang file dalam bytes
$ukuran_file = $_FILES['userfile']['size'];
// $tipe_file merupakan tipe dari file seperti text/plain
$tipe_file = $_FILES['userfile']['type'];
// $file_error jika ada error terjadi
$file_error = $_FILES['userfile']['error'];

// jika ada error tangani error tersebut
if ($file_error > 0)
{
 echo 'Error: ';
 // tentukan error dengan switch
 switch ($file_error)
 {
 case 1: echo 'Ukuran file melebihi upload_max_filesize di php.ini'; break;
 case 2: echo 'Ukuran file melebihi max_file_size'; break;
 case 3: echo 'Hanya sebagian file yang terupload'; break;
 case 4: echo 'Tidak ada file yang diupload'; break;
 }
 exit; // keluar dari rutin script
}

// cek tipe dari file jika bukan text biasa keluar
if ($tipe_file != 'text/plain')
 exit('Error: Tipe file bukan text biasa');

// tentukan letak file yaitu di folder temp
// cek dulu jika sudah ada temp maka tidak perlu membuat
@ $baca_dir = opendir('temp/');
if (!$baca_dir)
{
 mkdir('temp/', 0777);
 $letak_file = 'temp/' . $nama_file;
}
else

```

```

$letak_file = 'temp/' . $nama_file;

// tutup direktori
closedir($baca_dir);

// cek apakah file sudah terupload
if (is_uploaded_file($userfile))
{
 if (!move_uploaded_file($userfile, $letak_file))
 exit('Error: Tidak dapat mengupload file ke direktori tersebut');
 else
 {
 // Jika tidak ada kesalahan maka tampilkan pesan berhasil
 echo 'File berhasil di upload

';
 echo 'Lihat folder temp/.';
 }
}

?>
</body>
</html>

```

Simpan dengan nama ***upload.php***.

### PENJELASAN SCRIPT

Pada file *upload.html* kita membuat input dengan type *file*. Nama dari data ini adalah ***userfile***. Kita juga membuat input dengan tipe *hidden*. Input tipe ini tidak akan nampak dilayar. Pada input *hidden* ini kita membuat data dengan nama *MAX\_FILE\_SIZE* yang isinya 1000000. Data ini yang digunakan untuk menentukan ukurang maksimal file yang dapat diupload dalam bytes. Jadi dalam kb berarti 1000 kb dan dalam MB berarti 1 MB.

Nah, ini bagian yang menarik. Kebanyakan script yang ada pada file *upload.php* adalah untuk menangani suatu kesalahan atau error. Untuk itu kita harus menentukan tipe dari error tersebut. `$_FILES['userfile']['tmp_name']` dalam hal ini *tmp\_name* merupakan string yang digunakan script untuk menghandel informasi dari data yang diupload. Daftar lengkap dapat anda lihat pada tabel 2.3

**Tabel 2.3** Daftar string penghandel informasi

Kunci	Keterangan
tmp_name	Nama sementara di sistem server. PHP akan menghapusnya setelah request selesai.
name	Nama file asli pada sistem user yang ia upload ke server.
size	Ukuran file yang diupload dalam bytes.
type	Tipe dari MIME file yang diupload. Untuk text biasa MIME-nya text/plain untuk file JPEG MIME-nya image/jpeg.
error	Kode error saat proses upload. Lihat tabel 2.4 untuk lebih lengkapnya.

**Tabel 2.4** Daftar kode error dan konstantanya

Kode	Konstanta	Keterangan
0	UPLOAD_ERR_OK	File berhasil diupload dan tidak ada error.
1	UPLOAD_ERR_INI_SIZE	Ukuran file melebihi nilai yang ditetapkan di konfigurasi php.ini.
2	UPLOAD_ERR_FORM_SIZE	Ukuran file melebihi yang ditentukan pada MAX_FILE_SIZE
3	UPLOAD_ERR_PARTIAL	File berhasil diupload namun hanya sebagian saja.
4	UPLOAD_ERR_NO_FILE	Tidak ada file yang diupload karena user tidak memilih file.

Jadi variabel `$file_error` isinya adalah untuk mengembalikan nilai error saat proses upload berlangsung. Karena jika nilai error = 0 maka tidak ada error. Kita gunakan statemen `if ($file_error > 0)`. Sebagai alternatif kita gunakan `switch` bukan `if`. Masa `if` terus kan bosen?!. Namun jika anda lebih suka menggunakan statemen `if` silakan rubah kode diatas.

`switch ($file_error)` artinya yang akan kita handel adalah nilai dari `$file_error`. Kita gunakan statemen `case` untuk menghandel setiap nilai dari error mulai dari 1 hingga 4. Jangan lupa akhir dari suatu statemen `case` harus diberi `break`.

```
@$baca_dir = opendir('temp/');
```

Maksud dari tanda `@` adalah jika nanti ada kesalahan kita dapat menghandel error tersebut. Kita tahu untuk penggunaan pertama kali folder temp belum ada. Fungsi `opendir('temp/')` maksudnya adalah membuka direktori temp. Karena pada kali pertama temp belum ada maka kita akan menangani error ini. Untuk itu kita gunakan statemen `if(!$baca_dir)` untuk menangani error tersebut.

Jika ada error artinya tidak dapat membuka temp/ karena memang belum ada. Maka buat direktori temp/ menggunakan fungsi `mkdir('temp/', 0777)`. Parameter pertama merupakan nama direktori yang akan dibuat. Sedangkan 0777 pada parameter kedua adalah permission dari direktori tersebut yang artinya dapat ditulis.

Jika tidak ada error pada `opendir()` berarti temp/ sudah ada. Jadi kita tidak perlu membuatnya dengan `mkdir`. Kita cukup menentukan letak file yang akan diupload yaitu pada variabel `$letak_file`. Dimana isinya adalah `'temp/' . $nama_file`. Langkah yang terakhir adalah mengecek apakah file upload kita sudah sampai tujuan dengan aman. Kayak bepergian saja...!.

```
if (is_uploaded_file($userfile))
```

Statemen if diatas mengecek apakah file sudah terupload. Lalu kita akan memindahkan file itu sesuai dengan keinginan kita. Dengan menggunakan fungsi `move_uploaded_($userfile, $letak_file)`. Jadi tanda `!` sebelum fungsi tersebut menandakan jika ada suatu kesalahan maka exit akan dijalankan. Selain itu berarti berhasil dan blok kode `else` yang dijalankan.

## Direktori dan File

Pada bahasan kali ini kita akan mengupas bagaimana membuat, membaca dan menghapus direktori. Hal yang sama juga akan kita bahas mengenai file. Untuk yang pertama kita akan membahas direktori terlebih dahulu.

### Direktori

Setelah user berhasil mengupload file kita juga dapat menggunakan PHP untuk membaca file yang telah diupload. PHP memiliki beberapa fungsi untuk memanipulasi direktori.

### Membaca Direktori

Untuk membaca suatu direktori kita dapat gunakan fungsi `readdir()`. Namun kita harus membuka direktori tersebut terlebih dulu dengan perintah `opendir()`. Jika sudah selesai kita dapat menutup direktori tersebut dengan menggunakan perintah `closedir()`. Untuk lebih memperjelas kita akan mempraktekkannya saja.

#### PENERAPAN TEORI MEMBACA DIREKTORI

Jalankan PHP DESIGNER 2005 buat file PHP baru dengan mengklik **File-New-PHP**. Ialu ketik kode berikut ini

```
<?php
```

```

echo '<h2>Membaca isi dari Direktori</h2>';
$mydir = 'temp/';
$dir = opendir($mydir);

echo "Direktori upload adalah di $mydir

 . Daftar file di direktori
<hr>
";

while ($file = readdir($dir))
{
 echo "$file -- "Lihat Isi
";
}

echo '
<hr>
';
closedir($dir);

?>

```

Simpan dengan nama **bacadir.php**. Output dari kode diatas akan terlihat seperti gambar 2.7



Gambar 2.7 output file bacadir.php

**Catatan:** Hasil yang ditampilkan pada komputer anda mungkin berbeda tergantung dari isi dari folder temp dan file yang anda upload.

### PENJELASAN SCRIPT

Pertama kita membuat variabel `$mydir` yang isinya string nama direktori upload kita yaitu `temp`. Setelah itu kita membuat variabel `$dir` yang berisi fungsi untuk membuka direktori `temp` yaitu `opendir('$mydir')`. Untuk daftar dari file yang ada pada folder `temp`. Kita gunakan looping while seperti berikut

```
while ($file = readdir($dir))
```

Pada looping diatas kita membuat variabel `$file` yang isinya adalah fungsi untuk membaca isi dari direktori yaitu `readdir()`. Karena dalam konteks looping maka perintah `$file = readdir($dir)` akan diulang sebanyak file yang ada pada direktori tersebut. Nilai yang akan dikembalikan dari variabel `$file` adalah nama file. Jadi pada blok kode while kita menampilkan nama file dengan cara

```
echo "$file -- "Lihat Isi
";
```

Karena isi kita menggunakan tanda petik " " sebagai string. Maka jika ingin menampilkan tanda string dalam tampilan kita gunakan escaping character yaitu \". Dan yang terakhir kita tutup direktori dengan menggunakan perintah closedir(\$dir).

## File

Setelah kita dapat masuk dan melihat isi direktori. Kita juga dapat memanipulasi file didalamnya. Seperti membuat, mengganti nama dan menghapus file. Kita akan bahas ketiganya sesaat lagi.

### Membuat, meng-copy, me-rename dan menghapus file

Untuk yang pertama yaitu membuat file kita dapat menggunakan fungsi touch(). Fungsi ini memiliki tiga parameter namun yang wajib diisi hanya satu. Struktur dasar penggunaan fungsi touch() adalah

```
touch(string nama_file, [integer waktu_pembuatan], [integer waktu_akses])
```

Jika nama file yang akan anda buat sudah ada. Maka waktu pembuatan akan diubah sesuai dengan waktu anda membuat file tersebut. Atau jika anda ingin menuliskan waktu pembuatan yang baru maka waktu itulah yang digunakan.

Untuk meng-copy file kita dapat menggunakan fungsi copy(). Fungsi ini memiliki dua parameter yaitu file asal dan file tujuan. Lihat contoh berikut

```
$file_asal = 'upload.html';
$file_tujuan = 'temp/upload.html';
copy($file_asal, $file_tujuan);
```

Contoh diatas mengilustrasikan kita akan meng-copy file *upload.html* ke direktori *temp*. Namun dengan asumsi script yang kita jalankan untuk kode diatas letaknya sama dengan file asal yaitu *upload.html*.

Untuk mengubah nama file PHP menyediakan fungsi rename. Sama dengan copy() fungsi ini juga membutuhkan dua parameter yaitu nama file lama dan nama file baru. Contohnya

```
rename('gambarku.jpg', 'gambar_dia.jpg');
```

karena PHP tidak menyediakan fungsi untuk memindahkan file (move). Anda dapat menggunakan rename untuk keperluan tersebut. Fungsi file yang terakhir adalah menghapus file. Untuk keperluan ini kita dapat menggunakan fungsi unlink(). Fungsi ini hanya membutuhkan satu paramater yaitu file yang akan dihapus.

```
unlink('password.txt');
```

---

## Kalender

27		28	29	30	31	1	2
3	✓	4	✓	5	✓	6	✓
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

# Proyek Aplikasi Bab 2 – Java Online News

Wah..., kemampuan anda sekarang sudah banyak berkembang rupanya. Maka dari itu proyek kita juga akan mengikuti kemampuan anda. Pada proyek ini kita akan coba membuat surat kabar online sederhana. Surat kabar ini kita beri nama Java Online News. Proyek pada bab 2 ini merupakan proyek terakhir kita menggunakan media file. Pada bab-bab selanjutnya kita akan menggunakan database(MySQL) sebagai media penyimpanan.

Apa yang kita dapat pada bab 2 akan kita terapkan disini. Fitur-fitur yang terdapat pada Java Online News antara lain:

- Berita Utama
- Fasilitas Iklan
- Pembatasan iklan
- Admin Login
- Penghapusan berita
- Penambahan berita
- Menggunakan Skin/template

Ikuti langkah-langkah berikut sebelum kita memulai proyek bab 2 ini:

- Buat folder baru pada direktori C:\Apache2\htdocs\webmaster\Bab2\
- Namai folder baru tersebut **hari-10**
- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab2\hari-10\
- Buat folder baru, namai folder tersebut **Proyek-Bab2**
- Masuk pada direktori C:\Apache\htdocs\webmaster\Bab2\hari-10\Proyek-Bab2\
- Buat beberapa folder baru, yaitu folder **admin**, **berita** dan **template**

Daftar file yang kita buat untuk membangun website Java Online News ditunjukkan oleh tabel 2.5.

**Tabel 2.5** Daftar file proyek Java Online News

<b>Direktori Proyek-Bab2\template</b>	
<b>Nama File</b>	<b>Fungsi</b>
myskin.php	Untuk menyimpan format tampilan website
template.php	Untuk menyimpan class skin
var_tampilan.php	Untuk menyimpan variabel-variabel tampilan
index.php	Untuk mengembalikan ke halaman utama

<b>Direktori Proyek-Bab2\admin</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.php	Untuk menampilkan form login dan menu utama
berita_utama.php	Untuk mengedit berita utama(menambah dan menghapus)
berita_lainnya.php	Untuk mengedit berita lainnya
iklan.php	Untuk menghapus iklan
fungsi.php	Untuk menyimpan kumpulan fungsi

<b>Direktori Proyek-Bab2\berita</b>	

Nama File	Fungsi
berita_utama.txt	Untuk menyimpan berita utama
berita_lainnya.txt	Untuk menyimpan berita lainnya
iklan.txt	Untuk menyimpan iklan
index.php	Untuk mengembalikan ke halaman utama

Direktori Proyek-Bab2	
Nama File	Fungsi
index.php	Untuk menampilkan halaman utama
pasangiklan.php	Untuk menampilkan form pasang iklan

**Tabel 2.6** Daftar format penyimpanan file

Nama File	Format
berita_utama.txt	tanggal berita#-#Judul Berita#-#Isi berita
berita_lainnya.txt	Nomor Berita#-#Link Berita#-#Judul Berita
iklan.txt	Isi iklan#-#link Iklan

OK, File pertama yang akan kita buat adalah file *fungsi.php*. File ini nantinya akan menyimpan fungsi-fungsi seperti session untuk administrator, login dan mengecek session. Buat file baru pada PHP DESIGNER 2005 dengan mengklik **File-New-PHP** lalu ketik kode berikut

Script dari file *fungsi.php*

```
<?php

//fungsi untuk login
function login($name, $pass)
{
 if ($name == 'admin' && $pass == 'pas123')
 return true; // jika benar maka true
 else
 return false; // jika salah maka false
}

// fungsi untuk logout
function logout()
{
 if (isset($_SESSION['login']))
 {
 unset($_SESSION['login']);
 session_destroy();
 return true;
 }
 else
 {
 return false;
 }
}

// fungsi untuk mengecek session
function cek_session()
{
 if (isset($_SESSION['login']))
 return true; // session login terisi
 else
 return false; // session login kosong
}
?>
```

Simpan pada direktori **Proyek-Bab2\admin\** dengan nama ***fungsi.php***.

### PENJELASAN SCRIPT

Pada fungsi yang pertama yaitu `login()`. Kita membutuhkan dua parameter untuk username dan untuk password. Jika parameter yang diisi user untuk username sama dengan `admin` dan password sama dengan `pas123` maka kembalikan nilai `true` selain itu kembalikan nilai `false`.

Fungsi yang kedua yaitu `logout()`. Pada fungsi ini kita mengecek kondisi dari `$_SESSION['login']` jika sudah terisi maka hapus dan hancurkan session dengan `session_destroy()`. Namun jika `$_SESSION['login']` belum terisi maka kembalikan nilai `false`

Fungsi yang terakhir adalah fungsi `cek_session()`. Fungsi ini berguna untuk memblok user yang belum login agar tidak bisa mengakses file-file administrator. Fungsi ini sangat sederhana. Pada fungsi ini kita hanya mengecek apakah session login telah terisi jika ya kembalikan nilai `true` selain itu `false`.

Kali ini file yang akan kita buat adalah `index.php` untuk administrator. File ini terletak pada direktori admin. File ini merupakan file multifungsi karena menyediakan interface. Diantaranya untuk login, menu utama dan logout. Untuk itu kita membuat variabel pada URL dengan nama `proses`. Contoh jika isi proses sama dengan `logout` maka yang dieksekusi adalah blok `logout..` Semuanya yang menghandel adalah file ini. Untuk membuatnya klik **File-New-PHP** pada PHP DESIGNER 2005 lalu ketik kode dibawah ini

#### Script dari file ***index.php***

```
<?php
session_start();

// panggil file fungsi.php
require('fungsi.php');

// daftarkan variabel dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'cek_login';

switch ($proses)
{
 // blok kode untuk proses = cek_login
 case 'cek_login': // titik dua bukan titik koma
 // cek apakah sesion login terisi
 if (!cek_session())
 {
 echo "Please, Login First" .
 '<form action="index.php?proses=proses_login" method="post">' .
 'username: <input type="text" name="username">
' .
 'password: <input type="password" name="password">
' .
 '<input type="submit" value="Login">' .
 '</form>' .
 '<p>Back to Java Online News</p>';
 }
 else
 {
 echo "<h2>Welcome To Admin Area of Java Online News</h2>" .
 "<p>Silahkan klik link dibawah ini untuk melakukan operasi yang " .
 ".
" .
 "Edit berita utama
" .
 "Edit berita lainnya
" ;
 }
}
```

```

 . "Edit iklan
"

 . "Logout" ;

 }

break; // akhir dari proses cek_login

// blok kode untuk proses = proses_login

case 'proses_login':

 $username = $_POST['username'];

 $password = $_POST['password'];

 // panggil fungsi login

 if (!login($username, $password))

 echo "username atau password salah. Login Lagi" ;

 else

 {

 $_SESSION['login'] = $username;

 echo "Anda berhasil login. Admin Menu" ;

 }

break; // akhir dari proses proses_login

case 'logout':

 if (!logout())

 echo "Tidak bisa logout. Login dulu" ;

else

 echo "Anda telah logout dari sistem. Login" ;

 break; // akhir dari proses logout

}

?


```

Simpan pada direktori **admin** dengan nama **index.php**. Untuk mengaksesnya anda dapat mengarahkan browser anda ke alamat <http://localhost/webmaster/Bab2/hari-10/Proyek-Bab2/admin/>.

### PENJELASAN SCRIPT

Karena pada file ini kita menggunakan session, maka pada awal script kita panggil fungsi `session_start()`. Selanjutnya kita memanggil file `fungsi.php` dengan `require()`. Lalu kita buat variabel `$proses` yang isinya adalah data dari proses yang ada pada URL misalnya `index.php?proses=xxx`. Jika proses kosong atau pada URL tidak ada maka tetapkan isi dari `$proses` sama dengan `cek_login`.

Untuk menangani setiap kondisi dari `$proses` kita gunakan statemen switch bukan if. Karena if terlalu banyak kita gunakan dia kan capek?!?. Untuk menghandel setiap nilai dari `$proses` kita gunakan statemen `case nama_proses:`. Akhir dari suatu statemen case harus ditandai dengan statemen `break`.

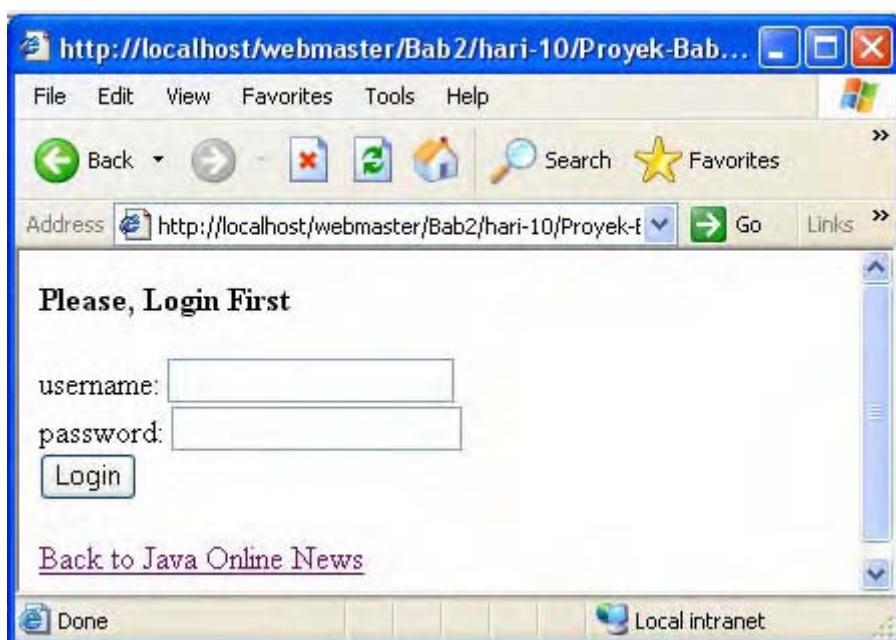
Pada kasus yang pertama yaitu `cek_login`. Kita menggunakan fungsi `cek_session()` untuk mengecek apakah admin sudah login atau tidak. `if (!cek_session())`. Ingat tanda `!` artinya jika ada kesalahan atau fungsi mengembalikan nilai false. Jadi jika nilai yang dihasilkan false maka munculkan form login. Selain itu munculkan menu utama admin area. Lihat gambar 2.8 dan 2.9.

Pada kasus yang kedua yaitu `proses_login`. Kita akan memproses data dari form login. Untuk itu kita buat variabel `$username` dan `$password` dari yang isinya masing-masing adalah data `username` dan `password` dari form login. Kita gunakan fungsi `if (!login($username, $password))` untuk mengecek nilai yang dikembalikan oleh fungsi. Pada kode diatas jika nilainya false maka pesan yang ditampilkan adalah `username atau password salah`. Sebaliknya jika true maka menu utama admin area akan ditampilkan.

Kasus yang terakhir adalah *logout*. Sama seperti sebelum-sebelumnya kita mengecek dulu apakah ada user yang telah login. Pada fungsi `!login()` maka jika belum ada yang login tampilkan Tidak bisa logout. Selain itu hancurkan session login dan pesan tampilkan user telah logout dari sistem.



Gambar 2.8: user yang telah login.



Gambar 2.9: user yang belum login.

File berikutnya yang akan kita buat adalah ***berita\_utama.php***. Seperti file sebelumnya yaitu index.php file ini juga memiliki beberapa fungsi yaitu untuk menampilkan judul dari berita utama, menghapus berita dan menambah berita. Oleh karena itu pada URL kita membuat data yaitu *proses*. Buka PHP DESIGNER 2005 klik menu **File-New-PHP** untuk membuat file PHP baru. Ketik kode berikut ini

Script dari file ***berita\_utama.php***

---

```
<?php
// mulai session
session_start();
require('fungsi.php');
```

```

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'lihat';

// jika belum login exit
if (!cek_session())
 exit("Anda belum login. Login dulu.");

switch ($proses)
{
 case 'lihat':
 echo '<h2>Daftar berita utama</h2><hr>';
 $berita_utama = file('../berita/berita_utama.txt');
 $jml_utama = count($berita_utama);

 // looping untuk menampilkan daftar berita
 for ($i=0; $i<$jml_utama; $i++)
 {
 $bagian = explode("#-#", $berita_utama[$i]);
 echo '<form action="berita_utama.php?proses=hapus" method="post">
 .<table border="0">
 .<tr>
 .<td>Tanggal</td>
 .<td>' . $bagian[0] . '</td></tr>
 .<tr>
 .<td>Judul</td>
 .<td>' . $bagian[1] . '</td></tr>
 .<tr>
 .<td><input type="submit" value="HAPUS"></td></tr>
 .</table>
 .<input type="hidden" name="judul" value="' . $bagian[1] . '">
 .</form>';
 }
 echo 'Tambah berita '
 . 'Halaman utama';
 break; // akhir dari proses lihat

 case 'hapus':
 $judul = $_POST['judul'];
 $berita_hapus = file('../berita/berita_utama.txt');
 $file_baru = fopen('../berita/berita_utama.txt', 'w');

 foreach ($berita_hapus as $data)
 {
 $data = str_replace("\n", "", $data);
 $bagian = explode("#-#", $data);
 // tulis kembali berita yang judulnya bukan judul yang dihapus
 if ($bagian[1] != $judul)
 fwrite($file_baru, $data . "\n");
 }
 fclose($file_baru);
 echo 'Berita telah terhapus '
 . 'Lihat';
 break; // akhir dari proses hapus

 case 'tambah':
 echo '<h2>Tambah berita utama</h2>
 .<form action="berita_utama.php?proses=proses_tambah" method="POST">
 .Tanggal:
<input type="text" name="tgl">

 .Judul:
<input type="text" name="judul" size="50">

 .Isi:
<textarea name="isi" rows="8" cols="75"></textarea>

 .<input type="submit" value="TAMBAH">
 .</form>
 .Lihat Berita '

```

```

 . 'Halaman utama';
break; // akhir dari proses tambah

case 'proses_tambah':
 $tgl = $_POST['tgl'];
 $judul = $_POST['judul'];
 $isi = $_POST['isi'];
 // jika ada field yang kosong maka keluar
 if (empty($tgl) || empty($judul) || empty($isi))
 exit('Error: ada field yang masih kosong');
 // tulis data ke file berita_utama.txt
 $isi = str_replace("\n", "
", $isi);
 $isi = str_replace("\r", "", $isi); // menghilangkan karakter kosong
 @$file_utama = fopen('../berita/berita_utama.txt', 'a');
 if (!$file_utama)
 echo "Tidak dapat membuka file";
 else
 {
 $berita = $tgl."#-#".$judul."#-#".$isi."\n";
 fwrite($file_utama, $berita);
 fclose($file_utama);
 }

 echo 'Berita berhasil ditambah. '
 .'Lihat';
 break; // akhir dari proses proses_tambah
}

?>

```

Simpan dengan nama ***berita\_utama.php*** letakkan pada direktori **Proyek-Bab2\admin**. Coba jalankan lalu tambahkan beberapa berita baru.

### PENJELASAN SCRIPT

Tujuan dari proses *lihat* adalah untuk menampilkan judul berita utama yang ada. Proses ini adalah proses yang dijalankan apabila isi proses kosong atau tidak ada data proses pada URL *berita\_utama.php*. Pada setiap judul berita kita berikan tombol submit untuk mempost berita mana yang akan dihapus. Lihat gambar 2.10 untuk memperjelas gambaran anda tentang proses *lihat*.

Pada kasus yang kedua yaitu proses *hapus* kita menerapkan metode yang cukup unik. Pertama kita membuat variabel *\$judul* yang isinya adalah judul berita utama yang dihapus. Lalu kita membuka file *berita\_utama.txt* dan menjadikannya array ke variabel *\$berita\_hapus* menggunakan fungsi *file()*. Kita buka kembali file tersebut tapi menggunakan metode yang berbeda yaitu *fopen*. Dan hasilnya kita masukkan pada variabel *\$file\_baru*.

Lalu kita gunakan looping *foreach* untuk untuk mengidentifikasi setiap elemen array. Kita masukkan setiap elemen array *\$berita\_hapus* ke variabel *\$data*. Jadi *\$data* merupakan setiap baris dari file *berita\_utama.txt* yang tidak lain adalah berita utama. Agar pada saat menulis ulang tidak terjadi baris baru maka kita hilangkan dulu karakter baris (*\n*) dengan menggunakan *str\_replace()*.

Selanjutnya kita akan memecah setiap bagian dari berita utama untuk mendapatkan judul berita. Sehingga nantinya kita bisa melakukan perbandingan judul berita yang dihapus dengan yang tidak. Seperti biasa kita gunakan fungsi *explode()* untuk memecah setiap bagian lalu kita masukkan hasilnya pada variabel *\$bagian*. Maka yang merupakan array judul adalah *\$bagian[1]*.

Logika kerja yang kita gunakan adalah kita akan menulis kembali berita yang judulnya bukan merupakan judul berita yang ingin dihapus. Untuk itu kita gunakan statemen *if (\$bagian[1]*

**!= \$judul).** Lalu kita tulis kembali setiap berita dengan menggunakan **fwrite()** dan menambahkan karakter baris baru (\n) pada setiap berita.



Gambar 2.10: output file berita\_utama.php?proses=lihat

Pada kasus proses *tambah* tidak ada yang istimewa. Tujuan proses ini hanyalah untuk menampilkan form untuk menambah berita. Disini disediakan textbox untuk tanggal dan judul sedangkan untuk isi berita disediakan textarea. Lihat gambar 2.11.

Pada kasus yang terakhir yaitu *proses\_tambah* mengambil data dari form tambah yang telah di-submit sebelumnya. Seperti biasa kita membuka dulu file yang akan kita gunakan untuk menyimpan data. Dalam hal ini kita membuka file *berita\_utama.txt* menggunakan **fopen()**. Mode file yang kita gunakan adalah **a** (append). Jadi data yang kita tulis otomatis akan berada pada baris terakhir karena ditambah.

Seperti yang pernah disebutkan sebelum kita memulai proyek ini. Bahwa format yang kita gunakan adalah **tanggal#judul#isi berita**. Maka kita harus gunakan format tersebut agar valid dengan apa yang ada pada file *var\_tampilan.php*. Langkah terakhir adalah kita menulis file *berita\_utama.txt* lalu memberitahuhan pada user kalau proses penambahan berita telah selesai.

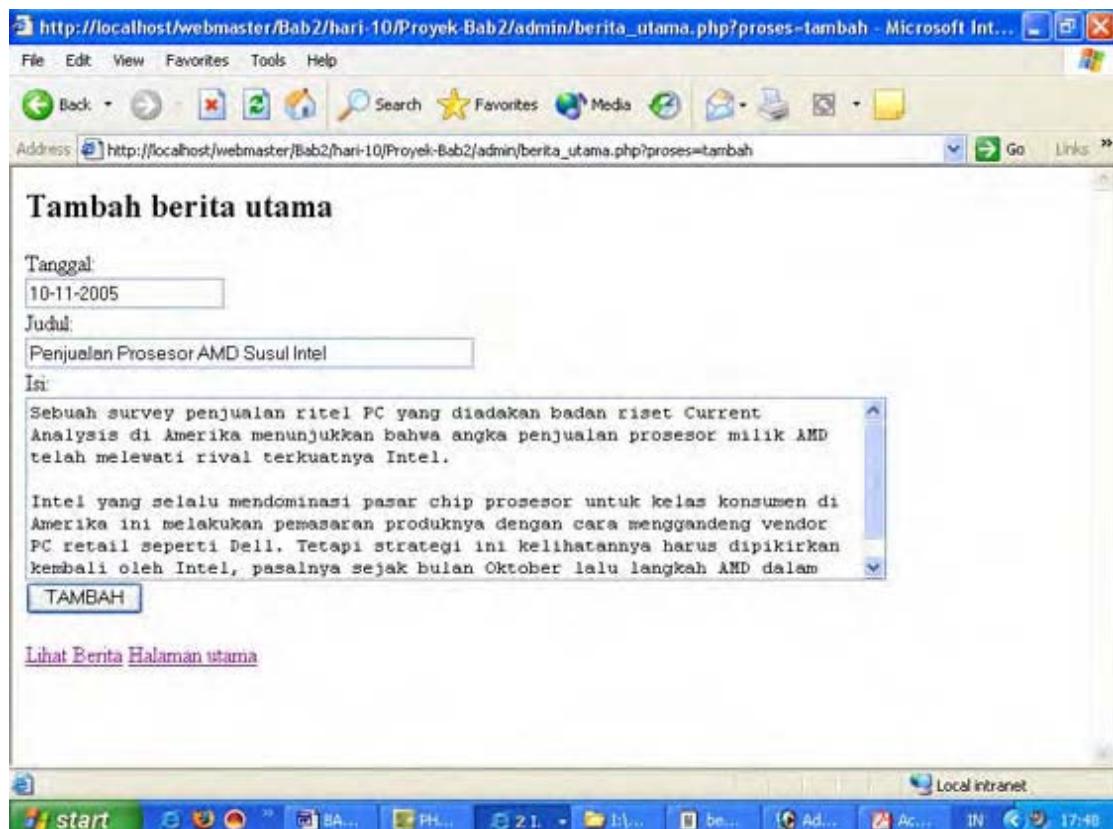
Pada kasus yang terakhir yaitu *proses\_tambah* mengambil data dari form tambah yang telah di-submit sebelumnya. Seperti biasa kita membuka dulu file yang akan kita gunakan untuk menyimpan data. Dalam hal ini kita membuka file *berita\_utama.txt* menggunakan **fopen()**. Mode file yang kita gunakan adalah **a** (append). Jadi data yang kita tulis otomatis akan berada pada baris terakhir karena ditambah.

Seperti yang pernah disebutkan sebelum kita memulai proyek ini. Bahwa format yang kita gunakan adalah **tanggal#-#judul#-#isi berita**. Maka kita harus gunakan format tersebut agar valid dengan apa yang ada pada file *var\_tampilan.php*. Langkah terakhir adalah kita menulis file *berita\_utama.txt* lalu memberitahuhan pada user kalau proses penambahan berita telah selesai.

Pada kasus yang terakhir yaitu *proses\_tambah* mengambil data dari form tambah yang telah di-submit sebelumnya. Seperti biasa kita membuka dulu file yang akan kita gunakan untuk

menyimpan data. Dalam hal ini kita membuka file *berita\_utama.txt* menggunakan `fopen()`. Mode file yang kita gunakan adalah `a` (append). Jadi data yang kita tulis otomatis akan berada pada baris terakhir karena ditambah.

Seperti yang pernah disebutkan sebelum kita memulai proyek ini. Bahwa format yang kita gunakan adalah `tanggal#-#judul#-#isi` berita. Maka kita harus gunakan format tersebut agar valid dengan apa yang ada pada file *var\_tampilan.php*. Langkah terakhir adalah kita menulis file *berita\_utama.txt* lalu memberitahukan pada user kalau proses penambahan berita telah selesai.



Gambar 2.11: output file berita\_utama.php?proses=tambah

Pada kasus yang terakhir yaitu *proses\_tambah* mengambil data dari form tambah yang telah di-submit sebelumnya. Seperti biasa kita membuka dulu file yang akan kita gunakan untuk menyimpan data. Dalam hal ini kita membuka file *berita\_utama.txt* menggunakan `fopen()`. Mode file yang kita gunakan adalah `a` (append). Jadi data yang kita tulis otomatis akan berada pada baris terakhir karena ditambah.

Seperti yang pernah disebutkan sebelum kita memulai proyek ini. Bahwa format yang kita gunakan adalah `tanggal#-#judul#-#isi` berita. Maka kita harus gunakan format tersebut agar valid dengan apa yang ada pada file *var\_tampilan.php*. Langkah terakhir adalah kita menulis file *berita\_utama.txt* lalu memberitahukan pada user kalau proses penambahan berita telah selesai.

Pada kasus yang terakhir yaitu *proses\_tambah* mengambil data dari form tambah yang telah di-submit sebelumnya. Seperti biasa kita membuka dulu file yang akan kita gunakan untuk menyimpan data. Dalam hal ini kita membuka file *berita\_utama.txt* menggunakan `fopen()`. Mode file yang kita gunakan adalah `a` (append). Jadi data yang kita tulis otomatis akan berada pada baris terakhir karena ditambah.

Seperti yang pernah disebutkan sebelum kita memulai proyek ini. Bahwa format yang kita gunakan adalah `tanggal#-#judul#-#isi` berita. Maka kita harus gunakan format tersebut agar valid dengan apa yang ada pada file *var\_tampilan.php*. Langkah terakhir adalah kita

menulis file *berita\_utama.txt* lalu memberitahukan pada user kalau proses penambahan berita telah selesai.

Pada kasus yang terakhir yaitu *proses\_tambah* mengambil data dari form tambah yang telah di-submit sebelumnya. Seperti biasa kita membuka dulu file yang akan kita gunakan untuk menyimpan data. Dalam hal ini kita membuka file *berita\_utama.txt* menggunakan *fopen()*. Mode file yang kita gunakan adalah *a* (append). Jadi data yang kita tulis otomatis akan berada pada baris terakhir karena ditambah.

Seperti yang pernah disebutkan sebelum kita memulai proyek ini. Bahwa format yang kita gunakan adalah *tanggal#-#judul#-#isi* berita. Maka kita harus gunakan format tersebut agar valid dengan apa yang ada pada file *var\_tampilan.php*. Langkah terakhir adalah kita menulis file *berita\_utama.txt* lalu memberitahukan pada user kalau proses penambahan berita telah selesai.

Berikutnya kita akan membuat file untuk menambah dan menghapus berita lainnya. Yang menyimpan data berita lainnya adalah file *berita\_lainnya.txt*. File *berita\_lainnya.txt* kita letakkan pada direktori **berita**. Buka PHP DESIGNER 2005 buat file baru lalu ketik kode berikut ini.

Script dari file ***berita\_lainnya.php***.

```
<?php
// mulai session
session_start();
require('fungsi.php');
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'lihat';

// jika belum login exit
if (!cek_session())
 exit('Anda belum login. Login dulu.');

switch($proses)
{
 case 'lihat';
 echo '<h2>Daftar link berita lainnya</h2>';
 . '';
 $file_lainnya = file('../berita/berita_lainnya.txt');
 $jml_lainnya = count($file_lainnya);
 for ($i=0; $i<$jml_lainnya; $i++)
 {
 $bagian = explode("#-", $file_lainnya[$i]);
 echo '<form action="berita_lainnya.php?proses=hapus" method="post">';
 . ''. $bagian[1]. ' --- ';
 . '<input type="submit" value="HAPUS">';
 . '<input type="hidden" name="id" value="'. $bagian[0]. '">';
 . '</form>';
 }
 echo '';
 . 'Tambah berita';
 . 'Halaman utama';
 break; // akhir dari proses lihat

 case 'hapus':
 $id = $_POST['id'];
 $berita_hapus = file('../berita/berita_lainnya.txt');
 $file_baru = fopen('../berita/berita_lainnya.txt', 'w+');

 foreach ($berita_hapus as $data)
 {
```

```

$data = str_replace("\n", "", $data);
$bagian = explode("#-#", $data);
echo $bagian[0];
// tulis kembali berita yang nomornya bukan nomor yang dihapus
if ($bagian[0] != $id)
 fwrite($file_baru, $data."\n");
}
fclose($file_baru);
echo 'Berita berhasil dihapus. '
 .'Lihat';
break; // akhir dari proses hapus

case 'tambah':
echo '<h2>Tambah Daftar Link Berita</h2><hr>'
 . '<form action="berita_lainnya.php?proses=proses_tambah" method="POST">'
 . 'Nama berita:
<input type="text" name="nama_berita" size="50">
'
 . '<input type="submit" value="TAMBAH">
'
 . '</form>'
 . 'Daftar berita'
 . 'Halaman utama';
break; //akhir dari proses tambah

case 'proses_tambah':
$nama_berita = $_POST['nama_berita'];
if (empty($nama_berita))
 exit('Anda belum mengisi nama_berita');

// buka file berita_lainnya.txt
@$file_lainnya = fopen('../berita/berita_lainnya.txt', 'a');
$file_cek = file('../berita/berita_lainnya.txt');
// pindah ke elemen array paling akhir dengan end()
$id_akhir = end($file_cek);
$bagian = explode("#-#", $id_akhir);
// buat $id yang berisi nomor id array terakhir + 1
$id = $bagian[0] + 1;

if (!$file_lainnya)
 exit('Tidak dapat membuka file');
else
{
 $isi_file = $id.'#-#'.$nama_berita.''."\n";
 fwrite($file_lainnya, $isi_file);
 fclose($file_lainnya);
 echo 'Berita berhasil ditambah. '
 .'Lihat';
}
}
?>

```

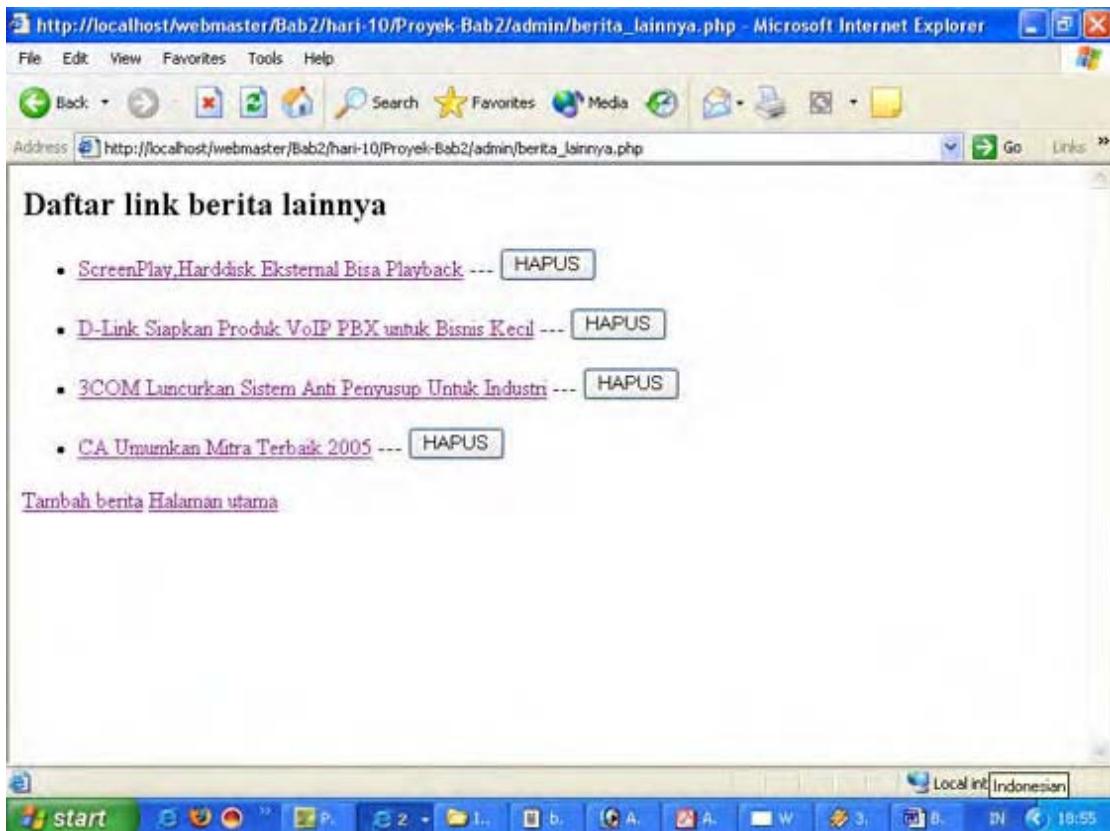
Simpan dengan nama ***berita\_lainnya.php*** letakkan pada direktori **Proyek-Bab2\admin\**. Sekarang coba anda jalankan file ini lalu tambahkanlah beberapa berita. Nama berita terserah anda karena hanya bersifat testing saja.

### PENJELASAN SCRIPT

Secara garis besar cara pemrosesan pada file *berita\_lainnya.php* sama dengan file *berita\_utama.txt*. Hanya ada sedikit perbedaan pada kasus hapus dan proses\_tambah. Jika pada hapus sebelumnya yang kita jadikan perbedaan adalah judul berita maka kali ini yang kita gunakan adalah nomor berita tersebut. Kita post data ini menggunakan nama ***id*** yang input typenya hidden.

Pada proses proses\_tambah ada bagian yang menarik yaitu saat kita ingin menambah nomor berita setiap file itu ditulis. Pada kasus diatas kita menggunakan variabel **\$file\_cek** untuk

menyimpan nilai dari array file *berita\_lainnya.txt*. Lalu kita gunakan fungsi `end()` untuk memindahkan posisi array yang aktif ke elemen terakhir. Selanjutnya kita memecah setiap bagian dengan `explode()` untuk mendapatkan nomor berita. Dalam hal ini nomor berita ada pada array `$bagian[0]`. Langkah selanjutnya adalah kita membuat variabel `$id` yang isinya adalah nilai dari nomor berita terakhir ditambah 1.



Gambar 2.12: output file berita\_lainnya saat proses lihat.

File untuk administrator yang terakhir adalah *iklan.php*. Fungsi file ini untuk menampilkan dan menghapus iklan saja. Kita tidak menambahkan fasilitas tambah iklan pada admin area karena pengunjunglah yang akan mengisi ini. File *iklan.txt* nantinya akan kita simpan pada direktori **berita**. Buka PHP DESIGNER 2005 buat file PHP baru kemudian ketik kode berikut:

#### Script dari file *iklan.php*

```
<?php

// mulai session
session_start();
require('fungsi.php');
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'lihat';

if (!cek_session())
 exit('Anda belum login. Login dulu');

switch ($proses)
{
 case 'lihat':
 echo '<h2>Daftar Iklan</h2><hr>';
 $iklan = file('../berita/iklan.txt');
 $jml_iklan = count($iklan);
 for ($i=0; $i<$jml_iklan; $i++)
 {
```

```

$bagian = explode("#-#", $iklan[$i]);
echo '<form action="iklan.php?proses=hapus" method="post">'
. '<table border="0">'
. '<tr>'
. '<td>Isi Iklan: </td>'
. '<td>'. $bagian[0]. '</td></tr>'
. '<tr>'
. '<td>Link: </td>'
. '<td>'. $bagian[1]. '</td></tr>'
. '<tr>'
. '<td><input type="submit" value="HAPUS"></td></tr>'
. '</table>'
. '<input type="hidden" name="judul" value="'. $bagian[0]. '">'
. '</form>';
}
echo 'Halaman utama';
break; // akhir dari proses lihat

case 'hapus':
$judul = $_POST['judul'];
$iklan_hapus = file('../berita/iklan.txt');
$file_baru = fopen('../berita/iklan.txt', 'w');

foreach ($iklan_hapus as $data)
{
 $data = str_replace("\n", "", $data);
 $bagian = explode("#-#", $data);
 // tulis kembali iklan yang isinya bukan iklan yang dihapus
 if ($bagian[0] != $judul)
 fwrite($file_baru, $data."\n");
}
fclose($file_baru);
echo 'Iklan berhasil dihapus. Lihat';
break; // akhir dari proses hapus
}
?>

```

Simpan pada direktori **Proyek-Bab2\admin\** dengan nama ***iklan.php***.

## PENJELASAN SCRIPT

Tidak ada yang istimewa pada script ini. Masih sama seperti kedua file sebelumnya. Disini pada proses hapus yang digunakan sebagai pembanding adalah judul dari berita.

Jalankan PHP DESIGNER 2005 buat file baru klik **File-New-PHP**. Lalu ketik kode berikut ini.

Script dari file ***myskin.php***

```

<html>
<head>
<title>JAVA ONLINE NEWS</title>
<style>
 body{font-family: verdana; font-size: 11px; margin-top: 0; margin-left: 0;
margin-right: 0}
 td{font-family: verdana; font-size: 11px}
 h2{font-family: verdana; color: white}
 table{border-color: #cccccc; border-collapse: collapse}
 .putih{color: white; font-weight: bold}
 a{color: #7cb500}
 a:visited{color: #7cb500}
</style>
</head>
<body>

```

```

<table border="1" width="100%" cellpadding="4" bordercolor="#cccccc">
<tr bgcolor="#7cb500">
 <td colspan="3" align="center" height="50"><h2>{JUDUL}</h2></td>
</tr>
<tr>
 <td colspan="2" align="right"><marquee scrollamount="4">{SLOGAN}</marquee></td>
 <td>{TODAY}</td>
</tr>
<tr bgcolor="#7cb500" height="25" class="putih">
 <td width="25%">berita lainnya</td>
 <td width="50%">berita utama</td>
 <td width="25%">iklan</td>
</tr>
<tr>
 <td width="25%" valign="top">{LAINNYA}</td>
 <td width="50%" valign="top">{UTAMA}</td>
 <td width="25%" valign="top">{IKLAN}</td>
</tr>
<tr>
 <td colspan="3" align="center" height="25">{FOOTER}</td>
</tr>
</table>

</body>
</html>

```

Simpan dengan nama ***myskin.php***, letakkan di direktori **Proyek-Bab1\template**. Coba anda perhatikan output dari file ini yang ditunjukkan pada gambar 2.13.



Gambar 2.13: tampilan *myskin.php* saat dijalankan.

Selanjutnya kita file yang akan kita buat kita gunakan untuk mengembalikan ke halaman utama jika ada yang mencoba masuk direktori template. Buat file PHP baru kemudian ketik kode berikut.

#### Script dari file *index.php*

```

<?php
// panggil fungsi header untuk mencetak HTML header
// gunakan header location untuk membuka halaman
header("Location: ../index.php");
?>

```

Simpan pada direktori **Proyek-Bab2\template** dengan nama ***index.php***. Jika anda mencoba membuka direktori template dari URL maka anda akan dibawa kembali ke halaman utama.

Berikutnya kita akan membuat file ***template.php***. Isi dari file ini merupakan class skin. Yang digunakan untuk mengganti tag dan skin. File template.php ini kita nantinya akan kita panggil

dari file index.php. Langsung saja, buat file PHP baru pada PHP DESIGNER 2005. Ketik kode berikut.

#### Script dari file ***template.php***

---

```
<?php

class skin
{
 var $tag = array();
 var $file_skin;
 var $halaman;

 function ganti_tag($namatag, $str_pengganti)
 {
 $this->tag[$namatag] = $str_pengganti;
 }

 function ganti_skin($namafile)
 {
 $this->file_skin = $namafile;
 }

 function ganti_tampilan()
 {
 $this->halaman = file($this->file_skin);
 $this->halaman = implode("", $this->halaman);
 foreach($this->tag as $str_dicari => $str_baru)
 {
 $this->halaman = ereg_replace($str_dicari, $str_baru, $this-> halaman);
 }
 echo $this->halaman;
 }
}

?>
```

---

Simpan dengan nama ***template.php*** letakkan file berikut pada direktori **Proyek-Bab2\template**.

File ini pernah kita buat pada bahasan OOP. Namun ada sedikit perubahan yaitu pada deklarator properti. Jika pada bahasan OOP kita menggunakan keyword **private** kali ini kita menggunakan **var**. Hal ini kita gunakan untuk menjaga kompatibilitas dengan PHP versi 4 yang tidak mendukung keyword **private**, **public** dan **protected**.

Hampir lupa!, direktori file index.php pada direktori **berita** kan belum ada?!?. Jadi kita akan membuatnya sekarang. Pada PHP DESIGNER 2006 buat file PHP baru kemudian ketik script berikut.

#### Script dari file ***index.php***

---

```
<?php
// panggil fungsi header untuk mencetak HTML header
// gunakan header location untuk membuka halaman
header("Location: ../index.php");
?>
```

---

Simpan dengan nama ***index.php*** dan letakkan pada direktori **Proyek-Bab2\berita**. Sama seperti sebelumnya tugas file ini hanya melakukan redirect halaman ke halaman utama.

Berikutnya kita akan membuat file ***var\_tampilan.php***. File ini menyimpan variabel-variabel yang berisi tampilan yang kita gunakan untuk mengganti tag {X}. Nama variabel kita sesuaikan dengan nama tag yang akan kita ganti. Ini berguna untuk memudahkan kita dalam penggantian tampilan.

Karena pada variabel ini kita akan membuka file berita dan menggunakan fungsi explode untuk mendapatkan isi berita. Maka format yang kita gunakan pada file-file kita nantinya adalah harus kita sesuaikan dengan format yang ada pada tabel 2.6.

Ketiga file tersebut menggunakan tanda #-# sebagai pemisah antar isi. Langsung saja Jalankan PHP DESIGNER 2005 klik **File-New-PHP** lalu ketik kode berikut:

#### Script dari ***var\_tampilan.php***

---

```
<?php

// variabel-variabel berikut akan digunakan untuk mengganti {NAMATAG}
$judul = "JAVA ONLINE NEWS";
$slogan = "<< Koran Online Nomor Satu Di Indonesia . . .";

$nama_hari = array('Minggu', 'Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat',
 'Sabtu');
$nama_bulan = array('Januari', 'Februari', 'Maret', 'April', 'Mei', 'Juni',
 'Juli', 'Agustus', 'September', 'Oktober', 'Nopember',
 'Desember');
$tanggal = date('j'); // tanggal sekarang 1-31
$hari = date('w'); // kode hari 0=minggu, 1=senin dst.
$bulan = date('n') - 1; //dikurangi satu agar index awal 0
$tahun = date('Y');

$hari_ini = $nama_hari[$hari];
$bulan_ini = $nama_bulan[$bulan];
$today = $hari_ini.'.'.$tanggal.'.'.$bulan_ini.'.'.$tahun;

// buka file berita_utama.txt
$berita_utama = file('berita/berita_utama.txt');
// urutkan mulai dari elemen array yang terakhir
$isi_utama = end($berita_utama);
while($isi_utama)
{
 $bagian = explode("#-#", $isi_utama);
 $utama .= $bagian[0]."
".$bagian[1]."".
 "<p>".$bagian[2]."</p>";
 $isi_utama = prev($berita_utama); // mundur satu elemen
}

// tampilkan berita lainnya (hanya berupa link)
$berita_lainnya = file('berita/berita_lainnya.txt');
$lainnya = ""; // kita beri tanda bullet
// urutkan mulai dari elemen array yang terakhir
$isi_lainnya = end($berita_lainnya);
while ($isi_lainnya)
{
 $bagian = explode("#-#", $isi_lainnya);
 $lainnya .= "<p>".$bagian[1]."</p>";
 $isi_lainnya = prev($berita_lainnya); // mundur satu elemen
}
$lainnya .= "";

// tampilkan iklan
$iklan = '<p>Ingin pasang iklan Klik disini '.
 '. GRATIS!.</p>';
$file_iklan = file('berita/iklan.txt');
```

```
// urutkan mulai dari elemen array yang terakhir
$isi_iklan = end($file_iklan);
while($isi_iklan)
{
 $bagian = explode("#-#", $isi_iklan);
 $iklan .= $bagian[0]. '
' .
 . 'Klik disini<p></p>';
 $isi_iklan = prev($file_iklan); // mundur satu elemen
}

$footer = '(c) 2005 javaonlinenews.localhost';
?>
```

Simpan dengan nama ***var\_tampilan.php***. Letakkan file berikut pada direktori **Proyek-Bab2\template\**.

### PENJELASAN SCRIPT

Untuk variabel **\$today** tidak perlu saya jelaskan lagi. Karena prinsip kerjanya sama dengan file yang pernah kita buat untuk materi *tanggal dan waktu*. Yang perlu kita bahas disini adalah isi dari variabel **\$utama**. Lihat cuplikan script berikut,

```
$berita_utama = file('berita/berita_utama.txt');
//mulai dari elemen array yang terakhir
$isi_utama = end($berita_utama);
while($isi_utama)
{
 $bagian = explode("#-#", $isi_utama);
 $utama .= $bagian[0]. "
". $bagian[1] . "" .
 . "<p>". $bagian[2] . "</p>";
 $isi_utama = prev($berita_utama); //mundur satu elemen
}
```

Pada cuplikan kode diatas hal pertama yang kita lakukan adalah membuka file *berita\_utama.txt* dengan *file()*. Jadi variabel *\$berita\_utama* merupakan array yang isinya adalah setiap baris dari file *berita\_utama.txt*. Karena berita baru yang kita tulis otomatis berada pada baris yang terakhir. Maka array *\$berita\_utama* kita urut mulai dari baris yang terakhir hingga baris pertama. Untuk itu kita gunakan fungsi *end(\$berita\_utama)* yang akan memindahkan posisi array yang aktif ke elemen yang paling akhir.

Agar dapat kita gunakan pada looping while maka kita masukkan nilai dari fungsi *end(\$berita\_utama)* pada variabel *\$isi\_utama*. Jadi isi dari *\$isi\_utama* pada saat looping while adalah elemen array yang aktif mulai dari elemen array terakhir pada eksekusi pertama. Hingga elemen array paling awal pada eksekusi terakhir. Ingat, karena pada blok looping maka, kode blok pada looping dalam kasus ini while akan dieksekusi sebanyak jumlah elemen array yang tidak lain adalah banyaknya baris pada file.

Selanjutnya pada blok kode while tersebut kita membuat variabel array *\$bagian* yang isinya adalah pecahan dari setiap berita. Kita memecah setiap bagian yang dipisahkan tanda *#-#* dengan menggunakan fungsi *explode()*. Jadi *\$bagian[0]* merupakan tanggal berita, *\$bagian[1]* merupakan judul berita dan *\$bagian[2]* merupakan isi berita.

Variabel *\$utama* kita berikan tanda *.=* agar isi dari variabel *\$utama* sebelumnya tidak tertindih nilai dari variabel berikutnya akibat adanya eksekusi berulang karena looping.

Dan yang terakhir kita set nilai dari *\$isi\_utama* agar berpindah pada elemen array sebelumnya atau mundur satu elemen. Untuk itu kita gunakan perintah *prev(\$berita\_utama)*. Jika sudah sampai pada elemen array paling awal maka eksekusi akan dihentikan karena sudah tidak ada lagi elemen array. Untuk variabel *\$lainnya* dan *\$iklan* prinsip kerjanya sama. Jadi tidak perlu saya jelaskan ulang.

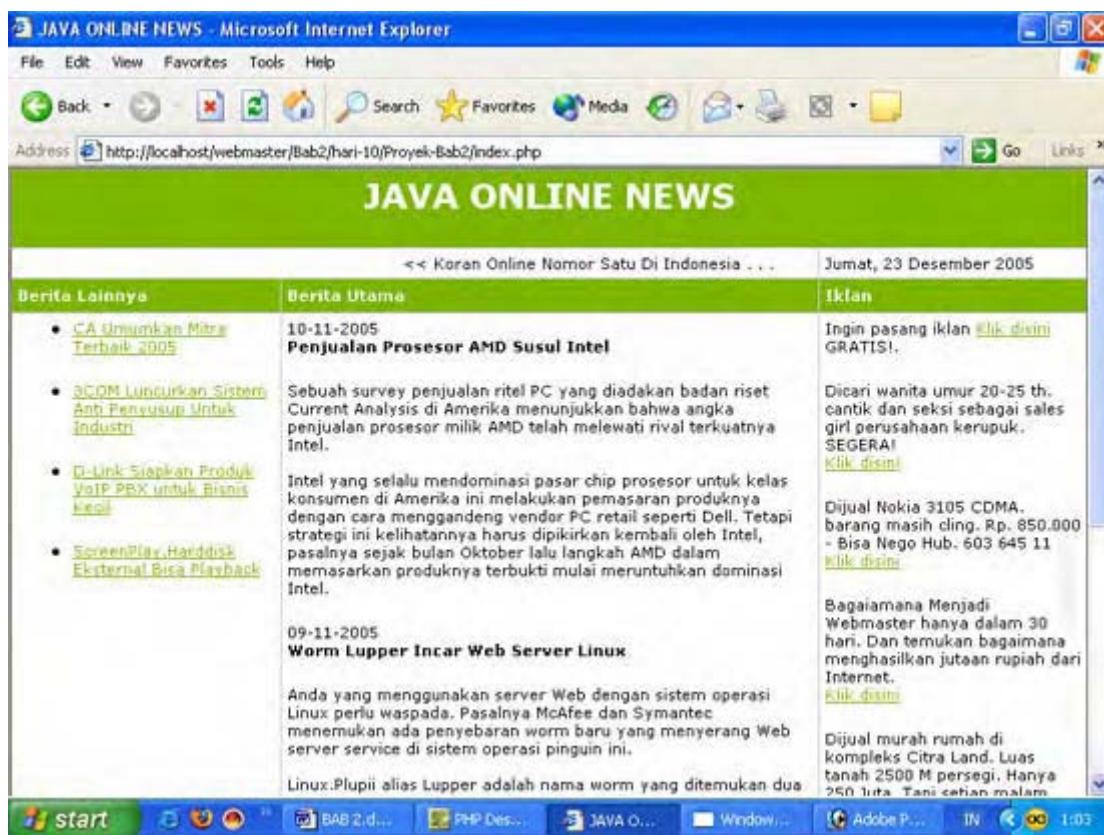
Selanjutnya kita akan membuat file yang paling penting yaitu ***index.php***. Ingat WebServer akan selalu mencari file *index.php* atau *index.html* pada suatu direktori jika tidak ditemukan maka webserver akan memunculkan daftar file pada direktori tersebut. Jalankan PHP DESIGNER 2005 klik menu **File-New-PHP** kemudian ketik listing kode berikut

#### Script dari file ***index.php***

```
<?php
// panggil file template.php pada direktori template
require('template/template.php');
// panggil file var_tampilan.php yang menyimpan variabel tampilan
require('template/var_tampilan.php');

$skin = new skin;
$skin->ganti_skin('template/myskin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{SLOGAN}', $slogan);
$skin->ganti_tag('{TODAY}', $today);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{LAINNYA}', $lainnya);
$skin->ganti_tag('{IKLAN}', $iklan);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();
?>
```

Simpan dengan nama ***index.php*** letakkan pada direktori **Proyek-Bab2\**. Setelah itu coba jalankan hasilnya akan terlihat seperti gambar 2.14



Gambar 2.14: output file *index.php*.

#### PENJELASAN SCRIPT

Pada script *index.php* diatas kita memanggil dulu class skin yang ada pada file *template.php* dan memanggil *var\_tampilan.php* yang berisi variabel-variabel tampilan yang digunakan untuk mengganti tag {X}. Lalu kita membuat objek \$skin yang merupakan class skin.

Disini kita menggunakan tiga metode yaitu **ganti\_skin()** untuk mengganti skin template. **ganti\_tag()** untuk mengganti tag-tag {X} pada halaman. Dan yang terakhir adalah melakukan penggantian tampilan menggunakan metode **ganti\_tampilan()**.

File untuk bagian pengunjung lainnya adalah *pasangiklan.php*. File ini berguna untuk mempost iklan yang dipasang user. Namun juga ada batasan user tidak boleh memasang iklan lebih dari satu kali. Untuk memahami cara kerjanya buat file baru pada PHP DESIGNER 2005 lalu ketik kode berikut

Script dari file ***pasangiklan.php***

```
<?php
// panggil output buffering untuk cookie
ob_start();
// daftarkan proses
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'pasang';
?>
<html>
<head>
<title>Pasang Iklan</title>
<style>
 td{font-family: verdana; font-size: 11px}
 body{font-family: verdana; font-size: 11px}
 .putih{color: white; font-weight: bolder}
</style>
</head>
<body>
<center>
<?php
switch ($proses)
{
 case 'pasang':
 echo '<form action="pasangiklan.php?proses=proses_pasang" method="post">
 .<table border="0" width="75%">
 .<tr height="30" bgcolor="#7cb500">
 .<td align="center" class="putih">Pasang Iklan</td></tr>
 .<tr>
 .<td>Isi Iklan: (max. 100 karakter)

 .<input type="text" name="iklan" size="80" maxlength="100"></td></tr>
 .<tr>
 .<td>Link Iklan:

 .<input type="text" name="link" size="30">

 .cukup ketikkan domainnya saja tanpa http:// contoh www.domain.com. '
 .</td></tr>
 .<tr>
 .<td><input type="submit" value="PASANG"></td></tr>
 .<tr height="30" bgcolor="#7cb500">
 .<td align="center" class="putih">(c) 2005 - '
 .<td>javanewsonline.localhost</td></tr>
 .</table>
 .</form>';
 break; // akhir dari proses pasang

 case 'proses_pasang':
 // cek isi dari cookie iklan jika ada maka tidak boleh posting
 if (isset($_COOKIE['iklan']))
 echo "Anda tidak diperbolehkan memasang iklan lebih dari 1x.";
 else
 {
 $iklan = $_POST['iklan'];

```

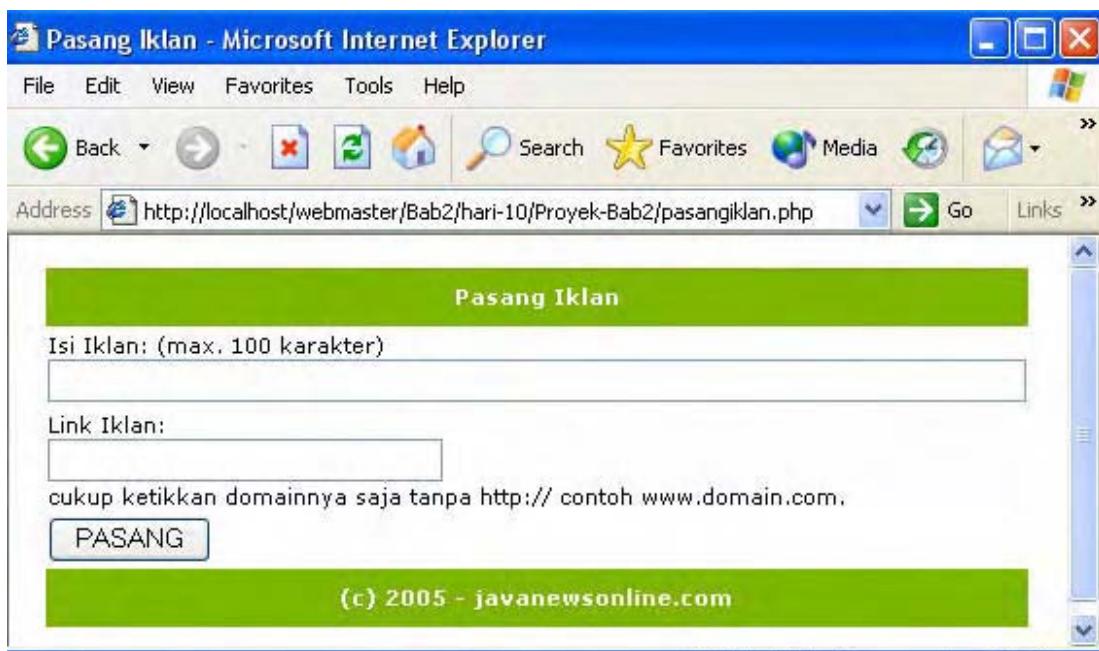
```

$link = $_POST['link'];
if (empty($iklan) || empty($link))
 exit('Error: Ada field kosong');

// buka file iklan.txt
$buka_iklan = fopen('berita/iklan.txt', 'a');
$link = 'http://'.$link.'/';
$isi_file = $iklan.'#-'.$link."\n";
// tulis ke file iklan.txt
fwrite($buka_iklan, $isi_file);
fclose($buka_iklan);
echo 'Iklan berhasil dipost. Lihat.';
// buat cookie iklan
setcookie('iklan', '1');
}
break; //akhir dari proses_pasang
}
?>
</body>
</html>

```

Simpan dengan nama ***pasangiklan.php*** pada direktori **Proyek-Bab2\**. Sekarang coba klik pada link iklan klik disini. Maka anda akan dibawa ke form pasang iklan seperti gambar 2.



Gambar 2.: output file pasangiklan.php

### PENJELASAN SCRIPT

Pada script ini kita pertama kali memanggil fungsi **ob\_start()**. Fungsi ini kita panggil karena nantinya kita menggunakan cookie. Ada dua proses yang dikerjakan file ini yaitu menampilkan form dan memasang atau menulis iklan.

Pada proses *pasang* kita hanya menampilkan form untuk memasang iklan. Yang terdiri dari dua textbox. Selanjutnya pada proses *proses\_pasang* kita dihadapkan pada dua kondisi. Yaitu untuk user yang belum pernah memasang iklan dan yang sudah pernah memasang iklan. Untuk itu kita gunakan statemen **if(isset(\$\_COOKIE['iklan']))**. Jika isi dari cookie iklan tidak kosong maka user sudah pernah memasang iklan. Untuk itu keluar dari rutin script dan tampilkan pesan bahwa ia tidak boleh memasang lebih dari satu kali.

Seperti biasa untuk menulis file kita perlu membukanya lebih dahulu dengan **fopen()** lalu menulisnya. Pada akhir dari proses *proses\_tambah* kita membuat cookie yang bernama iklan

dan isinya 1. Sebenarnya isi dari cookie ini tidak penting karena pada pengecekan kita hanya menentukan cookie tersebut terisi atau tidak.

Jika anda ingin memasang iklan lagi anda harus menghapus cookienya. Kita dapat melakukannya secara manual pada Internet Explorer. Caranya klik menu **Tools-Internet Options...-Delete Cookies...** Setelah itu tekan tombol refresh pada toolbar. Jika masih belum bisa tutup IE lalu buka kembali. Sekarang anda dapat memasang iklan kembali.

\*\*\*\*\*

## Kalender

27	28	29	30	31	1	2	3
3	4	5	6	7	8	9	
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

## Berikutnya

Pada bab berikutnya kita akan mempelajari tentang database yaitu MySQL, phpMyAdmin, dan Mengkoneksikan PHP dengan MySQL.

# Bab

# 3

- MySQL
- phpMyAdmin
- Mengakses MySQL  
Menggunakan PHP

Sebelum melanjutkan ikuti langkah-langkah berikut:

- Buatlah folder baru pada direktori C:\Apache2\htdocs\webmaster\
- Berinama folder tersebut **Bab3**

\*\*\*\*\*

Kita sudah menginjak hari ke sebelas untuk itu:

- Masuklah pada direktori C:\Apache2\htdocs\webmaster\Bab3\
- Buatlah folder baru dan beri nama **hari-11**

## MySQL

Pada hari ke sebelas ini materi yang akan kita pelajari meliputi database. Untuk pengolahan database ini kita menggunakan MySQL. Jadi diharapkan nantinya anda dapat menguasai beberapa materi tentang:

- Menggunakan MySQL
- Membuat Database
- Memasukkan data ke database
- Mendapatkan data dari database
- Mengupdate record pada database
- Meng-Alter tabel
- Menghapus record pada database
- Menghapus tabel
- Menghapus database
- Teknik MySQL Lainnya

Itulah beberapa materi yang harus anda kuasai. Sebenarnya materi ini tidak terlalu sulit. Jika anda pernah menggunakan program spreadsheet macam Microsoft Excel atau Microsoft Access. Anda akan mengerti dengan cepat bagaimana membuat database. Karena database dapat diartikan sebagai kumbulan dari banyak tabel dapat yang saling berhubungan (Relational).

### Menggunakan MySQL

MySQL merupakan salah satu aplikasi database yang menggunakan SQL sebagai bahasa standar. SQL sendiri merupakan singkatan dari *Structured Query Language*. SQL digunakan untuk menyimpan dan mendapatkan data dari dan ke sebuah database. Aplikasi database lain diantaranya Microsoft SQL Server, Oracle, PostgreSQL dan SyBase. Pada database dikenal istilah query. Query merupakan perintah yang digunakan untuk melakukan manipulasi pada database.

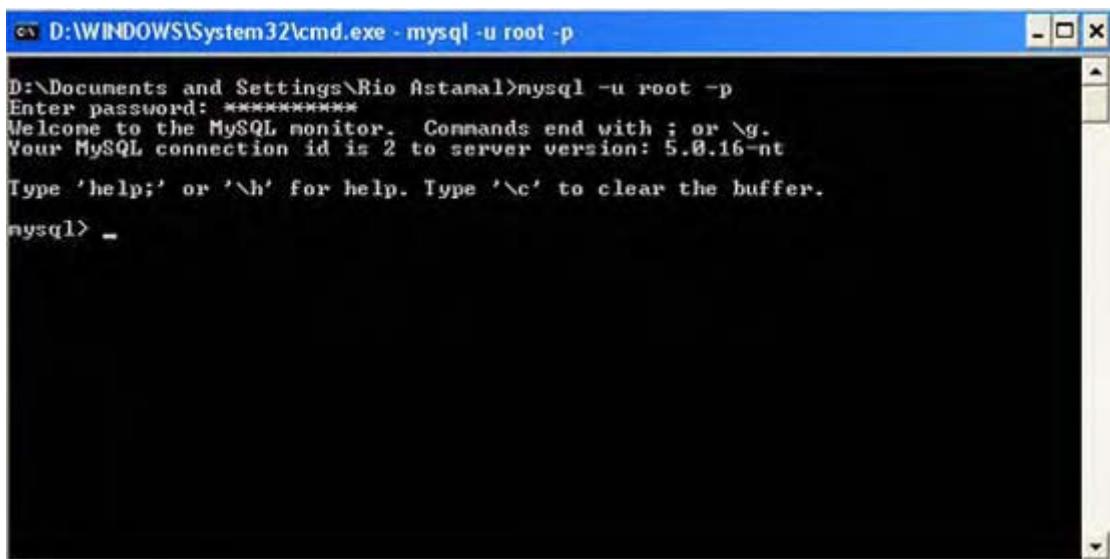
Untuk memulai MySQL jalankan **command prompt** lalu ketikkan:

C:\>mysql -u root -p [ENTER]

Lalu ketikkan password anda:

Enter password: \*\*\*\*\* [ENTER]

Jika benar maka akan muncul prompt `mysql>` seperti gambar 3.0. Untuk mendapatkan daftar dari perintah yang dapat anda gunakan pada mysql prompt ketikkan `\h` atau `help` lalu tekan ENTER. Perintah-perintah seperti `help`, `use`, `go`, `exit` dan sebagainya juga memiliki sinonim perintah. Artinya perintah-perintah tersebut dapat kita panggil dengan cara yang berbeda. Contohnya untuk `exit` kita dapat menggunakan perintah `\q` [ENTER]. Keduanya sama-sama valid. Perlu anda ketahui untuk perintah-perintah yang berhubungan dengan query database akhir dari perintah tersebut harus diberi tanda ";" (titik koma).



```
D:\Documents and Settings\Rio Astamal>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2 to server version: 5.0.16-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

Gambar 3.0: tampilan utama dari MySQL monitor

## Membuat Database

Untuk membuat database perintahnya sangatlah simpel. Query untuk membuat database adalah `create database nama_database`. Pada mysql prompt ketikkan perintah berikut:

```
mysql> create database karyawan;
```

Arti dari perintah diatas adalah kita membuat database dengan nama karyawan. Jika tidak ada kesalahan maka akan muncul pesan seperti berikut

```
Query OK, 1 row affected (0.04 sec)
```

Itu artinya semua pembuatan database berhasil dilakukan. Jika anda tidak mendapatkan hasil seperti diatas pastikan anda telah memberi titik koma pada akhir perintah. Tanda titik koma memberitahu MySQL query yang anda berikan telah selesai dan MySQL harus mengeksekusinya.

Sekarang mari kita lihat apakah benar database kita sudah ada atau belum. Untuk itu pada mysql prompt ketik perintah berikut

```
mysql> show databases;
```

Hasil yang ditampilkan kurang lebih akan seperti berikut

-----	-----
Database	
-----	-----
mysql	
karyawan	
-----	-----

Ternyata memang benar database karyawan berhasil dibuat sehingga ada pada daftar database. Perintah `show` juga dapat digunakan untuk menampilkan tabel pada database.

## Mengaktifkan Database

Sebelum dapat melakukan banyak hal pada database. Anda perlu mengaktifkan dulu database mana yang akan menerima perlakuan query. Untuk mengaktifkan database digunakan perintah `use`, lihat contoh berikut

```
mysql> use karyawan;
```

Perintah diatas akan mengaktifkan database karyawan. Sehingga nantinya database inilah yang akan menerima query-query dari kita. Jika tidak ada kesalahan pesan yang dimunculkan MySQL adalah Database changed. Untuk mengetahui daftar tabel yang ada pada suatu database dapat digunakan perintah show tables.

```
mysql> show tables;
```

Pesan yang muncul adalah Empty set (0.00 sec). Itu berarti tidak ada tabel pada database karyawan. Untuk itu kita akan segera membuatnya. Namun sebelum membuat tabel ada beberapa hal yang perlu anda ketahui. Untuk membuat tabel tentu terdapat kolom-kolom. Pada saat menyimpan data ke tabel jenis data yang kita masukkan tentunya berbeda-beda. Seperti string, integer, date/time dan lain sebagainya. Tabel-tabel berikut menuntukan tipe-tipe data dan ukurannya disimpan pada MySQL.

## Tipe Data Numerik

Pada tipe data ini kolom dapat berisi angka baik bilangan cacah ataupun real. Untuk daftar lengkap tipe data numerik dapat anda lihat pada tabel 3.0.

**Tabel 3.0** Daftar tipe data numerik

Tipe	Jangkauan	Ukuran(byte)	Deskripsi
TINYINT	0...255 atau -127...128	1	Angka yang sangat kecil
BIT			Sama dengan TINYINT
BOOL			Sama dengan TINYINT
SMALLINT	-32768...32367 atau 0...65535	2	Angka yang kecil
MEDIUMINT	- 8388608...8388607 atau 0...16777215	3	Angka sedang
INT	- $2^{31}$ ... $2^{31}$ -1 atau 0... $2^{32}$ -1	4	Angka regular
INETEGER			Sama dengan INT
BIGINT	- $2^{63}$ ...- $2^{63}$ -1 atau 0... $2^{64}$ -1	8	Angka Yang besar

## Tipe Data Date and Time

Pada tipe date and time dapat menyimpan data berdasarkan tanggal dan waktu. Untuk daftar lengkap tipe data ini lihat tabel 3.1.

**Tabel 3.1** Daftar tipe data Date and Time

Tipe	Jangkauan	Deskripsi
Date	1000-01-01 sampai 9999-12-31	Tanggal yang ditampilkan dalam format YYYY-MM-DD
TIME	-838:59:59 sampai 838:59:59	Waktu yang ditampilkan dalam format HH:MM:SS. Jangkauannya sangat panjang mungkin lebih dari yang anda inginkan
DATETIME	1000-01-01 00:00:00 sampai 9999-12-31 23:59:59	Tanggal dan Waktu yang ditampilkan dalam format YYYY-MM-DD HH:MM:SS
TIMESTAMP	1970-01-01 00:00:00	Timestamp cukup berguna jika digunakan dalam transaksi. Format lengkap dapat anda lihat pada tabel 3.2

YEAR	70-69 atau 1970-2069	Untuk YEAR(2) tahun dalam 2 digit. Sedangkan YEAR(4) dalam 4 digit.
------	----------------------	------------------------------------------------------------------------

**Tabel 3.2** Daftar format TIMESTAMP

Format	Tampilan
TIMESTAMP	YYYYMMDDHHMMSS
TIMESTAMP (14)	YYYYMMDDHHMMSS
TIMESTAMP (12)	YYMMDDHHMMSS
TIMESTAMP (10)	YYMMDDHHMM
TIMESTAMP (8)	YYYYMMDD
TIMESTAMP (6)	YYMMDD
TIMESTAMP (4)	YYMM
TIMESTAMP (2)	YY

## Tipe Data String

Kolom pada tipe ini tidak hanya dapat menyimpan data dalam format teks biasa. Tetapi juga dapat menyimpan dalam bentuk binary seperti gambar dan sound. Untuk teks biasa kita dapat menggunakan CHAR dan VARCHAR. Perbedaan keduanya adalah jika membuat kolom dengan tipe data CHAR MySQL akan menambahkan spasi pada data tersebut jika belum memenuhi data maksimal yang dapat ditampung. Sedangkan VARCHAR tetap menyimpan data itu apa adanya.

Tipe data berikutnya yang dapat kita gunakan adalah TEXT dan BLOB. Kedua tipe ini memiliki banyak variasi ukuran. Seperti yang pernah disinggung sebelumnya. Kita dapat menyimpan data dalam bentuk binari. Yang dapat melakukan hal tersebut adalah tipe BLOB. BLOB merupakan *Binary Large Objects*. Jadi pada tipe ini anda dapat menyimpan file-file gambar ataupun suara.

Dalam prakteknya tipe kolom BLOB dan TEXT hampir sama. Hanya saja pada tipe TEXT bersifat case sensitive artinya membedakan penggunaan huruf kecil dan besar sedangkan BLOB tidak. Lihatlah tabel berikut untuk daftar tipe data string.

**Tabel 3.3** Daftar Tipe Data string

Tipe	Jangkauan	Deskripsi
CHAR(n)	0...255	Karakter sepanjang n. Dimana n antara 0 sampai 255
CHAR	1	Sama dengan CHAR(1)
VARCHAR(n)	1...255	Sama dengan CHAR(n) hanya perbedaan panjang variabel
TINYBLOB	$2^8$ -1 (atau 255)	BLOB ukuran kecil
TINYTEXT	$2^8$ -1 (atau 255)	TEXT ukuran kecil
BLOB	$2^{16}$ -1 (atau 65.535)	BLOB ukuran normal
TEXT	$2^{16}$ -1 (atau 65.535)	TEXT ukuran normal
MEDIUMBLOB	$2^{24}$ -1 (atau 16.777.215)	BLOB ukuran medium
MEDIUMTEXT	$2^{24}$ -1 (atau 16.777.215)	TEXT ukuran medium
LONGBLOB	$2^{32}$ -1 (atau 4.294.967.295)	BLOB ukuran besar
LONGTEXT	$2^{32}$ -1 (atau 4.294.967.295)	TEXT ukuran besar

## Membuat Tabel

Seperti yang kita tahu tabel terdiri dari kolom dan baris. Dan setiap kolom biasanya memiliki tipe data yang berbeda. Untuk itu sebelum membuat tabel kita harus tahu setiap kolom menyimpan data apa. Sehingga pada saat membuat tabel kita dapat menentukan tipe data

---

yang tepat untuk setiap kolom. Untuk membuat tabel kita dapat menggunakan perintah create table nama\_tabel. Pada mysql prompt ketiklah perintah berikut, namun pastikan database karyawan telah aktif.

```
mysql> create table data_pribadi
-> (
-> kode_kar varchar(4) not null,
-> nama_kar varchar(30),
-> alamat varchar(50),
-> jabatan varchar(15),
-> gaji mediumint(10),
-> primary key(kode_kar)
->) type='MyISAM';
```

Jika tidak ada kesalahan maka akan muncul pesan `Query OK, 0 rows affected (0.09 sec)`. Pada query diatas kita membuat tabel yang bernama `data_pribadi` dengan tipe MyISAM (akan dibahas lebih lanjut). Tabel ini memiliki lima kolom. Untuk kolom satu sampai dengan empat tipe kolomnya adalah varchar. Sedangkan untuk kolom gaji karena merupakan angka maka tipenya kita buat mediumint. Pada kolom kode\_kar kita beri tanda `primary key`. Artinya data pada kolom ini harus bersifat unik, tidak boleh ada data yang sama.

Untuk mengetahui struktur tabel kita dapat menggunakan perintah `describe`. Ketik query berikut pada mysql prompt

```
mysql> DESCRIBE data_pribadi; [ENTER]
```

Tampilan yang muncul adalah seperti berikut

Field	Type	Null	Key	Default	Extra
kode_kar	varchar(4)	NO	PRI		
nama_kar	varchar(30)	YES		NULL	
alamat	varchar(50)	YES		NULL	
jabatan	varchar(15)	YES		NULL	
gaji	mediumint(10)	YES		NULL	

5 rows in set (0.02 sec)

## Memasukkan Data ke Database

Untuk memasukkan data ke dalam database perintah yang digunakan adalah `INSERT`. Struktur dasar dari penulisan query insert adalah sebagai berikut

```
INSERT [INTO] nama_table [(kolom1, kolom2, kolom3,...)] VALUES (Isi1, Isi2, Isi3,...);
```

Contoh berikut ini menunjukkan kita akan memasukkan data ke dalam tabel `data_pribadi`. Pada mysql prompt ketikkan perintah berikut

```
mysql> INSERT INTO data_pribadi VALUES ('001A', 'Web Janarko', 'Jl. Cyber 19',
'Manajer', 5000000); [ENTER]
```

Jika berhasil akan muncul pesan `Query OK, 1 row affected (0.01 sec)`. Perhatikan setiap nilai yang kita masukkan harus diberi tanda petik satu atau petik dua. Untuk jenis integer hal ini tidak diperlukan. Lalu keyword kolom1 dan seterusnya disebelah kanan nama table itu buat apa?. Itu digunakan jika kita ingin memasukkan data hanya pada kolom tertentu saja. Contoh ketik query berikut pada mysql prompt

```
mysql> INSERT INTO data_pribadi (kode_kar, nama_kar) VALUES ('002B', 'Surfi');
[ENTER]
```

---

Jika berhasil MySQL akan menampilkan `Query OK, 1 row affected (0.00 sec)`. Pada query diatas kita hanya memasukkan data pada kolom kode\_kar dan nama\_kar. Sedangkan isinya adalah 002B untuk kode karyawan dan Surfi untuk Nama. Pada penentuan kolom kita tidak perlu memberi tanda petik.

## Menyimpan Perintah Query pada File

Selain dengan mengetikkan langsung pada MySQL Monitor. Kita juga dapat menyimpan kumpulan perintah perintah query database pada sebuah file. Perintah yang kita tulis akan dieksekusi oleh MySQL secara urut dari atas ke bawah. Kita dapat menyimpan query untuk membuat database, memasukkan data, menghapus database dan sebagainya.

Untuk mencobanya buka PHP DESIGNER 2006 klik menu **File-New-SQL** untuk membuat file SQL. Lalu ketik kode berikut

```
-- masukkan data ke tabel data_pribadi
INSERT INTO data_pribadi VALUES ('003C', 'Hacky', 'Jl. Port 80', 'PR', 4000000);
INSERT INTO data_pribadi VALUES ('004D', 'Cracky', 'Jl. Port 21', 'Office Boy',
750000);
INSERT INTO data_pribadi VALUES ('005E', 'Preaky', 'Jl. Tembaga 911', 'PR',
3000000);
```

Simpan dengan nama `data_pribadi.sql`, letakkan file ini direktori hari-11. Untuk mengeksekusinya ada dua cara. Cara yang pertama adalah mengeksekusinya melalui mysql prompt. Ketik perintah berikut

```
mysql> source c:/Apache2/htdocs/webmaster/Bab3/hari-11/data_pribadi.sql;
```

Jika tidak ada kesalahan pesan yang muncul adalah  
Database changed

```
Query OK, 1 row affected (0.00 sec)
```

```
Query OK, 1 row affected (0.01 sec)
```

```
Query OK, 1 row affected (0.00 sec)
```

Cara yang kedua adalah melalui DOS prompt bukan mysql prompt. Jika anda masih pada mysql prompt ketikkan `\q` [ENTER] untuk keluar dari MySQL Monitor. Lalu pada DOS prompt ketikkan perintah berikut

```
C:\>mysql -u root -p -D karyawan < c:/Apache2/htdocs/webmaster/Bab3/hari-11/data_pribadi.sql [ENTER]
```

Masukkan password,  
Enter password: \*\*\*\*\* [ENTER]

Opsiion `-D` merupakan database yang digunakan yaitu karyawan. Jika berhasil maka tidak ada pesan kesalahan yang muncul. Namun jika muncul pesan seperti

```
ERROR 1062 (23000) at line 4: Duplicate entry '003C' for key 1
```

Itu artinya ada data yang sama pada kolom kode\_kar. Itu dikarenakan anda sudah memasukkan query itu sebelumnya pada mysql prompt. Ingat sifat kolom kode\_kar adalah **primary key** maka data tidak boleh ada yang sama. Jadi setiap data pada kolom ini harus bersifat *unik*.

## Mendapatkan Data dari Database

Untuk keperluan ini query SQL yang kita gunakan adalah `SELECT`. Query ini kita gunakan untuk mendapatkan data dari database dengan menyeleksi baris yang cocok dengan kriteria

---

tertentu. Ada banyak variasi dalam menggunakan query SELECT. Struktur dasar penggunaan SELECT adalah sebagai berikut

```
SELECT item FROM nama_tabel [WHERE kondisi]
 [GROUP BY tipe_group]
 [HAVING where_kondisi]
 [ORDER BY urut_secarra]
 [LIMIT kriteria_limit];
```

Kita akan membahasnya lebih detil lagi. Namun sebelumnya kita akan masuk kembali ke MySQL Monitor dulu. Pada DOS prompt ketik perintah berikut

```
C:\>mysql -u root -p [ENTER]
Enter password: ***** [ENTER]
```

Setelah masuk pada mysql prompt aktifkan database karyawan dengan mengetikkan perintah.

```
mysql> use karyawan; [ENTER]
```

Sekarang kita akan coba menampilkan data dari database karyawan. Khusunya pada tabel data\_pribadi. Query berikut ini akan menampilkan data yang ada pada kolom kode\_kar dan nama\_kar.

```
mysql> SELECT kode_kar, nama_kar FROM data_pribadi; [ENTER]
```

Output dari query diatas adalah sebagai berikut

```
+-----+-----+
| kode_kar | nama_kar |
+-----+-----+
| 001A | Web Janarko |
| 002B | Surfi |
| 005E | Preaky |
| 004D | Cracky |
| 003C | Hacky |
+-----+-----+
5 rows in set (0.00 sec)
```

Seperti yang anda lihat yang ditampilkan hanyalah data pada kolom kode dan nama karyawan. Itu karena pada query setelah select kita menuliskan kode\_kar, nama\_kar.

Oh..., jadi kalau ingin menampilkan semuanya saya harus menuliskan semua nama kolom. Pemikiran yang bagus namun sayangnya anda tidak perlu melakukannya. Kita cukup memberikan simbol asterik (\*) maka secara otomatis seluruh kolom akan diikutkan. Lihat contoh berikut

```
mysql> SELECT * FROM data_pribadi; [ENTER]
```

Output yang ditampilkan dari query diatas adalah

```
+-----+-----+-----+-----+-----+
| kode_kar | nama_kar | alamat | jabatan | gaji |
+-----+-----+-----+-----+-----+
| 001A | Web Janarko | Jl. Cyber 19 | Manajer | 5000000 |
| 002B | Surfi | NULL | NULL | NULL |
| 005E | Preaky | Jl. Tembaga 911 | PR | 3000000 |
| 004D | Cracky | Jl. Port 21 | Office Boy | 750000 |
| 003C | Hacky | Jl. Port 80 | PR | 4000000 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

## Mendapatkan data secara spesifik

---

Seperti yang sudah disinggung sebelumnya kita dapat melakukan pencarian data sesuai dengan kriteria yang kita maksud. Untuk itu kita dapat menggunakan klausula WHERE. Untuk lebih memahaminya ketik query berikut pada mysql prompt

```
mysql> SELECT * FROM data_pribadi WHERE jabatan='PR'; [ENTER]
```

Output yang ditampilkan MySQL adalah sebagai berikut

```
+-----+-----+-----+-----+-----+
| kode_kar | nama_kar | alamat | jabatan | gaji |
+-----+-----+-----+-----+-----+
| 005E | Preaky | Jl. Tembaga 911 | PR | 3000000 |
| 003C | Hacky | Jl. Port 80 | PR | 4000000 |
+-----+-----+-----+-----+-----+
2 rows in set (0.02 sec)
```

Seperti yang anda lihat MySQL hanya mengembalikan baris yang kolom jabatannya adalah PR. Pada klausula WHERE kita dapat melakukan perbandingan menggunakan operater matematik ataupun operator perbandingan seperti pada PHP. Contoh lain penggunaan klausula WHERE adalah sebagai berikut

```
mysql> SELECT * FROM data_pribadi WHERE jabatan='PR' AND gaji>3000000; [ENTER]
```

Query diatas akan menghasilkan tampilan sebagai berikut

```
+-----+-----+-----+-----+-----+
| kode_kar | nama_kar | alamat | jabatan | gaji |
+-----+-----+-----+-----+-----+
| 003C | Hacky | Jl. Port 80 | PR | 4000000 |
+-----+-----+-----+-----+-----+
1 row in set (0.02 sec)
```

Seharusnya yang keluar adalah dua baris. Namun karena adanya operator AND yang juga mengharuskan kolom gaji lebih dari 3000000. Maka yang ditampilkan hanya satu baris dalam hal ini karyawan yang bernama Hacky yang gajinya lebih dari 3000000.

## Mengurutkan Data

Kita juga dapat mengurutkan data yang ada. Untuk keperluan kita akan menggunakan klausula ORDER BY. Pada klausula ini kita dapat mengurutkan data dari yang terkecil ke yang terbesar (Ascending). Atau mulai dari yang terbesar ke yang terkecil(Descending). Lihat contoh berikut

```
mysql> SELECT * FROM data_pribadi ORDER BY gaji DESC;
```

Output yang dihasilkan MySQL adalah

```
+-----+-----+-----+-----+-----+
| kode_kar | nama_kar | alamat | jabatan | gaji |
+-----+-----+-----+-----+-----+
| 001A | Web Janarko | Jl. Cyber 19 | Manajer | 5000000 |
| 003C | Hacky | Jl. Port 80 | PR | 4000000 |
| 005E | Preaky | Jl. Tembaga 911 | PR | 3000000 |
| 004D | Cracky | Jl. Port 21 | Office Boy | 750000 |
| 002B | Surfi | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Secara default jika kita tidak mencantumkan ASC atau DESC. Maka yang dijadikan patokan adalah ASC. Seperti yang anda lihat pada query diatas kita mengurutkan data pada tabel berdasarkan kolom gaji secara descending(DESC). Anda dapat mengkombinasikan klausula ORDER BY dengan klausula lain seperti WHERE.

## Memilih baris yang ditampilkan

Pada saat melakukan query kita juga dapat menentukan baris mana yang ingin ditampilkan. Untuk itu kita dapat menggunakan klausula LIMIT. Klausula LIMIT sangat berguna pada beberapa aplikasi web. Seperti pada suatu halaman kita hanya ingin menampilkan 5 item dan sebagainya. Pada contoh berikut kita hanya ingin menampilkan record dari baris ke-3 hingga ke-5;

```
mysql> SELECT * FROM data_pribadi LIMIT 1, 3; [ENTER]
```

Output yang ditampilkan adalah

```
+-----+-----+-----+-----+-----+
| kode_kar | nama_kar | alamat | jabatan | gaji |
+-----+-----+-----+-----+-----+
| 002B | Surfi | NULL | NULL | NULL |
| 005E | Preaky | Jl. Tembaga 911 | PR | 3000000 |
| 004D | Cracky | Jl. Port 21 | Office Boy | 750000 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

Arti dari query diatas adalah "*pilih semua dari tabel data\_pribadi kembalikan sebanyak 3 baris, mulai dari baris ke-1*". Catatan, index baris pada MySQL dimulai dari 0 **bukan** 1.

## Mengupdate Record pada Database

Pada tabel data\_pribadi kita tahu bahwa pada data karyawan yang bernama Surfi datanya belum lengkap. Untuk itu kita perlu mengupdate data pada record ini. Query SQL yang dapat kita gunakan untuk mengupdate data adalah **UPDATE**. Karena kita hanya ingin mengupdate data dari karyawan yang bernama Surfi. Maka kita juga gunakan klausula **WHERE** agar tidak seluruh tabel ikut terupdate. Pada mysql prompt ketik perintah berikut

```
mysql> UPDATE data_pribadi SET alamat='Jl. Optik 80', jabatan='Sekretaris', gaji=3500000 WHERE nama_kar='Surfi'; [ENTER]
```

Output yang ditampilkan adalah

```
Query OK, 1 row affected (0.05 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

Untuk mengecek apakah memang sudah terupdate maka kita dapat menampilkannya menggunakan SELECT.

```
mysql> SELECT * FROM data_pribadi WHERE nama_kar='Surfi'; [ENTER]
```

Output yang ditampilkan MySQL adalah

```
+-----+-----+-----+-----+-----+
| kode_kar | nama_kar | alamat | jabatan | gaji |
+-----+-----+-----+-----+-----+
| 002B | Surfi | Jl. Optik 80 | Sekretaris | 3500000 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Ternyata memang benar record telah terupdate. Dan data alamat, jabatan dan gaji yang semula NULL sekarang sudah terisi.

## Meng-Alter Tabel

Suatu saat mungkin anda ingin merubah struktur dari tabel anda. Mungkin saja anda ingin merubah primary key, index menyisipkan kolom dan sebagainya. Untuk kerperluan ini kita dapat menggunakan query ALTER. Daftar lengkap query yang dapat kita lakukan dengan ALTER ditunjukkan pada tabel 3.4

**Tabel 3.4** Daftar query yang dipasangkan dengan ALTER TABLE

Query	Deskripsi
ADD [KOLOM] deskripsi_kolom [FIRST ATAU AFTER KOLOM ]	Menambahkan kolom baru pada lokasi yang telah ditentukan. Deskripsi_kolom memerlukan nama dan tipe sama dengan saat membuat tabel
ADD [KOLOM] deskripsi_kolom, deskripsi_kolom, ...	Menambahkan kolom pada akhir dari tabel.
ADD INDEX [index] (kolom,...)	Membuat index pada tabel di kolom yang ditentukan.
ADD PRIMARY KEY (kolom,...)	Membuat primary key pada kolom yang ditentukan.
ADD UNIQUE [index] (klom,...)	Menambahkan index unik pada kolom yang ditentukan
ALTER [KOLOM] nama_kolom {SET DEFAULT nilai ATAU DROP DEFAULT}	Menambah atau menghapus nilai default dari kolom
CHANGE [KOLOM] nama_kolom deskripsi_kolom_baru	Mengganti kolom nama_kolom dengan deskripsi baru. Catatan anda bisa mengubah nama kolom dengan perintah ini karena pada deskripsi ada statemen name
MODIFY [KOLOM] deskripsi_kolom	Mirip dengan CHANGE. Tapi tidak dapat digunakan untuk mengubah nama table.
DROP [KOLOM] nama_kolom	Menghapus kolom nama_kolom
DROP PRIMARY KEY	Menghapus primary key pada tabel
DROP INDEX index	Menghapus index
RENAME [AS] nama_tabel_baru	Mengubah nama tabel

Kali ini kita akan coba mengubah struktur dari tabel data\_pribadi. Kita akan menambahkan kolom kota setelah kolom alamat. Sekarang buka mysql prompt ketik query berikut

```
mysql> ALTER TABLE data_pribadi ADD kota varchar(10) AFTER alamat; [ENTER]
```

Untuk melihat hasilnya kita lakukan query SELECT

```
mysql> select * from data_pribadi; [ENTER]
```

Output dari query diatas adalah

```
+-----+-----+-----+-----+-----+-----+
| kode_kar | nama_kar | alamat | kota | jabatan | gaji |
+-----+-----+-----+-----+-----+-----+
| 001A | Web Janarko | Jl. Cyber 19 | NULL | Manajer | 5000000 |
| 002B | Surfi | Jl. Optik 80 | NULL | Sekretaris | 3500000 |
| 005E | Preaky | Jl. Tembaga 911 | NULL | PR | 3000000 |
| 004D | Cracky | Jl. Port 21 | NULL | Office Boy | 750000 |
| 003C | Hacky | Jl. Port 80 | NULL | PR | 4000000 |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
```

Seperti yang anda lihat kolom kota telah disisipkan pada tabel pribadi. Kita akan mengisinya dengan Surabaya.

```
mysql> UPDATE data_pribadi SET kota='Surabaya'; [ENTER]
```

Jika tidak ada kesalahan yang muncul adalah Query OK, 5 rows affected (0.00 sec)  
Rows matched: 5 Changed: 5 Warnings: 0. Lakukan query berikut untuk melihat hasilnya.

```
mysql> SELECT nama_kar, alamat, kota FROM data_pribadi; [ENTER]
```

Outputnya akan terlihat seperti berikut

```
+-----+-----+-----+
| nama_kar | alamat | kota |
+-----+-----+-----+
| Web Janarko | Jl. Cyber 19 | Surabaya |
| Surfi | Jl. Optik 80 | Surabaya |
| Preaky | Jl. Tembaga 911 | Surabaya |
| Cracky | Jl. Port 21 | Surabaya |
| Hacky | Jl. Port 80 | Surabaya |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

## Menghapus Record pada Database

Menghapus record cukup simpel. Anda dapat menggunakan query **DELETE** untuk melakukannya. Struktur dasar penulisan query DELETE adalah sebagai berikut

```
DELETE FROM nama_tabel [WHERE kondisi] [LIMIT jumlah];
```

Tanpa adanya klausua WHERE maka maka semua baris pada tabel tersebut akan dihapus jadi hati-hati. Karena kinerja dari salah satu pegawai kita buruk maka kita akan memecatnya. Wih..., kejam sekali!. Pegawai yang akan kita pecat adalah si Hacky. Yang merupakan PR (public relation). Untuk itu kita akan menghapusnya dari tabel daftar karyawan. Ketik perintah berikut pada mysql\_prompt

```
mysql> DELETE FROM data_pribadi WHERE nama_kar='Hacky'; [ENTER]
```

Jika tidak ada kesalahan pesan yang muncul adalah Query OK, 1 row affected (0.02 sec). Untuk mengeceknya kita dapat menggunakan query SELECT seperti berikut

```
mysql> SELECT nama_kar, kota, jabatan FROM data_pribadi; [ENTER]
```

Output query diatas adalah

```
+-----+-----+-----+
| nama_kar | kota | jabatan |
+-----+-----+-----+
| Web Janarko | Surabaya | Manajer |
| Surfi | Surabaya | Sekretaris |
| Preaky | Surabaya | PR |
| Cracky | Surabaya | Office Boy |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

## Menghapus Tabel

Untuk menghapus tabel langkahnya lebih simpel dari menghapus record. Query yang kita gunakan untuk menghapus tabel adalah DROP. Struktur dasar penggunaan DROP adalah sebagai berikut

```
DROP TABLE nama_tabel;
```

Kita tidak akan mempraktekkan ini karena tabel ini masih kita perlukan pada pembahasan selanjutnya.

## Menghapus Database

Sama dengan menghapus tabel untuk menghapus database kita kita melakukan query DROP. Strukturnya pun sama dengan menghapus tabel yaitu

```
DROP DATABASE nama_database;
```

---

Perintah diatas akan menghapus seluruh baris, seluruh tabel, dan database itu sendiri. Jadi dapat anda bayangkan jika anda keliru melakukan penghapusan. So be careful!.

## Teknik MySQL Lainnya

Kali ini kita akan membahas beberapa teknik lainnya pada MySQL. Diantaranya yang akan kita bahas adalah tabel join, backup database, mengoptimalkan database tipe tabel dan user privilege. Untuk yang pertama kita akan membahas tabel join.

### Tabel Join

Kadang kala anda mungkin memerlukan informasi lebih dari satu tabel. Sebagai contoh anda ingin mengetahui karyawan yang berjabatan sebagai Manajer. Selain itu anda juga ingin melihat tunjangannya. Karena jabatan terletak pada tabel data\_pribadi dan tunjangan terletak pada tabel tunjangan. Maka kita harus melakukan *join* pada kedua tabel untuk melakukan query yang diinginkan.

Karena kita hanya memiliki satu tabel kita harus membuat satu lagi agar dapat melakukan join. Untuk itu bukalah PHP DESIGNER 2006 klik menu **File-New-SQL** untuk membuat file SQL baru lalu ketik kode berikut

```
-- aktifkan database karyawan
USE karyawan;
-- buat tabel tunjangan
CREATE TABLE tunjangan
(
kode_kar varchar(4) not null,
tunjangan mediumint(9),
PRIMARY KEY(kode_kar)
) TYPE='MyISAM';

-- masukkan data pada tabel
INSERT INTO tunjangan VALUES ('001A', 250000), ('002B', 175000),
 ('004D', 175000), ('005E', 100000);
```

Simpan pada direktori hari-11 dengan nama **tunjangan.sql**. Pada kode SQL diatas kita tidak menuliskan perintah INSERT 4x. Perintah itu kita ganti dengan tanda koma yang memisahkan setiap record. Selanjutnya kita akan mengeksekusi file *tunjangan.sql* melalui mysql prompt. Ketik perintah berikut lalu tekan ENTER

```
mysql> source i:/Apache2/htdocs/webmaster/Bab3/hari-11/tunjangan.sql;
```

Jika berhasil pesan yang muncul adalah

```
Database changed
Query OK, 0 rows affected, 1 warning (0.01 sec)

Query OK, 4 rows affected (0.00 sec)
Records: 4 Duplicates: 0 Warnings: 0
```

Jika muncul pesan diatas maka semua query yang kita lakukan pada file *tunjangan.sql* tidak ada masalah. Sekarang kita akan menampilkan record dari tabel tunjangan.

```
mysql> SELECT * FROM tunjangan; [ENTER]
+-----+-----+
| kode_kar | tunjangan |
+-----+-----+
| 001A | 250000 |
| 002B | 175000 |
| 004D | 175000 |
| 005E | 100000 |
```

```
+-----+
4 rows in set (0.00 sec)
```

Untuk melakukan join dari dua tabel. Kedua tabel tersebut harus memiliki kolom yang dapat berhubungan dalam hal ini kode\_kar. Query berikut ini menunjukkan join dari tabel tunjangan ke tabel data\_pribadi. Dimana dalam menampilkan hasil dari tabel tunjangan, kita memilih kriteria dari tabel data\_pribadi bukan dari tabel tunjangan. Untuk lebih memperjelas ketik query berikut pada mysql prompt

```
mysql> SELECT tunjangan.kode_kar, tunjangan.tunjangan [ENTER]
-> FROM tunjangan LEFT JOIN data_pribadi [ENTER]
-> USING (kode_kar) [ENTER]
-> WHERE data_pribadi.nama_kar = "Web Janarko"; [ENTER]
```

Karena query diatas cukup panjang maka lebih nyaman untuk kita menuliakannya dengan cara seperti diatas. Output yang ditampilkan dari query diatas adalah

```
+-----+
| kode_kar | tunjangan |
+-----+-----+
| 001A | 250000 |
+-----+-----+
1 row in set (0.00 sec)
```

Tanda titik pada query diatas menghubungkan antara nama tabel dan kolom. Keyword yang digunakan untuk menghubungkan dua tabel adalah LEFT JOIN. Kemudian kita mentukan kolom mana yang punya relational(hubungan) dengan kolom lain pada tabel berbeda dengan menggunakan USING. Dalam kasus ini yang memiliki hubungan dengan kolom lain adalah kode\_kar. Selanjutnya kita menentukan kriteria yang kita inginkan dengan menggunakan klaus WHERE.

Selain LEFT JOIN juga terdapat beberapa keyword lain. Untuk daftar lengkapnya lihat tabel 3.5

**Tabel 3.5** Daftar tipe join pada MySQL

Keyword	Penjelasan
CARTESIAN PRODUCT	Semua kombinasi pada semua baris dalam semua tabel. Digunakan koma diantara nama tabel. Tidak dapat menggunakan klaus WHERE
FULL JOIN	Sama dengan diatas
CROSS JOIN	Sama dengan diatas. Dapat juga diletakkan diantara nama tabel yang <i>dijoinkan</i> .
INNER JOIN	Hampir sama dengan pemisahan dua tabel dengan koma. Jika tanpa WHERE maka sama dengan FULL JOIN.
EQUI JOIN	Digunakan pada kondisi yang menggunakan = untuk mencocokkan tabel. Dapat digunakan dengan klaus WHERE.
LEFT JOIN	Digunakan untuk mencocokkan baris antar tabel dan baris yang tidak cocok diisi dengan NULL. Digunakan dengan WHERE.

## Membackup Database

Untuk berjaga-jaga jika terjadi kerusakan atau database tidak sengaja terhapus. Maka secara periodik kita perlu melakukan backup. Untuk membackup database kita dapat menggunakan program kecil yaitu mysqldump. Jika saat ini anda masih pada mysql propmt maka ketikkan \q [ENTER] untuk keluar dari mysql prompt. Pada DOS prompt ketikkan

```
C:\>mysqldump -u root -p --compact -B karyawan > c:/Apache2/htdocs/
webmaster/Bab3/hari-11/backup_karyawan.sql [ENTER]
```

```
Enter password: ***** [ENTER]
```

Pada perintah mysqldump kita membackup database dan memasukkannya pada file *backup\_karyawan.sql*. File tersebut kita letakkan pada direktori hari-11. Untuk mengetahui pilihan-pilihan lainnya dari mysqldump pada DOS prompt anda dapat mengetikkan perintah berikut

```
C:\>mysqldump --help | more [ENTER]
```

## Mengoptimalkan Tabel

Jika tabel sering digunakan seperti diupdate, dihapus dan sebagainya. Maka tabel tersebut menjadi fragment atau terpecah-pecah. Jika tabel itu sudah fragment maka pengaksesan pun menjadi lambat. Oleh karena itu kita harus mengoptimalkan kembali tabel tersebut. Untuk keperluan tersebut kita dapat menggunakan query `OPTIMIZE TABLE`.

Contoh berikut ini menunjukkan pengoptimalan pada tabel *data\_pribadi* dan *tunjangan*. Pada mysql prompt ketik query berikut

```
mysql> OPTIMIZE TABLE data_pribadi, tunjangan; [ENTER]
```

```
+-----+-----+-----+-----+
| Table | Op | Msg_type | Msg_text |
+-----+-----+-----+-----+
| karyawan.data_pribadi | optimize | status | OK |
| karyawan.tunjangan | optimize | status | OK |
+-----+-----+-----+-----+
2 rows in set (0.02 sec)
```

## Tipe Tabel pada MySQL

Ingat pada saat anda membuat tabel kita meyisipkan perintah `TYPE='MyISAM'`. Itu berarti tabel yang kita buat bertipe MyISAM. Selain MyISAM ada beberapa tipe tabel lagi. Lihat tabel 3.6 untuk daftar lengkapnya.

**Tabel 3.6** Daftar tipe tabel pada MySQL

Tipe	Penjelasan
MyISAM	Tipe ini berdasarkan ISAM, yang merupakan singkatan dari <i>Indexed Sequential Access Method</i> . Tipe ini yang akan banyak kita gunakan. Ini merupakan metode standar untuk menyimpan record dan file.
ISAM	Seperti yang dijelaskan diatas
HEAP	Tabel tipe ini disimpan pada memori komputer. Ini yang membuat pengaksesan HEAP sangat cepat dibanding tipe lainnya. Tipe ini cocok untuk menyimpan data yang sifatnya sementara. Namun pada saat membuat tabel ini anda harus menentukan nilai <code>MAX ROWS</code> pada query <code>CREATE TABLE</code> . Tipe ini juga tidak dapat menampung kolom dengan tipe <code>TEXT</code> , <code>BLOB</code> dan <code>AUTO_INCREMENT</code> .
BDB	Tabel jenis ini cocok digunakan untuk transaksi. Karena tabel ini memiliki fasilitas <code>COMMIT</code> dan <code>ROLLBACK</code> . Jenis ini lebih lambat daripada MyISAM namun memiliki banyak kelebihan jika digunakan pada transaksi.
InnoDB	Tabel tipe ini merupakan standar yang digunakan pada MySQL versi 4.x dan 5.x. Jenis ini sama dengan BDB.

**AUTO\_INCREMENT** merupakan tipe kolom yang nilainya otomatis bertambah satu jika ada penambahan record baru.

## User Privilege

Saat mengakses MySQL lewat PHP sangatlah risikan jika kita menggunakan root sebagai user. Karena root memiliki semua hak (privilege) dalam memanipulasi database. Penggunaan root hanya diperlukan untuk kegiatan administrasi saja. Dengan demikian kita perlu membuat user baru yang hanya memiliki hak atau privilege pada query-query tertentu saja.

Untuk membuat privilege anda harus login terlebih dahulu sebagai root. Hal ini sudah kita lakukan. Query yang kita gunakan untuk keperluan privilege adalah GRANT. Struktur dasar dari query GRANT adalah sebagai berikut.

```
GRANT privilege [KOLOM] ON database TO username [IDENTIFIED BY 'password']
[WITH GRANT OPTION];
```

Nah karena saat ini privilege dari database karyawan hanya root yang memegangnya. Kita akan membuat user baru yang bernama *satpam*. User ini akan kita beri privilege SELECT, UPDATE, INSERT dan DELETE. User satpam ini nantinya hanya punya wewenang pada database karyawan. Dan hanya boleh mengeksekusi keempat query seperti yang sudah disebutkan sebelumnya. Pada mysql prompt ketikkan query berikut

```
mysql> GRANT SELECT, INSERT, UPDATE, DELETE [ENTER]
-> ON karyawan.* TO satpam [ENTER]
-> IDENTIFIED BY "satpam123"; [ENTER]
```

Jika berhasil pesan yang ditampilkan adalah *Query OK, 0 rows affected (0.00 sec)*. Password dari user satpam adalah *satpam123*. Tentu saja anda harus mengganti password yang lebih baik jika pada aplikasi sebenarnya. Karena saat ini anda masih login sebagai root anda masih bisa mengakses database mysql. Untuk melihat daftar user yang ada, sekarang ketikkan query berikut

```
mysql> use mysql; [ENTER]
mysql> SELECT user, password FROM user; [ENTER]
```

Output yang ditampilkan adalah seperti berikut

user	password
root	*AF74DC9E71AF0EE54053466BB3BBE00336A5B4C3
root	*AF74DC9E71AF0EE54053466BB3BBE00336A5B4C3
satpam	*E44D241271A19F90FBC6DE8FA5BC6EDF596BD8EC

3 rows in set (0.00 sec)

User satpam telah berhasil kita masukkan pada daftar user. Tapi... passwordnya kok berbeda? Bukan *satpam123*. Untuk keamanan password telah dienkrip menggunakan fungsi *PASSWORD()*. Namun untuk login anda tetap menggunakan password *satpam123*. Jadi saat login otomatis password yang anda ketikkan anda dienkrip menggunakan fungsi *PASSWORD()* sehingga hasilnya akan selalu sama.

Untuk mengetahui apakah benar user satpam hanya memiliki privilege pada database karyawan dan query yang diperbolehkan hanya SELECT, UPDATE, INSERT dan DELETE. Keluarlah dari mysql prompt dengan menggunakan perintah *\q* [ENTER]. Setelah anda berada pada DOS prompt, login kembali ke MySQL Monitor menggunakan user satpam seperti berikut

```
C:\>mysql -u satpam -p [ENTER]
```

Pada Enter Password ketikkan *Satpam123*  
Enter password: \*\*\*\*\* [ENTER]

---

Sekarang anda sudah login tetapi bukan sebagai root melainkan sebagai satpam. Kita akan coba mengakses database mysql yang bukan hak dari user ini. Pada mysql prompt ketik query berikut

```
mysql> use mysql; [ENTER]
ERROR 1044 (42000): Access denied for user 'satpam'@'%' to database 'mysql'
```

Error tersebut menunjukkan kita tak mempunyai privilege untuk mengakses database mysql. Sekarang cobalah untuk mengakses database karyawan pada mysql prompt ketikkan

```
mysql> use karyawan; [ENTER]
Database changed
```

Ternyata memang bisa. Namun untuk melihat apakah semua query dapat dijalankan pada database ini lakukan query DROP TABLE. Yang bener nih...! ya lakukan saja

```
mysql> DROP TABLE data_pribadi; [ENTER]
ERROR 1142 (42000): DROP command denied to user 'satpam'@'localhost' for table
'data_pribadi'
```

Betulkan! User satpam tidak memiliki privilege terhadap query DROP sehingga MySQL tidak mengijinkan penggunaan query ini.

\*\*\*\*\*

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

Ikuti langkah-langkah berikut sebelum melanjutkan materi pada hari ke dua belas

- Masuklah pada direktori C:\Apache2\htdocs\webmaster\Bab3\
- Buat folder baru dan beri nama **hari-12**

## phpMyAdmin

Karena pada hari sebelumnya anda telah menguasai MySQL. Maka anda tidak akan kesulitan dalam menggunakan phpMyAdmin. Karena phpMyAdmin hanyalah alat untuk mempermudah manipulasi database. Untuk itu materi yang harus anda kuasi kali ini meliputi:

- Mengenal phpMyAdmin
- Mengenal Berbagai Tab phpMyAdmin

### Mengenal phpMyAdmin

Seperti yang pernah dijelaskan pada awal buku ini yaitu pada pendahuluan. phpMyAdmin merupakan aplikasi yang dibangun menggunakan script PHP dan digunakan untuk administrasi database pada MySQL. Jika pada hari sebelumnya anda berinteraksi dengan database MySQL menggunakan MySQL Monitor. Kali ini kita menggunakan media Web Browser seperti IE untuk keperluan administrasi database.

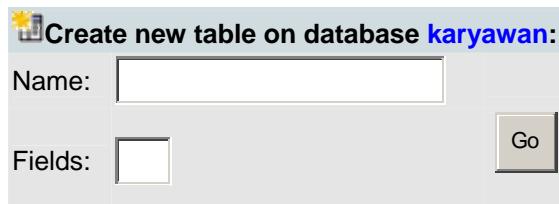


Gambar 3.1: Tampilan utama phpMyAdmin

Untuk mengakses phpMyAdmin buka browser anda dan arahkan ke <http://localhost/webmaster/phpMyAdmin-2.x.x/>. Dimana x adalah versi yang anda gunakan sekarang. Penulis sendiri menggunakan phpMyAdmin versi 2.6.1. phpMyAdmin sering disertakan pada fasilitas web hosting untuk melakukan administrasi database. Tampilan utama dari phpMyAdmin dapat anda lihat pada gambar 3.1.

Tampilan utama dari phpMyAdmin dibagi dalam dua frame utama yaitu frame kiri dan frame kanan. Pada frame kiri kita dapat memilih database yang akan diedit dan menampilkan seluruh tabel yang ada pada database tersebut. Pada frame sebelah kanan kita dapat membuat database baru, mengganti bahasa, melihat privilege dan sebagainya.

Sekarang coba klik pada karyawan pada pilihan daftar database. Anda dapat membuat tabel baru pada box Create new table on database karyawan. Fields merupakan jumlah kolom yang akan dibuat. Setelah itu anda dapat mengklik tombol Go. Lihat gambar 3.2 untuk lebih jelas



Gambar 3.2: membuat tabel baru

## Mengenal Berbagai Tab phpMyAdmin

Jika yang aktif hanya database dan anda belum memilih tabel maka tab yang ada hanya tujuh tab. Ketujuh tab tersebut diantaranya Structure, SQL, Export, Search, Query, Operation dan yang terakhir Drop. Sedangkan jika ada tabel yang anda pilih maka jumlah tab yang ada adalah sembilan tab. Yang akan kita bahas adalah kesembilan tab ini.

### Tab Structure

Untuk mengaktifkan kesembilan tab tersebut coba klik tabel data\_pribadi. Link data\_pribadi terletak dibawah pilihan daftar database. Klik tabel data\_pribadi maka akan muncul kolom-kolom yang pernah kita buat yaitu kode\_kar, nama\_kar dan sebagainya. Lihat gambar 3.3.

Pada tab **structure** anda dapat mengubah struktur dari tabel. Seperti mengubah primary key, menghapus kolom dan sebagainya. Untuk mengedit struktur anda dapat mengklik gambar pensil . Sedangkan untuk menghapus anda dapat mengklik gambar silang . Gambar kunci dan masing-masing untuk membuat primary key dan index. Gambar digunakan untuk memberikan nilai unik pada kolom. Yang terakhir adalah gambar untuk menambahkan FULLTEXT pada kolom.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
kode_kar	varchar(4)	latin1_swedish_ci		Yes			
nama_kar	varchar(30)	latin1_swedish_ci		Yes	NULL		
alamat	varchar(50)	latin1_swedish_ci		Yes	NULL		
jabatan	varchar(15)	latin1_swedish_ci		Yes	NULL		
gaji	mediumint(10)			Yes	NULL		

Check All / Uncheck All With selected:

Add 1 field(s) At End of Table At Beginning of Table After kode\_kar

Indexes:				Space usage:		Row Statistic:		
Keyname	Type	Cardinality	Action	Field	Type	Usage	Statements	V
PRIMARY	PRIMARY	0		kode_kar	Data	0 Bytes	Format	dynamic
Create an index on	columns	Go			Index	1,024 Bytes	Collation	latin1_sw

Gambar 3.3: Struktur dari tabel data\_pribadi

Jika anda terus scroll ke bawah anda akan menemukan **Run SQL query/queries** yaitu textarea untuk melakukan query seperti pada mysql prompt. Lihat Gambar 3.4.



Gambar 3.4: Query box dan import file pada tab struktur

Pada bagian ini anda dapat melakukan query seperti layaknya di mysql prompt. Sedangkan dibawah query box anda dapat mengimport file SQL yang telah ada. Untuk percobaan coba ketikkan query berikut pada query box lalu klik Go.

```
SELECT * FROM data_pribadi ORDER BY gaji DESC;
Output dari query diatas ditunjukkan seperti gambar 3.5
```

Showing rows 0 - 3 (4 total, Query took 0.0016 sec)						
SQL-query:						
<pre>SELECT * FROM data_pribadi ORDER BY gaji DESC LIMIT 0 , 30</pre>						
<a href="#">[Edit]</a> <a href="#">[Explain SQL]</a> <a href="#">[Create PHP Code]</a> <a href="#">[Refresh]</a>						
Show :	30	row(s) starting from record # 0				
in	horizontal	mode and repeat headers after 100 cells				
Sort by key:	None	<a href="#">[Go]</a>				
	<a href="#">←</a>	<a href="#">→</a>	kode_kar	nama_kar	alamat	kota
	<a href="#">📝</a>	<a href="#">✖</a>	001A	Web Janarko	Jl. Cyber 19	Surabaya
	<a href="#">📝</a>	<a href="#">✖</a>	002B	Surfi	Jl. Optik 80	Surabaya
	<a href="#">📝</a>	<a href="#">✖</a>	005E	Preaky	Jl. Tembaga 911	Sekretaris
	<a href="#">📝</a>	<a href="#">✖</a>	004D	Cracky	Jl. Port 21	PR
	<a href="#">↑</a>	<a href="#">Check All</a> / <a href="#">Uncheck All</a>		<a href="#">With selected:</a> <a href="#">📝</a> <a href="#">✖</a> <a href="#">[Import]</a>		

Gambar 3.5: hasil query dari query box

Hasil query langsung ditampilkan secara descending berdasarkan gaji. Perintah query kita juga ditampilkan ulang pada bagian atas. Query ini secara otomatis ditambahkan klausa LIMIT 0, 30 oleh phpMyAdmin.

## Tab Browse

Pada tab ini seluruh data pada tabel akan ditampilkan. Query otomatis yang digunakan untuk menampilkan adalah

```
SELECT * FROM data_pribadi LIMIT 0, 30;
```

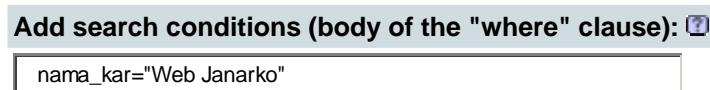
Hasil yang ditampilkan mirip dengan gambar 3.5 pada bahasan sebelumnya.

## Tab SQL

Sebenarnya tab ini sama dengan apa yang ada pada tab structure seperti yang ditunjukkan oleh gambar 3.4. Jadi untuk tab ini saya tidak perlu menjelaskan ulang.

## Tab Search

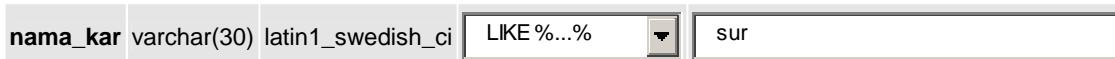
Pada tab ini anda dapat mencari suatu data yang ada pada tabel. Pertama anda harus memilih minimal satu kolom pada list Select fields (at least one). Terdapat pilihan hasil pencarian ditampilkan secara ascending atau descending pada **Display Order**.



Add search conditions (body of the "where" clause):

Gambar 3.6: Textbox pencarian kondisi

Pada textbox seperti yang ditunjukkan gambar 3.6 coba ketikkan `nama_kar="Web Janarko"` lalu klik Go. Apa yang anda ketikkan pada textbox tersebut akan dianggap sebuah kondisi setelah klausula **WHERE**. Cara yang kedua adalah dengan memilih operator yang kita inginkan pada kolom tertentu. Lihat gambar 3.7



nama\_kar varchar(30) latin1\_swedish\_ci  sur

Gambar 3.7 memilih kolom dan operator

Pada gambar diatas kita memilih kolom kota, operator yang kita gunakan adalah `LIKE %...%`. Dan pada **value** kita mengisikan sur. Operator `LIKE %...%` akan cocok dengan segala kata yang memiliki unsur "sur" didalamnya. Jadi query lengkapnya adalah sebagai berikut

```
SELECT * FROM data_pribadi WHERE kota LIKE "%sur%";
```

Sehingga query ini cocok dengan keempat record yang ada pada tabel data\_pribadi. Karena keempat record tersebut memiliki kata *Surabaya* yang berunsur *sur*.

## Tab Insert

Kita dapat menyisipkan record baru pada tab ini. Struktur kolom pada saat penyisipan sama dengan apa yang pernah kita buat. Karena dulu kita pernah memecat seorang pegawai yang berjabatan sebagai PR. Maka kita perlu orang baru untuk menggantikannya. Data-data yang kita masukkan ditunjukkan gambar 3.8



Field	Type	Function	Null	Value
kode_kar	varchar(4)		□	006F
nama_kar	varchar(30)		□	Porty
alamat	varchar(50)		□	Jl. Susup 20
kota	varchar(10)		□	Surabaya
jabatan	varchar(15)		□	PR
gaji	mediumint(10)		□	3000000

Gambar 3.8: Menambahkan record baru

Setelah selesai klik tombol Go. Maka pada record baru tersebut akan dimasukkan pada tabel data\_pribadi. Jika anda masih belum melihat daftar perubahan record baru klik tab **browse**. Secara query SQL kita dapat memasukkan data diatas dengan menuliskan

```
INSERT INTO data_pribadi VALUES ("006F", "Porty", "Jl. Susup 20", "Surabaya", "PR", 3000000);
```

## Tab Operations

Tab operations ini lebih mengarah ke arah maintenance (pemeliharaan) dan perubahan struktur database dan tabel. Disebelah kanan terdapat box kecil berjudul **Table Maintenance**. Lihat gambar 3.9



Gambar 3.9

Seperti yang anda pada gambar 3.9 kita dapat melakukan optimalisasi tabel, FLUSH tabel, memperbaiki tabel, mengecek tabel dan menganalisa tabel. Selain melakukan maintenance kita juga dapat mengganti struktur tabel dari MyISAM ke InnoDB dan sebagainya.

## Tab Empty

Jika anda mengklik tab ini maka akan muncul window dialog peringatan seperti gambar 3.10



Gambar 3.10

Jika anda mengklik tombol OK maka isi dari tabel data\_pribadi akan dikosongkan. Dan hanya itulah tampilan tab *empty*.

## Tab Drop

Hampir sama dengan tab empty. Tab drop akan memunculkan dialog peringatan seperti gambar 3.11



Gambar 3.11

---

Jika anda mengkil tombol OK maka tabel data\_pribadi akan dihapus.

\* \* \* \* \*

# Mengakses MySQL Menggunakan PHP

Pada bab-bab sebelumnya kita menyimpan data menggunakan file biasa. Kita akan mengubah kebiasaan itu dengan mengganti metode penyimpanan kita dari file biasa ke database (MySQL). Dengan menggunakan sistem database pengaksesan data pada aplikasi web kita menjadi lebih cepat, mudah dan aman. Untuk pembahasan ini materi yang harus anda kuasai antara lain:

- Langkah-langkah Mengakses Database Lewat PHP
- Melakukan Koneksi ke MySQL
- Memilih Database
- Melakukan Query pada Database
- Menampilkan Hasil dari Query
- Fungsi-fungsi lain

Database yang akan kita gunakan masih tetap sama yaitu database *karyawan* yang pernah kita buat sebelumnya.

## Langkah-langkah Mengakses Database Lewat PHP

Sebelum kita dapat melakukan manipulasi database, ada beberapa langkah yang diperlukan untuk keperluan tersebut. Langkah-langkah tersebut antara lain:

- Melakukan koneksi ke MySQL menggunakan `mysql_connect()`
- Memilih database menggunakan `mysql_select_db()`
- Melakukan Query pada database menggunakan `mysql_query()`
- Menampilkan Hasil dari Query menggunakan `mysql_fetch_array()`
- Menutup koneksi menggunakan `mysql_close()`

Langkah-langkah diatas merupakan materi fundamental yang harus anda kuasai dalam menghubungkan MySQL dengan PHP.

## Melakukan Koneksi ke MySQL

Untuk melakukan koneksi ke MySQL kita dapat menggunakan fungsi `mysql_connect()`. Fungsi ini memiliki struktur dasar sebagai berikut

```
mysql_connect ([string server], [string username], [string password], [boolean link baru], [integer flag]);
```

Kesemua parameter fungsi tersebut bersifat opsional artinya boleh ditulis boleh tidak. Namun untuk tiga paramater pertama yaitu server, username dan password sebaiknya kita tulis. Jika anda tidak menulis tiga parameter utama tersebut kemungkinan anda tidak dapat login ke MySQL. Ini dikarenakan jika tidak ditulis MySQL akan menggunakan localhost default, untuk username yang digunakan adalah username yang diproses PHP saat itu. Dan untuk password kosong.

Untuk memberi tanda pengenal pada koneksi kita maka buatlah variabel untuk menampung fungsi `mysql_connect()`. Contoh berikut menunjukkan kita koneksi ke MySQL menggunakan username *satpam* dan password *satpam123* yang pernah kita buat untuk database *karyawan*.

```
$koneksi = mysql_connect("localhost", "satpam", "satpam123");
```

Variabel yang kita gunakan sebagai tanda pengenal adalah `$koneksi`. Tanda pengenal untuk koneksi kita perlukan untuk menghandel adanya error saat melakukan koneksi. Ada perintah

---

yang hampir mirip dalam melakukan koneksi ke MySQL yaitu dengan menggunakan fungsi `mysql_pconnect()`. Kode diatas dapat kita ganti menjadi seperti berikut

```
$koneksi = mysql_pconnect("localhost", "satpam", "satpam123");
```

Perbedaan antara `mysql_connect()` dan `mysql_pconnect()` adalah `mysql_pconnect()` menggunakan koneksi *persistent*. Artinya sebelum membuat koneksi, ia akan mengecek apakah sudah ada koneksi yang sudah terbuka. Jika ya maka koneksi itu yang digunakan bukan membuat koneksi baru. Hal ini tentunya menghemat waktu dan *overhead server*.

## Memilih Database

Anda tentu ingat pada mysql prompt sebelum melakukan manipulasi pada database kita harus aktifkan dulu databasenya dengan perintah berikut

```
mysql> use karyawan;
```

Pada PHP pun berlaku demikian. Sebelum melakukan query kita dapat mengaktifkan database menggunakan fungsi `mysql_select_db()`. Struktur dasar penulisan fungsi ini adalah

```
mysql_select_db(string database, [koneksi pengenal]);
```

Parameter kedua yaitu koneksi pengenal bersifat opsional. Jika tidak ditulis akan digunakan koneksi yang aktif saat ini. Karena yang akan kita pakai adalah database *karyawan* maka perintahnya adalah

```
mysql_select_db("karyawan");
```

## Melakukan Query pada Database

Untuk melakukan query pada database fungsi yang dapat digunakan adalah `mysql_query()`. Struktur dasar dari fungsi `mysql_query()` adalah sebagai berikut

```
mysql_query(string query, [koneksi pengenal]);
```

Jika kita memasukkan string query kedalam fungsi `mysql_query()` secara langsung hal itu akan kurang baik jika query yang kita berikan sangat panjang. Cara yang lebih baik dan umum digunakan adalah menyimpan string query pada suatu variabel. Sehingga variabel inilah yang akan kita masukkan pada parameter fungsi. Lihat contoh berikut

```
$query = "SELECT * FROM data_pribadi ORDER BY gaji DESC";
$hasil = mysql_query($query);
```

Pada kode diatas kita membuat variabel `$query` yang isinya query dari tabel `data_pribadi`. Lalu kita membuat variabel `$hasil` yang isinya adalah nilai dari fungsi `mysql_query($query)`. Dengan demikian jika ada kesalahan kita dapat dengan mudah mengatasinya.

## Menampilkan Hasil dari Query

Banyak fungsi yang tersedia untuk mengolah hasil dari query yang telah diterima. Yang paling sering kita gunakan nantinya adalah `mysql_num_rows()` untuk menghitung jumlah baris yang dikembalikan oleh query. Dan `mysql_fetch_array()` untuk memecah setiap record pada baris menjadi array. Array yang dihasilkan dari fungsi ini dapat bertipe string maupun numerik.

Kedua fungsi yang telah disebutkan sebelumnya masing-masing hanya memiliki satu parameter yaitu hasil query dari fungsi `mysql_query()`. Jadi jika kita ingin mengetahui jumlah baris yang dikembalikan maka kita dapat menuliskannya seperti berikut

---

```
$jml_baris = mysql_num_rows($hasil);
```

Mengetahui jumlah baris diperlukan jika kita akan menampilkan data menggunakan looping for. Dengan demikian kita akan tahu berapa kali looping tersebut dilakukan. Didalam looping baru kita menggunakan fungsi mysql\_fetch\_array() untuk memecah setiap record pada baris yang aktif saat looping. Kode di bawah ini menunjukkan penggunaan looping for untuk menampilkan data.

```
for ($baris=0; $baris<$jml_baris; $baris++)
{
 $data = mysql_fetch_array($hasil);
 //proses data
}
```

Pada kode diatas variabel \$data otomatis menjadi array dari setiap kolom dari tabel data\_pribadi. Jadi \$data[0] atau \$data['kode\_kar'] mewakili data yang ada pada kolom kode\_kar pada baris yang aktif saat looping. Dan demikian seterusnya hingga \$data[5] atau \$data['gaji'] .

Alternatif lain adalah menggunakan looping while. Pada looping ini kita tidak harus mengetahui jumlah baris yang dikembalikan. Karena while akan terus mengulang blok kode hingga tercapainya elemen terakhir. Penulisan looping for diatas dapat kita ganti dengan kode berikut

```
while ($data = mysql_fetch_array($hasil))
{
 //proses data
}
```

Arti dari kedua looping ini sama persis. Namun nampak penggunaan while lebih singkat dibanding for untuk kasus ini.

Selain fungsi mysql\_fetch\_array() ada beberapa fungsi lain yang dapat digunakan untuk memecah setiap data. Fungsi-fungsi tersebut diantaranya mysql\_fetch\_row() dan mysql\_fetch\_object().

Fungsi mysql\_fetch\_row() hampir sama dengan mysql\_fetch\_array() hanya saja pada mysql\_fetch\_row() index array hanya bertipe numerik. Sedangkan mysql\_fetch\_object() akan membuat object berdasarkan baris pada setiap record. Jadi untuk mengakses setiap atributnya anda dapat menggunakan

\$data->kode\_kar, \$data->nama\_kar dan seterusnya. Jika sebelumnya \$data telah anda definisikan sebagai berikut

```
$data = mysql_fetch_object($hasil);
```

## Menutup Koneksi

Untuk menutup suatu koneksi kita cukup menggunakan fungsi

```
mysql_close();
```

Sebenarnya menutup koneksi tidak mutlak harus dilakukan. Hal ini dikarenakan koneksi akan secara otomatis ditutup ketika script selesai dieksekusi.

## Fungsi-fungsi Lain

Selain berberapa fungsi yang kita gunakan untuk langkah-langkah mengakses database. Beberapa fungsi berikut mungkin berguna bagi anda

### **mysql\_create\_db()**

---

Untuk membuat database anda dapat mempergunakan fungsi ini. Fungsi ini memiliki struktur sebagai berikut

```
mysql_create_db(string nama_database, [koneksi pengenal]);
```

Parameter kedua yang berada pada kurung siku berarti bersifat opsional

### **mysql\_drop\_db()**

Seperti terlihat seperti namanya *drop* maka sudah jelas fungsi ini untuk menghapus database. Struktur dasar penulisannya pun sama seperti diatas yaitu

```
mysql_drop_db(string nama_database, [koneksi pengenal]);
```

## **PENERAPAN TEORI MENGAKSES MYSQL MENGGUNAKAN PHP**

Untuk penerapan teori ini kita akan mencoba menampilkan record yang ada pada tabel data\_pribadi dan memasukkan record ke tabel data\_pribadi tersebut. Untuk itu kita akan membuat dua file. Satu untuk menampilkan record yang ada pada tabel data\_pribadi. Dan yang satu lagi untuk memasukkan record baru pada tabel data\_pribadi. Untuk itu jalankan PHP\_DESIGNER 2005 klik menu **File-New-PHP** untuk membuat file lalu ketik kode berikut

```
<?php

//lakukan koneksi ke MySQL
@ $koneksi = mysql_connect("localhost", "satpam", "satpam123");
if (!$koneksi)
 exit('Tidak bisa koneksi ke MySQL');
// pilih database karyawan
mysql_select_db("karyawan");
// lakukan query pada database
$query = mysql_query("SELECT * FROM data_pribadi ORDER BY gaji DESC");
// hitung jumlah baris yang dikembalikan
$jml_baris = mysql_num_rows($query);
// tutup koneksi
mysql_close();
// buat tabel
echo '<h2>Database karyawan - tabel data_pribadi</h2>';
.'<table border="1" cellpadding="3" width="100%">
.'<tr bgcolor="#cccccc">
.'<th>Kode</th><th>Nama</th><th>Alamat</th><th>Kota</th>
.'<th>Jabatan</th><th>Gaji</th></tr>';
// tampilan hasil menggunakan looping for
for ($baris=0; $baris<$jml_baris; $baris++)
{
 $data = mysql_fetch_array($query);
 echo '<tr>';
 for ($kolom=0; $kolom<6; $kolom++)
 {
 echo '<td>'.$data[$kolom].'</td>';
 }
 echo '</tr>';
}
echo '</table>';
.'<p>Masukkan record baru</p>';
?>
```

Ingin kita masih pada hari ke-12. Jadi simpan pada direktori **hari-12** dengan nama **lihat\_db.php**. Jalankan pada browser anda arahkan alamatnya ke [http://localhost/webmaster/Bab3/hari-12/lihat\\_db.php](http://localhost/webmaster/Bab3/hari-12/lihat_db.php). output dari script diatas akan terlihat seperti gambar 3.12

Gambar 3.12: output file lihat\_db.php

## PENJELASAN SCRIPT

Kita memberi tanda @ didepan variabel \$koneksi agar error yang terjadi karena kegagalan koneksi dapat kita handel sepenuhnya. Tanpa tanda @ memang benar pesan error kita tetap muncul tetapi sebelumnya PHP memberikan pesan peringatan file dan baris mana yang error. Untuk alasan keamanan maka kita beri tanda @ agar pesan error dari PHP tidak dicetak ke layar.

Selanjutnya kita memilih database yang digunakan dengan menggunakan fungsi mysql\_select\_db("karyawan"). Setelah ada database yang dipilih baru kita dapat melakukan query. Query yang kita gunakan adalah

```
$query = mysql_query("SELECT * FROM data_pribadi ORDER BY gaji DESC");
```

Artinya kita menampilkan data pada tabel data\_pribadi diurut secara descending menurut gaji. Jadi data ditampilkan mulai dari record yang memiliki gaji terbesar hingga terkecil. Setelah itu kita membuat variabel \$jml\_baris yang nilainya merupakan jumlah baris yang dihasilkan query SQL. Baru setelah itu kita tutup koneksi menggunakan mysql\_close().

```
for ($baris=0; $baris<$jml_baris; $baris++)
```

Untuk menampilkan seluruh record kita menggunakan looping for seperti diatas. Kode yang ada pada blok ini akan diulang sebanyak \$jml\_baris. Pada blok for ini kita memecah setiap kolom yang ada pada baris yang aktif menggunakan mysql\_fetch\_array(\$query) dan hasilnya kita masukkan pada variabel \$data. Karena kita tahu jumlah kolom dari tabel ini adalah 6. Maka dari pada harus menulis <td>...</td> enam kali berurut-turut kita gunakan looping for seperti berikut

```
for ($kolom=0; $kolom<6; $kolom++)
```

Pada blok looping diatas kita menuliskan array \$data. Karena \$data juga memiliki index numerik maka kita tulis \$data[\$kolom]. Sehingga nilai dari data nantinya \$data[0], \$data[1],... sampai \$data[5]. \$data[0] sama dengan isi dari kolom kode\_kar pada baris yang aktif saat eksekusi, \$data[1] sama dengan nam\_kar dan seterusnya.

File berikutnya yang akan kita buat adalah file **insert\_db.php**. Fungsi file ini adalah untuk menampilkan form insert dan memprosesnya. Klik **File-New-PHP** pada PHP DESIGNER 2006 lalu ketik kode berikut

Script dari file **insert\_db.php**

```
<?php

//dapatkan variabel data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form_insert';

switch ($proses)
{
 case 'form_insert':
?>

<h2>Memasukkan record baru ke tabel data_pribadi</h2><hr>
<form action="insert_db.php?proses=proses_insert" method="post">
<table border="0" cellpadding="3">
 <tr><td>Kode: </td>
 <td><input type="text" name="kode" maxlength="4" size="4"></td></tr>
 <tr><td>Nama: </td>
 <td><input type="text" name="nama" size="30"></td></tr>
 <tr><td>Alamat: </td>
 <td><input type="text" name="alamat" size="40"></td></tr>
```

```

<tr><td>Kota: </td>
 <td><input type="text" name="kota" size="15"></td></tr>
<tr><td>Jabatan: </td>
 <td><input type="text" name="jabatan" size="15"></td></tr>
<tr><td>Gaji: </td>
 <td><input type="text" name="gaji" size="15" maxlength="10"></td></tr>
<tr><td colspan="2"><input type="submit" value="Masukkan Data"></td></tr>
</table>
</form>
Lihat database

<?php
break; //akhir dari proses form_insert

case 'proses_insert':
//buat variabel dari data form insert
$kode = $_POST['kode'];
$nama = $_POST['nama'];
$alamat = $_POST['alamat'];
$kota = $_POST['kota'];
$jabatan = $_POST['jabatan'];
$gaji = $_POST['gaji'];

//cek apakah ada field yang kosong
foreach ($_POST as $field)
{
 if ($field == '' || !isset($field)) // || sama dengan or
 exit ('Error: Ada field yang masih kosong');
}

//lakukan koneksi ke MySQL
@$koneksi = mysql_connect('localhost', 'satpam', 'satpam123');
if (!$koneksi)
 exit('Error: Tidak bisa koneksi ke MySQL server');

mysql_select_db('karyawan'); //pilih database karyawan
//lakukan query
$query = "INSERT INTO data_pribadi VALUES ('$kode', '$nama', '$alamat',
'$kota', '$jabatan', '$gaji')";
@$hasil = mysql_query($query);
mysql_close(); //tutup koneksi ke MySQL
//tampilkan pesan hasil dari query
if ($hasil)
 echo mysql_affected_rows().' record berhasil dimasukkan ke database'
 .'<p>Lihat database</p>';
else
 echo 'Error: Tidak dapat memasukkan record baru ke database.
'
 .'Kemungkinan query salah atau ada duplikat primary key.';

break; //akhir dari proses proses_insert
}
?>

```

Simpan dengan nama ***insert\_db.php***. Jalankan pada browser anda hasilnya akan seperti gambar 3.13.

Gambar 3.13: tampilan file *insert\_db.php*

### PENJELASAN SCRIPT

Karena file ini dual-fungsi maka kita perlu menentukan aksi yang akan diproses. Kita memasukkan nilai dari proses pada URL ke dalam variabel **\$proses**. Dan jika proses tidak ada

---

atau kosong maka yang akan dijadikan proses default adalah proses form insert. Seperti biasa untuk menghandel proses yang akan dieksekusi kita gunakan statemen **switch()** dan **case**.

Untuk proses form\_insert tidak ada yang istimewa karena tujuan proses ini hanya menampilkan form yang isinya kemudian akan diproses oleh file

**insert\_db.php?proses=proses\_insert**. Titik berat penjelasan saya kali hanya meliputi kode pada proses\_insert.

Pertama kita membuat variabel dari data yang kita terima dari form insert lewat metode POST. Sebelum memasukkan data pada query kita mengecek dahulu apakah masih ada field yang kosong. Karena variabel global `$_POST` merupakan array maka kita dapat menggunakan `foreach` untuk mengecek isinya seperti berikut

```
foreach ($_POST as $field)
```

Jadi `$field` merupakan seluruh elemen array yang didefinisikan sebelumnya. Jika isi dari `$field` kosong atau tidak diset maka keluar dari rutin script. Selanjutnya kita melakukan koneksi ke MySQL dan memilih database. Karena kita akan memasukkan record baru maka query yang kita gunakan adalah `INSERT`. Lihat cuplikan script berikut

```
$query = "INSERT INTO data_pribadi VALUES ('$kode', '$nama', '$alamat', '$kota', '$jabatan', '$gaji')";
```

Hasil dari query kita masukkan pada variabel `$hasil`. Pengenal akan keberhasilan query telah kita buat yakni pada `$hasil`. Jadi kita akan menampilkan berhasil atau tidak berdasar nilai dari `$hasil`. Jika true maka query berhasil dieksekusi dan jika nilainya false maka ada error.

```
if ($hasil)
 echo mysql_affected_rows().' record berhasil dimasukkan ke database'
....
```

Pada cuplikan script diatas jika `$hasil` true maka eksekusi kode berikut. Ada fungsi yang nampaknya baru yaitu `mysql_affected_rows()`. Fungsi ini akan mengembalikan jumlah baris yang dihasilkan dari query `INSERT`, `DELETE` dan `UPDATE`.

\*\*\*\*\*

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

# Proyek Aplikasi Bab 3 – Java Kids Foundation

Karena proyek ini source codenya cukup banyak maka kita akan membuatnya dalam waktu dua hari. Yaitu hari ke tiga belas dan hari ke empat belas. Jadi proyek bab 3 ini diharapkan selesai dalam waktu dua hari.

Untuk proyek kali ini akan sangat berbeda dengan proyek-proyek sebelumnya. Ini karena kita sudah tidak memakai file sebagai media penyimpanan. Tapi kita sekarang menggunakan sistem database. Proyek bab 3 kali ini bertitel “Java Kids Foundation”. Website ini nantinya memiliki sistem **membership** artinya terdapat member area dan untuk masuk user harus mengetikkan username dan password.

Selain itu kita juga menampilkan berita disebelah kiri halaman. Berita ini juga menggunakan database sebagai media penyimpanan. Pada halaman utama berita yang kita tampilkan hanya cuplikannya saja yaitu sebanyak 25 kata pertama. Dan untuk membaca secara lengkap user perlu mengklik link [berita lengkap](#).

Pada proyek ini juga terdapat fasilitas buku tamu dan form kontak kami. Jadi boleh dibilang proyek kali ini merupakan proyek yang paling mendekati aplikasi web sebenarnya. Untuk keperluan manajemen berita, member yang bergabung dan penghapusan entry buku tamu. Kita sediakan **control panel** untuk admin.

Beberapa fasilitas yang ada pada proyek ini sudah pernah kita terapkan sebelumnya. Hanya saja kali ini lebih kita kembangkan. Fasilitas-fasilitas yang ada antara lain:

- Fasilitas member
- Fasilitas berita
- Buku tamu
- Kontak kami
- Lupa password
- Control panel untuk admin

Control panel yang kita sediakan nantinya dapat kita gunakan untuk menghapus, mengedit, dan menambah berita. Selain itu dapat pula digunakan untuk menghapus member atau buku tamu yang kita anggap “sampah”.

Untuk itu ikuti beberapa langkah konfigurasi sebelum kita mulai membuat proyek bab 3 ini, ikuti langkah-langkah berikut:

- Buat folder baru pada C:\Apache2\htdocs\webmaster\Bab3\
- Namai folder tersebut **hari-13-14**
- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab3\hari-13-14\
- Buat folder baru, namai folder tersebut **Proyek-Bab3**
- Masuk pada direktori C:\Apahce2\htdocs\webamster\Bab3\hari-13-14\Proyek-Bab3\
- Buat beberapa folder baru, yaitu **admin, member, dan template**

Daftar file yang akan kita buat untuk aplikasi proyek bab 3 ini ditunjukkan pada tabel 3.7 berdasarkan letak direktori.

**Tabel 3.7** Daftar file untuk proyek Java Kids Foundation

Direktori <b>Proyek-Bab3</b>	
Nama File	Fungsi
index.php	Menampilkan halaman utama dan cuplikan berita
daftar.php	Menampilkan form registrasi member

login.php	Menampilkan form login
bukutamu.php	Menampilkan dan memproses form buku tamu
kontak.php	Menampilkan form kontak
javakids.sql	Menampilkan form lupa password Untuk membuat database dan tabel
<b>Direktori Proyek-Bab3\admin</b>	
Nama File	Fungsi
index.php	Menampilkan form login atau menu utama untuk admin
berita.php	Mengedit, menambah dan menghapus berita
bukutamu.php	Menghapus entry buku tamu
member.php	Menampilkan halaman member area
fungsi.php	Menyimpan kumpulan fungsi
<b>Direktori Proyek-Bab3\template</b>	
Nama File	Fungsi
index_var.php	Menyimpan kumpulan variabel untuk halaman non member
member_var.php	Menyimpan kumpulan variabel untuk halaman member area
java_skin.php	Template untuk halaman non member
member_skin	Template untuk halaman member area
template.php	Menyimpan class skin
index.php	Meredirect ke halaman utama
<b>Direktori Proyek-Bab3\member</b>	
Nama File	Fungsi
index.php	Menampilkan halaman utama member
profil.php	Mengupdate profil

Untuk file pertama yang akan kita buat, kita mulai dari direktori **admin**. Sekarang Jalankan PHP DESIGNER 2006 klik menu **File-New-SQL** untuk membuat file SQL baru. Ketik kode berikut ini

```
-- File untuk membuat database
-- buat database bernama javakids
-- hapus jika sudah ada

DROP DATABASE IF EXISTS javakids;
CREATE DATABASE javakids;

-- aktifkan database javakids
USE javakids;

-- buat tabel admin
CREATE TABLE admin
(
 username varchar(16) NOT NULL,
 password varchar(16) NOT NULL,
 PRIMARY KEY(username)
) TYPE='MyISAM';

-- buat tabel bukutamu
-- kolom id_bt bersifat auto increment artinya otomatis bertambah satu
-- saat dilakukan UPDATE pada tabel
CREATE TABLE bukutamu
(
 id_bt int(4) NOT NULL auto_increment,
 tgl_post varchar(20) NOT NULL,
 nama varchar(30) NOT NULL,
 email varchar(30) NOT NULL,
 komentar varchar(255) NOT NULL,
```

---

```

PRIMARY KEY(id_bt)
) TYPE='MyISAM';

-- buat tabel member
CREATE TABLE member
(
 username varchar(16) NOT NULL,
 password varchar(16) NOT NULL,
 nama varchar(30) NOT NULL,
 email varchar(30) NOT NULL,
 alamat varchar(100) NOT NULL,
 kodepos int(7) NOT NULL,
 kota varchar(20) NOT NULL,
 PRIMARY KEY(username)
) TYPE='MyISAM';

-- buat tabel news
CREATE TABLE news
(
 id_berita int(4) NOT NULL auto_increment,
 jdl_berita varchar(75) NOT NULL,
 isi_berita text NOT NULL,
 tgl_berita varchar(12) NOT NULL,
 PRIMARY KEY(id_berita)
) TYPE='MyISAM';

-- masukkan username dan password untuk admin
INSERT INTO admin VALUES ('admin', 'javakids123');

```

---

Simpan pada direktori **Proyek-Bab3\admin\** dengan nama ***javakids.sql***.

Seperti yang pernah kita bahas untuk mengeksekusi file SQL kita dapat menggunakan MySQL Monitor atau phpMyAdmin. Karena pada webhosting banyak digunakan phpMyAdmin maka kita akan menggunakan metode ini agar anda terbiasa. Jalankan browser anda arahkan ke <http://localhost/webmaster/phpMyAdmin2.x.x> ganti x dengan versi anda. Klik gambar yang terletak diatas daftar database.

Setelah muncul window klik tombol tab **Import files** pada query box klik tombol browse arahkan ke C:\Apache2\htdocs\webmaster\Bab3\hari-13-14\Proyek-Bab3\admin\ pilih ***javakids.sql*** lalu klik Open. Langkah yang terakhir klik tombol **Go** untuk memproses file.

Selanjutnya kita akan membuat file yang membangun sistem Control Panel untuk admin. Kita mulai dengan membuat file yang isinya adalah kumpulan fungsi yang akan kita gunakan hampir diseluruh file yang kita buat. Jalankan PHP DESIGNER 2006 klik **File-New-PHP** lalu salin listing kode berikut

Script dari file ***fungsi.php***

---

```

<?php

// fungsi untuk koneksi ke MySQL
function koneksi_db($localhost='localhost', $user='javauser', $pass='pas123',
$db='javakids')
{
 @ $koneksi = mysql_connect($localhost, $user, $pass);
 if (!$koneksi)
 return false;

 mysql_select_db($db);
}

//fungsi untuk login

```

```

function login($tabel, $username, $password)
{
 $query = "SELECT * FROM $tabel WHERE username='$username' AND
 password='$password'";
 $hasil = mysql_query($query);
 // cek jumlah baris yang dikembalikan
 if (mysql_num_rows($hasil) > 0) //berarti cocok
 return true;
 else
 return false;
}

// cegah dari SQL Injection dan cross site scripting
function filter_str($string)
{
 $filter = ereg_replace('[^a-zA-Z0-9_]', '', $string);
 return $filter;
}

//cek setiap field apa ada yang kosong
function cek_field($var)
{
 foreach ($var as $field)
 {
 if ($field == '' || !isset($field))
 return false;
 }
 return true;
}

// cek kevalidan email
function cek_email($email)
{
 // fungsi untuk mengecek kevalidan email
 if (ereg('^[a-zA-Z0-9_-]+@[a-zA-Z0-9\-.]+\.[a-zA-Z0-9\-.]+\$', $email))
 return true;
 else
 return false;
}

// fungsi untuk logout
function logout($nama_session)
{
 if (isset($_SESSION[$nama_session]))
 {
 unset($_SESSION[$nama_session]);
 session_destroy();
 return true;
 }
 else
 return false;
}

// cek session
function cek_session($nama_session)
{
 if (isset($_SESSION[$nama_session]))
 return true; // session login terisi
 else
 return false; // session login kosong
}
?>

```

Simpan pada direktori **Proyek-Bab3\admin\** dengan nama ***fungsi.php***.

---

**SQL Injection** merupakan suatu metode untuk mengeksplorasi sistem dengan cara memanipulasi query SQL. Manipulasi biasanya dilakukan pada form input yang menyediakan textbox.

## PENJELASAN SCRIPT

Pada fungsi yang pertama yaitu `konek_db()`. Kita memberikan fungsi ini empat parameter yang semuanya bersifat opsional. Mengapa kita memberikan parameter ini? Karena fungsi ini menggunakan database dan tabel. Maka suatu waktu mungkin tabel atau user yang kita gunakan berbeda. Jadi kita tinggal memasukkannya pada parameter tanpa harus mengubah kode.

Pada blok fungsi `konek_db()` kita membuat variabel `$koneksi` yang isinya adalah nilai yang dikembalikan dari fungsi `mysql_connect()`. Jika berhasil maka nilainya *true* akan tetapi jika gagal melakukan koneksi kembalikan nilai *false*. Setelah terhubung dengan MySQL kita tentu harus memilih database maka kita panggil fungsi `mysql_select_db()`. Database yang digunakan adalah *javakids* jika user tidak menentukannya dari parameter.

Fungsi selanjutnya adalah fungsi untuk login yaitu `login()`. Fungsi ini dapat kita gunakan untuk login member maupun admin karena memiliki tiga parameter. Yaitu tabel, username dan password yang digunakan. Jika kita menggunakan fungsi ini pada halaman member maka kita isikan 'member' pada parameter pertama yang merupakan tabel member. Selanjutnya kita akan melakukan query untuk mencocokkan data. Jika jumlah baris yang dikembalikan lebih dari satu maka username dan password cocok. Karena itu kembalikan nilai *true* selain itu *false*.

Fungsi selanjutnya yaitu `filter_str()` bertujuan untuk mencegah SQL injection yaitu metode yang digunakan untuk memanipulasi perintah SQL. Pada fungsi ini kita mengganti karakter yang *bukan* alfabet, angka dan tanda \_ (underscore). Tanda ^ didalam kurung siku pada fungsi `ereg_replace()` menandakan *yang tidak termasuk*. Jadi jika user mengetikkan karakter ' atau > pada form input. Maka secara otomatis akan diganti dengan karakter kosong.

Fungsi `cek_field()` merupakan fungsi yang berguna untuk mengecek setiap field yang ada pada form input. Karena data yang kita ambil dari form input menggunakan variabel `$_POST` yang merupakan array. Kita menggunakan looping `foreach` untuk mengecek setiap field. Jadi jika masih ada field yang kosong maka kembalikan nilai *false*.

Fungsi `cek_email()` sudah pernah kita gunakan pada bab-bab sebelumnya. Sekedar mengingatkan fungsi ini mengecek kevalidan email dengan menggunakan fungsi `ereg()`. Jika diterjemahkan fungsi tersebut adalah sebagai berikut *setidaknya awali dengan karakter alfabet atau angka atau tanda – lalu tanda @ lalu alfabet atau angka lalu titik lalu alfabet atau angka atau titik hingga akhir string*.

Untuk fungsi `cek_session()` kita mengecek isi session dengan menggunakan parameter `$nama_session`. Nama session ini ditentukan oleh user ketika memanggil fungsi ini. Jika session terisi maka kembalikan nilai *true* selain itu kembalikan nilai *false*. Hal ini bertujuan agar session pada admin area berbeda dengan member area. Karena jika nama sessionnya sama maka jika user dapat login di member area. Secara otomatis dia dapat mengakses halaman admin area.

Kembali ke proyek kita, file berikutnya yang akan kita buat adalah ***index.php***. File ini memiliki banyak fungsi diantaranya menyediakan login bagi admin, fungsi logout, dan juga menampilkan menu untuk control panel jika admin berhasil login. Klik menu **File-New-PHP** pada PHP DESIGENR 2005 lalu ketik kode berikut

Script dari file ***index.php***

---

```
<?php
```

```

// mulai session
session_start();
include('fungsi.php');

// ambil data dari URL yaitu proses
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'login';
// cegah cross site scripting
$proses = filter_str($proses);

switch ($proses)
{
case 'login': // titik dua bukan titik koma
 // cek apakah user sudah login atau belum
 if (!cek_session('admin'))
 {
 echo '<h3>JKF Admin Area, Please Login</h3>' .
 '<form action="index.php?proses=proses_login" method="post">' .
 '<table border="0" cellpadding="0">' .
 '<tr><td>Username: </td>' .
 '<td><input type="text" name="username"></td></tr>' .
 '<tr><td>Password: </td>' .
 '<td><input type="password" name="password"></td></tr>' .
 '<tr><td><input type="submit" value="LOGIN"></td></tr>' .
 '</table></form>';
 }
 else
 {
 echo '<h3>Welcome to JKF\'s Admin Area</h3>' .
 '<p>Silahkan pilih link proses berikut:</p>' .
 'Manaje berita
' .
 'Manaje member
' .
 'Manaje buku tamu
' .
 'Logout
';
 }
break; // akhir dari proses login

case 'proses_login':
 // cegah SQL Injection
 $username = filter_str($_POST['username']);
 $password = filter_str($_POST['password']);

 // panggil fungsi login untuk mencocokan data
 koneksi_db(); // koneksi ke MySQL Server
 if (!login('admin', $username, $password))
 echo 'Username atau password salah!.

' .
 '<< Kembali';

 else
 {
 // buatkan session karena berhasil login
 $_SESSION['admin'] = $username; // buat session bernama admin
 echo 'Login berhasil. Klik di sini untuk masuk ' .
 'admin area.';
 }
break; // akhir dari proses proses_login

case 'logout':
 if (!logout('admin'))
 echo 'Tidak bisa logout. Login dulu.';
 else
 echo 'Anda telah logout dari sistem. Login lagi';
break; // akhir dari proses logout

```

```
} // akhir dari switch
?>
```

Simpan pada direktori **Proyek-Bab3\admin\** dengan nama **index.php**. Tampilan file index.php ditunjukkan oleh gambar 3.14 dan 3.15.

### PENJELASAN SCRIPT

Karena file-file administrator ini berhubungan dengan session maka pada awal file kita panggil fungsi `session_start()`. Selanjutnya kita mengambil data dari URL yaitu data *proses*. Jika proses kosong maka tentukan proses yang berlangsung sebagai login.

Kita juga memanggil fungsi `filter_str()`. Fungsi ini kita gunakan untuk mencegah *cross site scripting* yaitu manipulasi URL untuk mendapatkan data. Data yang kita dapat dari URL akan disaring dan yang tersisa adalah alfabet, angka dan \_ (underscore).

Untuk menghadapi setiap proses kita gunakan statemen `switch()` dan `case`. Kondisi yang ditangani oleh case tergantung dari isi proses pada URL. Pada case yang pertama yaitu 'login' kita mengecek terlebih dahulu session admin dengan menggunakan `cek_session('admin')`. Apakah sudah terisi atau belum. Jika sudah maka admin telah login maka tampilkan control panel dan jika belum maka admin belum login untuk itu tampilkan form login.

*Catatan: untuk login gunakan username **admin** dan password **javakids123***



Gambar 3.14: Form login untuk admin

Case yang selanjutnya yaitu 'proses\_login'. Pada blok kode case ini kita memproses form login yang dikirim dari halaman login admin. Pertama kita membuat variabel untuk menampung data dari form dengan menggunakan `$_POST`. Untuk mencegah SQL injection kita juga memanggil fungsi `filter_str($_POST['username'])`. Hal ini kita lakukan untuk memfilter setiap karakter yang dimasukkan user.

Setelah itu kita memanggil fungsi `konek_db()` untuk mengkoneksikan PHP dengan MySQL. Untuk mencocokkan username dan password kita memanggil fungsi `login()`. Karena didahului `if (!login('admin', $username, $password))` maka artinya jika ada kesalahan atau nilainya false(username atau password tidak cocok).

Arti dari 'admin' pada parameter pertama menunjukkan tabel yang ingin kita gunakan. Jika tidak cocok jangan buatkan session, sebaliknya jika cocok dengan yang ada di database buatkan session yang bernama *admin* yang isinya username admin.



Gambar 3.15: Control panel sederhana untuk admin

Case yang terakhir yaitu logout. Pada blok kode case ini kita menggunakan statemen `if (!logout('admin'))` untuk mengecek kondisi. Jadi jika session admin kosong atau belum diset, maka admin belum login. Selain itu berarti admin telah login jadi hapus semua session dan hancurkan dengan `session_destroy()`.

File berikut ini berguna untuk memanage member yang telah mendaftar di website `javakidsfoudnation.com`. Jika ada entry-entry yang kita anggap "sampah" kita dapat menghapusnya dari halaman ini. Untuk itu buatlah file PHP baru dengan mengklik **File-New-PHP** lalu ketik listing kode berikut

Script dari file ***member.php***

```
<?php
// mulai session
session_start();
include('fungsi.php');
koneksi_db(); // koneksi ke MySQL server

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';
// cegah cross site scripting
$proses = filter_str($proses);

// cek user apakah sudah login atau belum
if (!cek_session('admin'))
 echo 'Anda belum login. Silahkan Login dulu';
else
{
switch ($proses)
{
case 'view':
 $hasil = mysql_query("SELECT * FROM member ORDER BY nama");
 // tampilkan semua member
 echo '<h2>Daftar Member</h2><hr>';
 . '<table border="1" cellpadding="4" style="border-collapse:collapse">
 . <tr bgcolor="#cccccc">
 . <th>Username</th><th>Password</th><th>Nama</th><th>Email</th>
```

```

 . '<th>Alamat</th><th>Kode Pos</th><th>Kota</th><th>Hapus?</th>'
 . '</tr>';
 while ($data = mysql_fetch_array($hasil))
 {
 $hapus = '' .
 'Hapus';
 echo '<tr>';
 // lakukan looping untuk menulis semua field
 for ($i=0; $i<7; $i++)
 {
 echo '<td>' . $data[$i] . '</td>';
 }
 echo '<td>' . $hapus . '</td>' .
 '</tr>';
 }
 echo '</table>' .
 '<p>Halaman Utama;';
break; // akhir dari proses view

case 'hapus':
 // dapatkan username yang akan dihapus
 $user = $_GET['username'];
 // hapus dari database
 $hasil = mysql_query("DELETE FROM member WHERE username=' $user '");
 if (!$hasil)
 echo "Error: gagal menghapus data dari database.";
 else
 echo 'Data berhasil dihapus dari database. Lihat';

break; // akhir dari proses hapus
} // akhir dari switch
} // akhir dari else

mysql_close(); // tutup koneksi
?>

```

Simpan pada direktori **Proyek-Bab3** dengan nama **member.php**. Tampilan file member.php dapat anda lihat seperti gambar 3.16

**Cross Site Scripting** merupakan salah satu metode untuk mengeksplorasi sistem. Cara yang digunakan biasanya dengan memasukkan karakter-karakter tertentu pada URL.

## PENJELASAN SCRIPT

Pada file member.php ini kita memiliki dua proses yaitu proses *view* untuk menampilkan daftar member. Dan proses *hapus* untuk menghapus member. Pada proses *view* kita melakukan query terlebih dahulu untuk mendapatkan daftar member yaitu dengan perintah

```
$hasil = mysql_query("SELECT * FROM member ORDER BY nama");
```

Pada query SQL diatas kita akan mendaftar member yang ada urut berdasarkan nama mereka secara ascending. Kemudian kita membuat header/kepala dari tabel. Agar tidak ikut diulang dalam looping maka kita letakkan header tabel ini diluar kode looping while.

Untuk menampilkan semua daftar member kita gunakan looping while seperti berikut

```
while ($data = mysql_fetch_array($hasil))
```

Blok kode yang ada pada looping while akan dieksekusi sebanyak jumlah baris yang dikembalikan oleh query SQL diatas sebelumnya. Kita membuat variabel **\$hapus** yang isinya adalah link proses hapus dan username yang isinya adalah username yang akan dihapus. Kita

membuat data pada URL seperti ini agar kita dapat menentukan user mana yang akan dihapus ketika dilakukan query penghapusan.

The screenshot shows a Microsoft Internet Explorer window with the title bar "http://localhost/webmaster/Bab3/hari-13/Proyek-Bab3/admin/member.php - Microsoft Internet Explorer". The address bar also displays the same URL. Below the title bar is a toolbar with standard icons for Back, Forward, Stop, Refresh, and Favorites. The main content area is titled "Daftar Member" and contains a table with the following data:

Username	Password	Nama	Email	Alamat	Kode Pos	Kota	Hapus?
therno121	[REDACTED]	Rio Astamal	astamal@unlimitedmail.org	Jl. Made Selatan 19 3/1	60219	Surabaya	<a href="#">Hapus</a>
sanjaya	[REDACTED]	Sanjaya S.	san@jaya.com	Jl. Selalu Jaya no. wahid	60219	Surabaya	<a href="#">Hapus</a>

Gambar 3.16: Daftar member pada control panel

Karena kita tahu jumlah kolom atau field pada tabel member adalah 7. Maka kita gunakan looping for untuk mempersingkat kode. Jadi isi dari \$data nantinya adalah \$data[0], \$data[1] sampai \$data[6]. \$data[0] adalah kolom username, \$data[1] adalah kolom password dan seterusnya. Setelah looping for kita tambahkan satu kolom yang isinya adalah link *hapus*.

Case yang ke dua yaitu *hapus*. Pada case ini kita mengambil data yaitu username dari URL untuk menentukan record yang akan dihapus. Kita juga menfilter untuk data yang kita ambil menggunakan filter\_str(\$\_GET['username']). Langkah berikutnya tentu adalah melakukan query penghapusan seperti berikut

```
$hasil = mysql_query("DELETE FROM member WHERE username='\$user'");
```

Kita menghandle hasil dari query ini menggunakan if (! \$hasil). Jika ada kesalahan atau nilainya false tampilkan pesan gagal selain itu tampilkan pesan berhasil.

Hampir sama dengan file *member.php* file berikut ini juga memiliki dua proses yaitu *view* untuk melihat daftar buku tamu dan *hapus* untuk menghapus entry buku tamu. Langsung saja klik **File-New-PHP** lalu salin kode berikut ini

Script dari file **bukutamu.php**

```
<?php
// mulai session
session_start();
include('fungsi.php');
koneksi(); // koneksi ke MySQL server

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';
// cegah cross site scripting
$proses = filter_str($proses);

// cek user apakah sudah login atau belum
if (!cek_session('admin'))
 echo 'Anda belum login. Silahkan Login dulu';
else
{
 switch ($proses)
```

---

```

{
case 'view':
$hasil = mysql_query("SELECT * FROM bukutamu ORDER BY id_bt DESC");
// tampilkan semua bukutamu
echo '<h3>Daftar Buku Tamu</h3><hr>';
while ($data = mysql_fetch_array($hasil))
{
 echo 'Nama: '.$data['nama'].'

 .'Email: '.$data['email'].'

 .'Isi: '.$data['komentar'].'

 .'Hapus
 .'<hr>';
}
echo '<p>Halaman utama</p>';
break; // akhir dari proses view

case 'hapus':
// dapatkan id untuk bukutamu yang akan dihapus
$id = $_GET['id'];
// lakukan query DELETE
$hasil = mysql_query("DELETE FROM bukutamu WHERE id_bt='".$id "'");
if (!$hasil)
 echo 'Error: tidak dapat menghapus data dari database';
else
 echo 'Data berhasil dihapus. Lihat';

break; // akhir dari proses hapus
} // akhir dari switch
} // akhir dari else

mysql_close(); // tutup koneksi
?>

```

---

Simpan pada direktori **Proyek-Bab3** dengan nama **bukutamu.php**. Output dari file ini akan terlihat seperti yang ditunjukkan gambar 3.17.

### PENJELASAN SCRIPT

Karena sistem kerja dari file ini mirip dengan file sebelumnya yaitu *member.php* maka kali ini saya hanya akan menjelaskan poin-poinnya saja. Pada proses view intinya kita akan menampilkan seluruh isi dari tabel bukutamu. Query SQL yang kita gunakan adalah

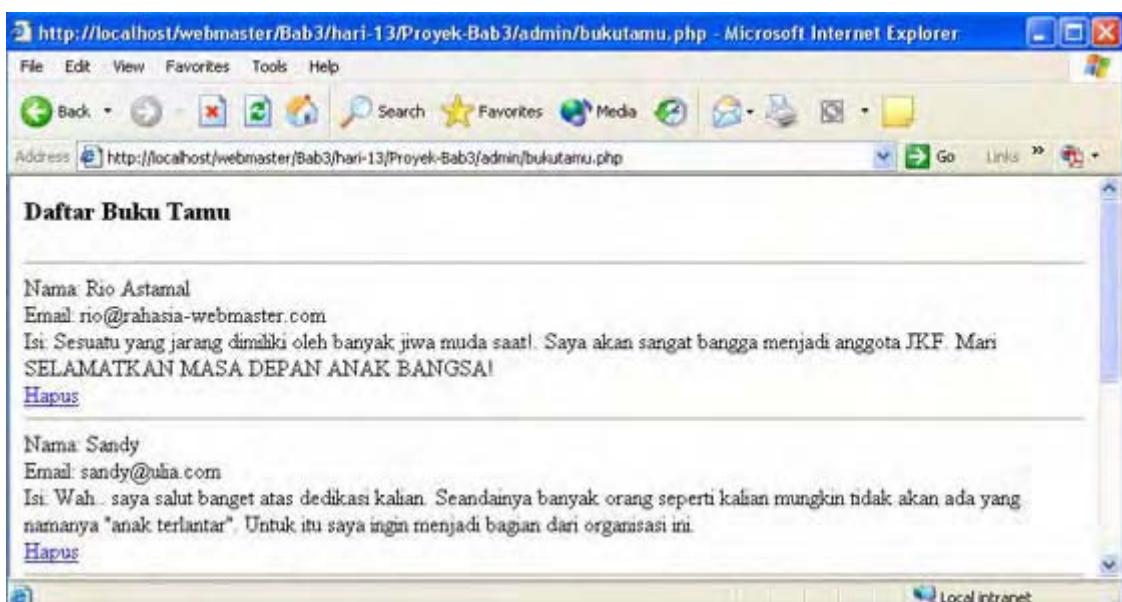
```
$hasil = mysql_query("SELECT * FROM bukutamu ORDER BY id_bt DESC");
```

Artinya kita menampilkan record berdasarkan id\_bt diurut secara descending(besar ke kecil). Kemudian untuk menampilkan semua recordnya kita gunakan looping while seperti yang pernah kita lakukan pada file *member.php*. Pada setiap record kita tambahkan sebuah link *Hapus* yang isi prosesnya adalah hapus dan variabel id yang isinya merupakan id\_bt dari record buku tamu yang akan dihapus.

Pada case *hapus* kita mengambil data pada variabel id yang ada pada URL. Lalu kita lakukan query DELETE untuk menghapus record buku tamu sebagai berikut

```
$hasil = mysql_query("DELETE FROM bukutamu WHERE id_bt='".$id "'");
```

Jika tidak berhasil tampilkan pesan error selain itu tampilkan pesan bahwa proses penghapusan berhasil.



Gambar 3.17: Tampilan buku tamu pada control panel

File berikut ini memiliki cara kerja yang cukup berbeda dengan dua file sebelumnya yaitu *member.php* dan *bukutamu.php*. File *berita.php* memiliki enam proses sekaligus. Wah raku benar file ini...!. Dengan menghandel banyak proses sekaligus kita dapat meminimalisir jumlah file untuk proyek kita.

Jika kita menyediakan file untuk tampilan form dan file untuk memproses form secara sendiri-sendiri. Jumlah file kita akan semakin banyak. Untuk itu daripada membuat banyak file akan lebih efisien jika kita menyatukan semua proses pada satu file.

Untuk membuat file ini buat file PHP baru pada PHP DESIGNER 2005 klik **File-New-PHP** lalu ketik kode berikut

Script dari file ***berita.php***

```
<?php
// mulai session
session_start();
include('fungsi.php');
koneksi_db(); // koneksi ke MySQL server

// ambil data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';
//cegah cross site scripting
$proses = filter_str($proses);

// cek apakah user telah login atau belum
if (!cek_session('admin'))
 echo 'Anda belum login. Silahkan Login dulu';
else
{
switch ($proses)
{
case 'view':
 echo '<h1>Daftar Berita</h1><hr>';
 $hasil = mysql_query("SELECT * FROM news ORDER BY id_berita DESC");
 while ($data = mysql_fetch_array($hasil))
 {
 echo $data['id_berita'] . ' [Posted: ' . $data['tg_berita'] . '] ' . '
```

```

.' [EDIT'
.']'
.' [HAPUS'
.']

';
}
echo 'Halaman utama <a href="'
.'berita.php?proses=tambah">Tambah Berita';
break; // akhir dari proses view

case 'edit':
// dapatkan id berita yang diedit
$id = $_GET['berita'];
// cegah cross site scripting
$id = filter_str($id);

$hasil = mysql_query("SELECT * FROM news WHERE id_berita='".$id "'");
$data = mysql_fetch_array($hasil);
// tampilkan form edit
echo '<h2>Edit Berita</h2><hr>'
.'<form action="berita.php?proses=proses_edit" method="post">'
.'Judul:
'
.'<input type="text" name="judul" size="50"
value='.$data['jdl_berita'].'>

'
.'Isi berita:
'
.'<textarea name="isi" cols="70" rows="10">
'.$data['isi_berita'].'</textarea>

'
.'Tanggal Berita:
'
.'<input type="text" name="tgl" value="'.$data['tgl_berita'].'">

'
.'<input type="hidden" name="id" value="'.$data['id_berita'].'">'
.'<input type="submit" value="EDIT BERITA">
'
.'</form>'
.'<p>Halaman Utama Tampilkan
Berita</p>';
break; // akhir dari proses edit

case 'proses_edit':
$id = $_POST['id'];
$judul = $_POST['judul'];
$isi = $_POST['isi'];
$tgl = $_POST['tgl'];
// lakukan update
$query = "UPDATE news SET jdl_berita='".$judul', isi_berita='".$isi',
tgl_berita='".$tgl' WHERE id_berita='".$id."'";
$hasil = mysql_query($query);
if (!$hasil)
echo 'Error: Gagal mengupdate database';
else
echo 'Berita berhasil diupdate. Tampilkan berita';

break; // akhir dari proses proses_edit

case 'tambah':
echo '<h2>Tambah Berita</h2><hr>'
.'<form action="berita.php?proses=proses_tambah" method="post">'
.'Judul:
<input type="text" name="judul" size="50">

'
.'Isi Berita:
<textarea name="isi" cols="70"
rows="10"></textarea>

'
.'Tanggal berita:
'
.'<input type="text" name="tgl" value="Format: DD-MM-YYYY">

'
.'<input type="submit" value="TAMBAH BERITA">'
.'</form>'
.'<p>Halaman Utama Tampilkan
Berita</p>';
break; // akhir dari proses tambah

```

---

```

case 'proses_tambah':
 $judul = $_POST['judul'];
 $isi = $_POST['isi'];
 $tgl = $_POST['tgl'];
 // cek semua field apa sudah terisi
 if (!cek_field($_POST))
 exit('Error: masih ada field yang kosong');
 // masukkan data ke database
 $query = "INSERT INTO news (jdl_berita, isi_berita, tgl_berita) VALUES
 ('$judul', '$isi', '$tgl')";
 $hasil = mysql_query($query);
 if (!$hasil)
 echo 'Error: Tidak dapat memasukkan berita ke database.';
 else
 echo 'Berita berhasil ditambah ke database. Lihat';

break; // akhir dari proses_proses_tambah

case 'hapus':
 // dapatkan id dari berita yang akan dihapus
 $id = $_GET['berita'];
 // cegah cross site scripting
 $id = filter_str($id);
 // lakukan query
 $hasil = mysql_query("DELETE FROM news WHERE id_berita='$id'");
 if (!$hasil)
 echo 'Error: Gagal menghapus berita.';
 else
 echo 'Berita berhasil dihapus. Lihat';

break; // akhir dari proses_hapus
} // akhir dari switch
} // akhir dari else

mysql_close(); // tutup koneksi
?>

```

---

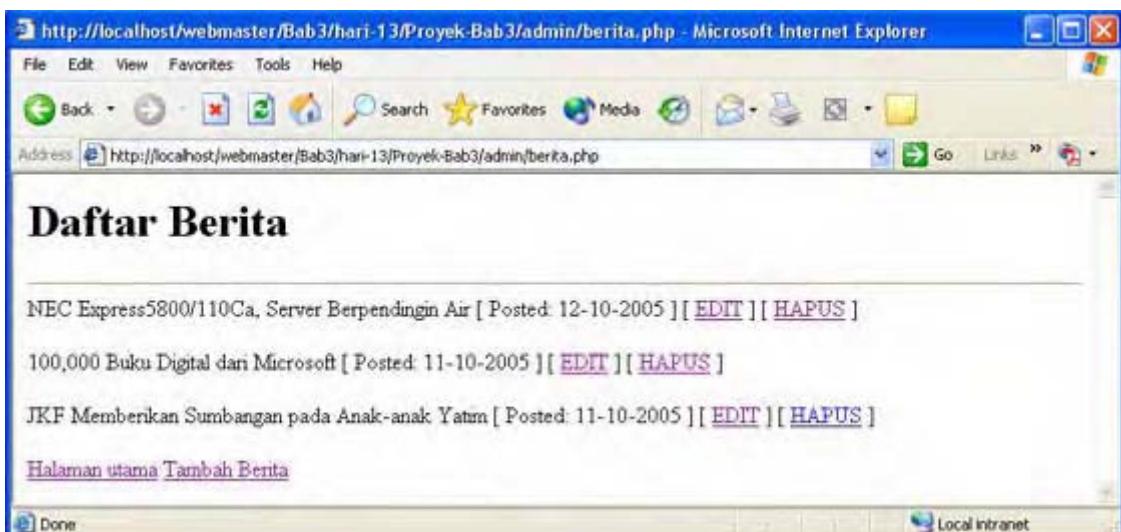
Simpan pada direktori **Proyek-Bab3** dengan nama **berita.php**. Lihat gambar 3.18 dan 3.19.

### PENJELASAN SCRIPT

Pada proses pertama yaitu *view* kita menampilkan seluruh berita yang ada pada tabel news dengan query sebagai berikut

```
$hasil = mysql_query("SELECT * FROM news ORDER BY id_berita DESC");
```

Lalu untuk menampilkan semua hasil kita gunakan looping while. Kita juga memberi link *EDIT* dan *HAPUS* disebelah setiap berita. Untuk link edit isi dari proses pada link adalah edit dan berita adalah id\_berita yang diklik. Sedangkan untuk link hapus isi dari proses adalah hapus dan berita adalah id\_berita yang diklik.



Gambar 3.18: Daftar berita pada control panel

A screenshot of Microsoft Internet Explorer version 7.0. The title bar says "http://localhost/webmaster/Bab3/hari-13/Proyek-Bab3/admin/berita.php?proses=tambah - Microsoft Internet Explorer". The menu bar includes File, Edit, View, Favorites, Tools, and Help. The toolbar has icons for Back, Forward, Stop, Refresh, Search, Favorites, Media, Mail, Print, and Home. The address bar shows the URL "http://localhost/webmaster/Bab3/hari-13/Proyek-Bab3/admin/berita.php?proses=tambah". Below the address bar, there are "Go" and "Links" buttons. The main content area contains a form for adding a news item:

Isi Berita:

Tanggal berita:

Format DD-MM-YYYY

The status bar at the bottom left says "Done" and the bottom right says "Local intranet".

Gambar 3.19: Form tambah berita pada control panel

Proses case selanjutnya yaitu *edit*. Proses ini akan aktif jika user mengklik link *EDIT* disebelah kanan judul iklan. Kita lalu menyediakan form yang mirip dengan form tambah berita *hanya saja* kita menambahkan atribut *value* pada setiap field. Isi dari atribut ini kita ambil dari hasil query

```
$hasil = mysql_query("SELECT * FROM news WHERE id_berita='$id'");
```

khusus untuk pada textbox judul valuenya kita beri escaping karakter \'. Ini berguna jika ada judul berita yang memiliki tanda " petik dua. Tanpa adanya escaping karakter \' HTML menganggap tanda " pada judul berita akan dianggap tanda penutup. Jadi isinya akan kosong.

Karena kita tahu hasil yang akan dikembalikan hanya satu record maka kita tidak perlu melakukan looping. Pada query diatas kita memilih berita yang akan diedit berdasar pada *id\_berita* yang nilainya kita dapatkan dari variabel *id* pada URL saat user mengklik link *EDIT*.

Proses pada case *proses\_edit* akan memproses data yang dikirim dari form edit berita. Untuk itu kita buat variabel untuk menampung setiap data yang dikirim. Karena perubahan yang akan dilakukan hanya isi pada suatu record. Maka kita lakukan query *UPDATE* untuk merubah isi dari record tersebut. Query yang kita tuliskan adalah sebagai berikut

---

```
$query = "UPDATE news SET jdl_berita='$judul', isi_berita='$isi',
tgl_berita='$tgl' WHERE id_berita='$id'";
```

Query diatas berarti kita mengupdate isi dari judul, isi dan tanggal berita berdasarkan id\_berita tersebut. Id berita ini kita POST menggunakan metode **hidden** artinya tidak tampak dilayar browser.

Pada proses case *tambah* tidak ada yang istimewa. Kode ini hanya berfungsi untuk menampilkan form tambah berita. Sedangkan pada *proses\_tambah* kita mengambil data dari form tambah berita dengan metode POST. Setelah semua data kita isikan pada variabel kemudian kita lakukan query untuk memasukkan record baru ke database.

```
$query = "INSERT INTO news (jdl_berita, isi_berita, tgl_berita) VALUES
('$judul', '$isi', '$tgl')";
```

Pada query SQL diatas kita memasukkan data hanya pada kolom jdl\_berita, isi\_berita dan tgl\_berita. Mengapa kita tidak mengisi data pada kolom id\_berita?. Pertanyaan bagus, sebelumnya ingat kembali tipe data dari kolom id\_berita yaitu **auto\_increment**. Jadi setiap ada record baru yang dimasukkan pada tabel maka nilai dari kolom ini otomatis ditambah satu.

Akhir dari file yang berada pada direktori **Proyek-Bab3\admin\**

---

File-file yang akan kita bahas berikutnya adalah file pada direktori **template**. File pertama yang akan kita bahas adalah file yang me-redirect lokasi jika user mencoba membuka direktori *template* pada URL. Klik Menu **File-New-PHP** untuk membuat file PHP baru, kemudian ketik kode berikut

Script dari file **index.php**

---

```
<?php
// redirect ke halaman utama
header('Location: ../index.php');
?>
```

Simpan pada direktori **Proyek-Bab3\template\** dengan nama **index.php**.

#### **PENJELASAN SCRIPT**

Fungsi file ini adalah untuk me-redirect lokasi halaman ke index.php pada **Proyek-Bab3**. Jika user coba membuka direktori template lewat browser maka secara otomatis halaman yang dipanggil *index.php* pada direktori **Proyek-Bab3**.

Tanpa kita redirect user dapat melihat daftar dari file yang berada pada direktori ini tentu merupakan masalah security. Fungsi yang kita gunakan untuk meredirect adalah header('Location: ../index.php'). Fungsi header() berguna untuk mengirimkan HTTP header kepada browser. Karena isinya *Location* maka fungsinya untuk me-redirect halaman.

File berikutnya adalah file yang menjadi dasar tampilan dari semua halaman non member yaitu file *java\_skin.php*. File ini sebenarnya hanya terdiri kode HTML saja. Untuk membuatnya klik menu **File-New-PHP** pada PHP DESIGNER 2006 untuk membuat file baru lalu ketik kode berikut

Script dari file **java\_skin.php**

---

```
<html>
<head>
<title>Java Kids Foundation</title>
<style>
body{font-family: verdana; font-size: 11px;}
td{font-family: verdana; font-size: 11px}
table{border-collapse: collapse; border-color: #cccccc}
```

---

```

a{color: #7cb500}
a:hover{color: #7cb500}
.batas{margin-top: 0; margin-left: 0; margin-right: 0; margin-bottom: 0}
.putih{font-weight: bold; color: white}
</style>
</head>
<body class="batas">

| | |
|--|-------------------------|
| <h2>{JUDUL}</h2></td> | |
| | |
| BERITA TERBARU | LETS SAVE THE CHILDREN! |
| {NEWS} | {MENU}<p></p>{UTAMA} |
| © 2005
javakidsfoundation.localhost | |

</body>
</html>

```

---

Simpan pada direktori **Proyek-Bab3** dengan nama ***java\_skin.php***. Jika anda jalankan tampilan file ini akan terlihat seperti gambar 3.20.



Gambar 3.20: Template dasar untuk semua halaman non member

File berikut ini adalah file template yang kita gunakan pada halaman member. Tampilan file ini hanya memiliki satu kolom. Untuk selebihnya anda dapat memodifikasi sendiri tampilan-tampilan lainnya. Untuk membuat file ini klik **File-New-PHP** pada PHP DESIGNER 2006 ketik kode berikut.

Script dari file ***member\_skin.php***

---

```

<html>
<head>
<title>Java Kids Foundation</title>
<style>
 body{font-family: verdana; font-size: 11px;}
 td{font-family: verdana; font-size: 11px}

```

```

table{border-collapse: collapse; border-color: #cccccc}
a{color: #7cb500}
a:hover{color: #7cb500}
.batas{margin-top: 0; margin-left:0; margin-right:0; margin-bottom: 0}
.putih{font-weight: bold; color: white}
</style>
</head>
<body class="batas">

| | | | |
|--|---------------------------|--|--|
| <h2>{JUDUL}</h2></td> | | | |
| MEMBER AREA</td> | {MENU}<p></p>{UTAMA}</td> | © 2005 javakidsfoundation.com</td> | |
| MEMBER AREA</td> | | | |
| {MENU}<p></p>{UTAMA}</td> | | | |
| © 2005 javakidsfoundation.com</td> | | | |

</body>
</html>

```

Simpan pada direktori **Proyek-Bab3/template** dengan nama **member\_skin.php**. Output dari file ini ditunjukkan oleh gambar 3.21.



Gambar 3.21: Template dasar untuk semua halaman member

Berikutnya kita akan membuat file template yaitu **template.php**. File sudah berkali-kali kita buat anda dapat mengcopy file ini dari Proyek-Bab2\template\ lalu *paste* ke Proyek-Bab3\template\. Namun untuk sekedar mengingatkan atau file tersebut sudah anda hapus? Maka saya tuliskan kembali sourcennya.

#### Script dari file **template.php**

```

<?php

class skin
{
 var $tag = array();
 var $file_skin;
 var $halaman;

```

---

```

function ganti_tag($namatag, $str_pengganti)
{
 $this->tag[$namatag] = $str_pengganti;
}

function ganti_skin($namafile)
{
 $this->file_skin = $namafile;
}

function ganti_tampilan()
{
 $this->halaman = file($this->file_skin);
 $this->halaman = implode("", $this->halaman);
 foreach($this->tag as $str_dicari => $str_baru)
 {
 $this->halaman = ereg_replace($str_dicari, $str_baru, $this->halaman);
 }
 echo $this->halaman;
}

}
?>

```

---

Simpan pada direktori **Proyek-Bab3\template\** dengan nama **template.php**. Untuk menjaga kompatibilitas dengan versi PHP 4. Untuk pendeklarasian properti kita tidak menggunakan **private**, **public** atau **protected**. Melainkan **var** yang artinya sama dengan **public**.

File selanjutnya adalah *index\_var.php* file ini akan selalu kita panggil pada semua halaman non member. File ini berfungsi untuk menyimpan variabel-variabel yang digunakan menampilkan judul website dan judul berita pada samping halaman. Buka PHP DESIGNER 2006 klik menu **File-New-PHP** lalu ketik kode berikut

Script dari file **index\_var.php**

---

```

<?php
// panggil file fungsi.php pada direktori admin
include('admin/fungsi.php');

$judul = 'Java Kids Foundation';

$menu = 'Home &nbsp Daftar
. '&nbsp Member Area
. '&nbsp Buku Tamu
. '&nbsp Kontak Kami';

/* Variabel untuk menampilkan judul berita */

// tampilkan semua judul berita pada samping halaman
koneksi(); // koneksi ke MySQL server
$hasil_news = mysql_query("SELECT * FROM news ORDER BY id_berita DESC");
$side_news = '<p>Daftar Berita</p>
.;
while ($data = mysql_fetch_array($hasil_news))
{
// gunakan stripslashes() untuk menghilangkan escaping character
$side_news .= '<p>
 stripslashes($data['jdl_berita']).'</p>';
}
$side_news .= '';

```

```
mysql_close();

// buatkan link untuk kembali, berguna jika ada error
$kembali = '
<< kembali' ;
?>
```

Simpan pada direktori **Proyek-Bab3\template** dengan nama **index\_var.php**.

### PENJELASAN SCRIPT

Mungkin anda bertanya file ini kan letaknya di direktori **template** jika ingin memanggil file **fungsi.php** pada direktori **admin** harusnya kan ditulis seperti ini

```
include('..../admin/fungsi.php');
```

Ya memang itu benar. Namun ingat file ini **bukan** yang kita panggil langsung dari browser. Tapi file **index\_var.php** kita panggil dari file **index.php** yang letaknya pada direktori **Proyek-Bab3** jadi kita tidak perlu memberi tanda ../ pada include.

Fungsi file ini adalah untuk memanggil file **fungsi.php** pada direktori **admin** dan menyimpan variabel \$side\_news. Isi dari variabel ini digunakan untuk menampilkan judul dari berita pada samping kiri halaman non member. Untuk memberi tanda **bullet** pada setiap judul berita, maka kita menambahkan tag <ul> dan <li>.

Seperti biasa sebelum mengambil data dari database kita harus melakukan koneksi dulu dengan MySQL. Untuk itu kita panggil fungsi **konek\_db()**. Untuk mendapatkan seluruh judul berita kita lakukan looping while. Pada link URL kita membuat variabel berita yang isinya **id\_berita**. Yang jika diklik isi lengkap berita tersebut akan ditampilkan disebelah kanan halaman.

Pada file ini kita juga membuat sebuah variabel **\$kembali** yang isinya adalah sebagai berikut

```
$kembali = '
<< kembali' ;
```

isi dari href yaitu javascript: history.back() merupakan fungsi javascript untuk kembali ke halaman sebelumnya. Dengan demikian user tidak perlu mengklik tombol back pada toolbar.

File berikutnya adalah **member\_var.php**. File ini sangat sederhana dan hanya terdiri dari beberapa variabel saja. Hal ini dikarenakan pada halaman member area kita hanya menyediakan dua halaman saja dan fokus kita pada halaman non member. Klik **File-New-PHP** pada PHP DESIGNER 2006 ketik kode berikut ini

Script dari file **member\_var.php**

```
<?php
include('..../admin/fungsi.php');

$judul = 'Java Kids Foundation - Member Area';
$menu = 'Home Profil ' .
' Logout' ;

// buatkan link untuk kembali, berguna jika ada error
$kembali = '
<< kembali' ;
?>
```

Simpan pada direktori **Proyek-Bab3/template** dengan nama **member\_var.php**. Rasanya tidak ada yang perlu dijelaskan dari file ini. Seperti yang sudah anda file ini hanya berisi beberapa variabel string.

Akhir dari file yang berada pada direktori **Proyek-Bab3\template\**

---

Kali ini file-file yang akan kita bahas adalah file yang berada pada direktori paling atas yaitu **Proyek-Bab3**. File pertama yang akan kita buat kali ini adalah file utama dari [javakidsfoundation.com](http://javakidsfoundation.com).

File ini adalah **index.php**, pada file ini nantinya akan menampilkan profil dari JKF dan menampilkan cuplikan berita sebanyak 25 kata pertama dari setiap berita.

Untuk menghemat waktu langsung saja klik **File-New-PHP** pada PHP DESIGNER 2006 lalu ketik kode berikut

Script dari file **index.php**

---

```
<?php

include('template/template.php');
include('template/index_var.php');

/* variabel-variabel pada halaman utama dari JKF */
/***********************/

$utama = '<h3>Tentang Kami</h3>
 . '<p>Java Kids foundation merupakan organisasi non profit yang didirikan oleh sekelompok anak muda lulusan UV (Universitas Virtual). Java Kids Foundation (JKF) berdiri tahun 2001. Selama kurang lebih empat tahun kami telah melakukan berbagai donasi ke yayasan anak yatim di hampir seluruh Jawa.</p>
 . '<p>Sekarang JKF sudah memiliki 18 cabang yang tersebar diseluruh kota di pulau Java. JKF juga berencana untuk membuka cabang di luar pulau Java seperti di Sumatra dan Kalimantan. Dana yang kami peroleh untuk disumbangkan berasal dari berbagai macam sumber. Namun yang terbesar adalah dari para donator tetap kami. Untuk itu kami sangat berterima kasih atas kesediaan mereka untuk terus mendukung program kami.</p>
 . '<p>Jika anda berminat untuk mengikuti program kami. Anda dapat mendaftar lewat form daftar. Setelah proses pendaftaran selesai anda akan menerima username dan password via email. Jika anda memang kurang mampu untuk menyumbang secara materi, itu tidak masalah. Anda dapat menyumbang lewat apa saja termasuk pikiran dan tenaga.</p>
 . '<p align="center">Mari Bersama Kita Selamatkan Masa Depan Mereka!</p>
 . '<p>Kantor Pusat:
Jl. Menuju Kebenaran 23
Jakarta 62152

 . 'Telp - (021) 7414698</p>

// dapatkan isi page dari URL
$page = $_GET['page'];
if ($page == '')
 $page = 0;
// cegah cross site scripting
$page = filter_str($page);
// tentukan berita per halaman
$berita_per_hal = 3;
// lakukan query lalu hitung jumlah halaman dengan ceil()
koneksi_db(); // koneksi ke MySQL server
$query_hal = mysql_query("SELECT * FROM news");
$jml_baris = mysql_num_rows($query_hal);
$jml_hal = ceil($jml_baris / $berita_per_hal); // jumlah halaman
// dapatkan nilai $offset
$record = $page * $berita_per_hal;

// cuplikan berita yang ada pada halaman utama
$hasil = mysql_query("SELECT * FROM news ORDER BY id_berita DESC LIMIT $record,
$berita_per_hal");
while ($data = mysql_fetch_array($hasil))
{

```

```

$isi_berita = str_replace("\n", "
", $data['isi_berita']);
// mencuplik berita, 25 kata pertama
$cuplikan = array();
$pecahan_kata = explode(" ", $isi_berita);
for ($i=0; $i<25; $i++)
 $cuplikan[$i] = $pecahan_kata[$i];

// gabungkan setiapkan array kata pisahkan kembali dengan spasi
$cuplikan = implode(" ", $cuplikan);
// gunakan stripslashes() untuk menghilangkan escaping character
$cuplikan = stripslashes($cuplikan);
$jdl_berita = stripslashes($data['jdl_berita']);
$link = 'berita lengkap';

$news .= '<p>'.$data['tgl_berita'].'
'.$jdl_berita.''
. '<p>'.$cuplikan.' . . .
'.$link.'</p></p><hr>';
}

// jika jumlah halaman lebih dari satu tampilkan link halaman
if ($jml_hal > 1)
{
 $news .= 'Halaman: ';
 for ($a=0; $a<$jml_hal; $a++)
 $news .= ' ' . intval($a+1)
 .' ';
}
mysql_close(); // tutup koneksi

$skin = new skin;
$skin->ganti_skin('template/java_skin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{NEWS}', $news);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **Proyek-Bab3** dengan nama **index.php**. Output dari file ini ditunjukkan oleh gambar 3.22.

### PENJELASAN SCRIPT

Seperti yang anda lihat isi dari variabel \$utama hanyalah string tag HTML yang kita gunakan untuk menampilkan profil JKF. Penjelasan di sini saya tekankan pada bagaimana cara kita mencuplik berita dan menampilkan sejumlah berita per halaman.

Untuk menampilkan sejumlah data tertentu klausa yang kita tambahkan adalah **LIMIT**. Untuk itu kita perlu suatu petunjuk halaman mana yang ingin kita tampilkan. Maka pada URL kita tambah variabel page. Jika page kosong maka tentukan page sama dengan nol. Ini untuk menghindari tampilan kosong karena page belum diset. Selanjutnya kita filter isi dari page agar tidak mengandung karakter-karakter berbahaya dengan fungsi **filter\_str()**.

Kita kan belum meng-include file ini, kenapa kok bisa dipanggil?. Hmm..., rupanya anda lupa. Pada file *index.php* kita sudah memanggil file *index\_var.php*, sedangkan didalam file *index\_var.php* kita sudah memanggil file *fungsi.php*. Jadi secara otomatis jika kita memanggil file *index\_var.php* file *fungsi.php* juga ikut terpanggil.

Selanjutnya kita membuat variabel **\$berita\_per\_hal** yang isinya adalah tiga. Jumlah cuplikan berita yang ditampilkan pada halaman utama nantinya berdasarkan isi dari variabel ini. Selanjutnya kita melakukan query seperti berikut

```
$query_hal = mysql_query("SELECT * FROM news");
```

---

```
$jml_baris = mysql_num_rows($query_hal);
```

Tujuan \$query\_hal hanyalah untuk mengembalikan jumlah baris yang kemudian dihitung oleh mysql\_num\_rows() yang hasilnya diletakkan pada variabel \$jml\_baris. Setelah jumlah baris kita ketahui maka untuk menghitung jumlah halaman kita akan membagi jumlah baris dengan berita per halaman. Lihat kode berikut

```
$jml_hal = ceil($jml_baris / $berita_per_hal); //jumlah halaman
//dapatkan nilai $offset
$record = $page * $berita_per_hal;
```

Fungsi yang kita gunakan untuk membagi jumlah baris dan berita per halaman adalah ceil(). Fungsi ceil() akan membulatkan hasil koma ke atas. Misalnya isi \$jml\_baris adalah 4 dan \$berita\_per\_hal adalah 3. Maka jika dibagi  $4/3 = 1,333\dots$  dibulatkan ke atas menjadi 2. Yaitu page 0 dan page 1.

Tujuan kita membuat variabel \$record adalah agar kita mengetahui harus dimulai dari mana baris yang ditampilkan saat kita menggunakan klausa LIMIT. Untuk page 0 atau pertama nilai dari \$record adalah  $0 * 3 = 0$ . Dengan demikian klausa LIMIT-nya menjadi seperti berikut

```
SELECT * FROM news ORDER BY id_berita DESC LIMIT 0, 3;
```

Yang artinya pilih semua dari news urutkan secara descending **MULAI/BATASI dari record 0** **tampilkan sebanyak 3 baris**.

Sedangkan untuk page 1 nilai dari \$record adalah  $1 * 3 = 3$ . Sehingga klausa limitnya menjadi seperti berikut

```
SELECT * FROM news ORDER BY id_berita DESC LIMIT 3, 3;
```

Jadi artinya pilih semua dari news urutkan secara descending **MULAI dari record 3** **tampilkan sebanyak 3 baris**. Jika jumlah record hanya 4 yaitu 0, 1, 2, 3. Maka yang ditampilkan sebanyak 1 baris bukan 3 baris. Karena memang jumlah dari record 3 sebanyak 1 baris. Sehingga untuk menampilkan berita kita menggunakan query seperti berikut

```
$hasil = mysql_query("SELECT * FROM news ORDER BY id_berita DESC LIMIT $record,
$berita_per_hal");
```

Seperti biasa kita menampilkan hasil dengan looping while. Pada blok kode while kita membuat variabel \$isi\_berita yang isinya adalah isi dari data['isi\_berita'] yang karakter garis baru (\n) kita ganti dengan tag html <br>. Ini dikarenakan HTML tidak mendukung karakter garis baru \n maka kita menggantinya dengan karakter garis baru berformat HTML yaitu tag <br>.

Kita tiba pada bagian yang cukup menarik yaitu mencuplik berita. Pertama kita membuat variabel \$cuplikan yang beritipe array. Untuk sementara file ini tidak ada nilainya. Kemudian kita memecah setiap kata pada berita menggunakan fungsi explode(). Tanda pemisah antar kata tentu adalah spasi. Setelah itu kita melakukan looping for untuk memberi nilai pada variabel \$cuplikan. Looping ini kita tentukan sebanyak 25 kali. Artinya 25 kata pertamalah yang kita ambil.

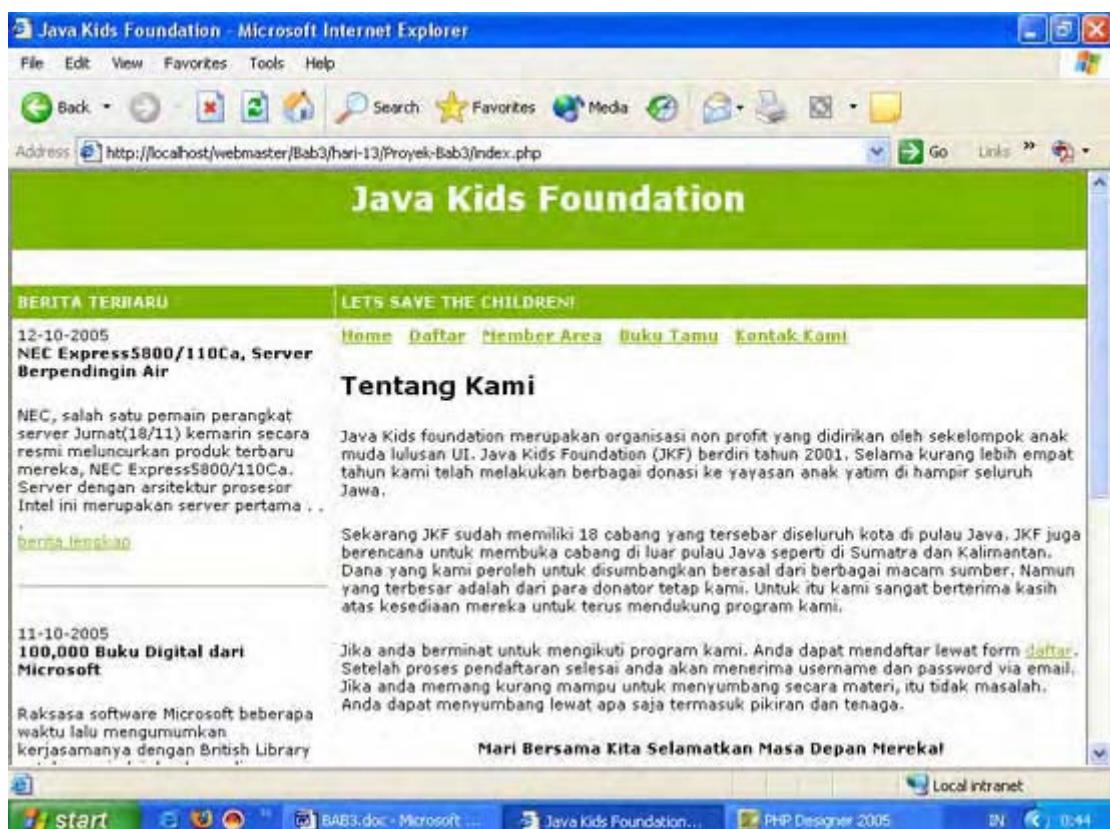
```
for ($i=0; $i<25; $i++)
 $cuplikan[$i] = $pecahan_kata[$i];
```

Jadi pada eksekusi pertama isi dari \$cuplikan[0] sama dengan \$pecahan\_kata[0] yang tidak lain adalah kata pertama dari isi berita. Pada eksekusi ke dua isi \$cuplikan[1] sama dengan \$pecahan\_kata[1] yang artinya sama dengan kata ke dua. Demikian seterusnya hingga \$cuplikan[24]. Setelah itu kita tambahkan link untuk membaca berita secara lengkap.

Agar dapat ditampilkan maka array \$cuplikan harus digabungkan dulu. Untuk menggabungkannya kita menggunakan fungsi `implode()`. String peng gabung yang kita gunakan adalah spasi. Agar kita mendapatkan nilai dari \$page dan dapat memilih halaman berita maka kita perlu membuat link URL yang terdapat variabel \$page didalamnya.

```
if ($jml_hal > 1)
{
 $news .= 'Halaman: ';
 for ($i=0; $i<$jml_hal; $i++)
 $news .= ' ' . intval($i+1)
 .' ';
}
```

Pada cuplikan kode diatas link halaman hanya muncul jika jumlah dari \$jml\_hal lebih dari satu. Yang artinya setidaknya ada halaman 0 dan 1. Karena angka 0 dari suatu halaman terkesan "aneh". Maka kita menambahkannya satu agar yang muncul pertama adalah Halaman: 1 dan seterusnya. Jika kita langsung menambahkan \$i+1 maka akan error karena akan dianggap string oleh PHP. Oleh karena itu kita gunakan fungsi `intval()`. Untuk membuktikannya masuklah pada admin area lalu tambah beberapa berita hingga jumlah beritanya lebih dari 3.



Gambar 3.22: Tampilan utama javakidsfoundation.localhost

Penjelasan dari file index.php cukup menguras otak. File berikutnya adalah file `bukutamu.php`. Cara kerja file ini hampir sama dengan file index.php. Hanya saja file `bukutamu.php` memiliki beberapa proses. Klik menu **File-New-PHP** lalu ketik kode berikut

Script dari file **bukutamu.php**

```
<?php
include('template/template.php');
include('template/index_var.php');
```

```

/* variabel-variabel untuk halaman bukutamu */
/*****************/
$bukutamu = $_GET['bukutamu_id'];
if ($bukutamu == '')
 $bukutamu = 'view';

$page = $_GET['page'];
if ($page == '')
 $page = 0;
// cegah cross site scripting
$bukutamu = filter_str($bukutamu);
$page = filter_str($page);
// tentukan komentar yang ditampilkan per halaman
$komen_per_hal = 3;
// lakukan query lalu tentukan jumlah halaman dengan ceil()
koneksi_db(); // koneksi ke MySQL server
$query_hal = mysql_query("SELECT * FROM bukutamu");
$jml_baris = mysql_num_rows($query_hal);
$jml_hal = ceil($jml_baris / $komen_per_hal);
// dapatkan nilai offset
$offset = $page * $komen_per_hal;

switch ($bukutamu)
{
case 'view':
 // lakukan query untuk mendapatkan data dari bukutamu
 $query = "SELECT * FROM bukutamu ORDER BY id_bt DESC LIMIT $offset,
 $komen_per_hal";
 $hasil = mysql_query($query);
 mysql_close(); // tutup koneksi
 //tampilkan bukutamu
 $page_bt = '<h3>Buku Tamu</h3>
 . '<p>Belum mengisi buku tamu, silahkan klik '
 . 'di sini untuk
mengisi.</p>'
 . '<p><table border="0" cellpadding="4">';
 while ($data = mysql_fetch_array($hasil))
 {
 $komentar = str_replace("\n", "
", $data['komentar']);
 $page_bt .= '<tr bgcolor="#7cb500" class="putih">
 .<td colspan="2">
 .<td>$data['tgl_post'].</td></tr>
 .<tr><td align="right">Nama: </td>
 .<td>$data['nama'].</td></tr>
 .<tr><td align="right">Email: </td>
 .<td>$data['email'].</td></tr>
 .<tr><td valign="top">Komentar: </td>
 .<td colspan="2">.$komentar.</td></tr>
 .<tr><td height="10"></td></tr>'; // beri jarak
 }
 $page_bt .= '</table></p>';
 // tampilkan link halaman jika jumlahnya lebih dari 1
 if ($jml_hal > 1)
 {
 $page_bt .= 'Halaman: ';
 for ($i=0; $i<$jml_hal; $i++)
 $page_bt .= ' ' . ' ';
 }
break; // akhir dari proses view

case 'form':
 $page_bt = '<h3>Form Buku Tamu</h3>

```

```

 . 'Silahkan mengisi form buku tamu berikut. Mohon komentar tidak '
 . 'mengandung unsur SARA!.'
 . '<form action="bukutamu.php?bukutamu_id=proses_form"
method="post">'
 . '<table border="0" cellpadding="4">'
 . '<tr bgcolor="#7cb500" class="putih">'
 . ' <td colspan="2" align="center">Form Buku Tamu</td> </tr>'
 . '<tr><td>Nama: </td>'
 . ' <td><input type="text" name="nama"></td></tr>'
 . '<tr><td>Email: </td>'
 . ' <td><input type="text" name="email"></td></tr>'
 . '<tr><td>Komentar: </td>'
 . ' <td><textarea name="komentar" rows="8" cols="45">
</textarea></td></tr>'
 . '<tr><td><input type="submit" value="KIRIM"></td></tr>'
 . '<tr bgcolor="#7cb500" height="15"><td colspan="2" > </td></tr>'
 . '</table></form>';
break; // akhir dari proses form

case 'proses_form':
$nama = $_POST['nama'];
$email = $_POST['email'];
$komentar = $_POST['komentar'];

$pesan_error = '';
// cek apakah ada field yang kosong
if (!cek_field($_POST))
 $pesan_error = 'Error: masih ada field yang kosong
';
// cek kevalidan format email
if (!cek_email($email))
 $pesan_error .= 'Error: email tidak valid
';
// cek kata-kata kotor, ganti dengan !@#$%
$kotor = array('fuck', 'bangsat', 'bullshit', 'brengsek', 'ass');
foreach ($kotor as $kata)
 $komentar = eregi_replace($kata, '!@#$%', $komentar);

//jika isi $pesan_error tidak kosong maka ada error
if ($pesan_error != '')
 $page_bt = $pesan_error.$kembali;
else
{
//lakukan query
koneksi_db();
$tgl = date('G:i, d-m-Y'); // Format HH:MM, DD-MM-YYYY
$query = "INSERT INTO bukutamu (tgl_post, nama, email, komentar) VALUES
('$tgl', '$nama', '$email', '$komentar')";
$hasil = mysql_query($query);
if (!$hasil)
 $page_bt = 'Error: gagal memasukkan data ke database.';
else
{
 $page_bt = 'Data anda berhasil dimasukkan ke database.'
 . ' Lihat buku tamu.';
} // akhir dari else ke dua
} // akhir dari else pertama
break; // akhir daro proses proses_form
} //akhir dari switch

$skin = new skin;
$skin->ganti_skin('template/java_skin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{UTAMA}', $page_bt);
$skin->ganti_tag('{NEWS}', $side_news);

```

```

$skin->ganti_tag('{FORUM}', $forum);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **Proyek-Bab3** dengan nama **bukutamu.php**. Output dari file ini akan terlihat seperti gambar 3.23

### PENJELASAN SCRIPT

Untuk menampilkan komentar per halaman dan masalah page tidak perlu saya jelaskan lagi disini. Karena pada penjelasan tentang file *index.php* kita sudah membahasnya panjang lebar.



Gambar 3.23: Output file bukutamu.php

Karena file *bukutamu.php* menghandel banyak proses maka kita gunakan switch dan case untuk menangani setiap kondisi. Kondisi yang dihandel adalah isi dari bukutamu\_id pada URL.

Pada case yang pertama yaitu *view*. Cara kerjanya sama dengan yang pernah kita tuliskan pada file *index.php*. Case yang kedua yaitu *form*. Pada case ini tidak ada yang spesial hanya menampilkan form isian untuk buku tamu.

Case yang berikutnya yaitu *proses\_form*. Kode pada case ini bertugas memproses data yang dikirm dari form isian buku tamu. Selanjutnya kita membuat variael **\$pesan\_error**. Variabel ini tidak berisi apa-apa. Namun jika nanti terdapat error otomatis nilai dari variabel ini akan diubah.

Seperti biasa sebelum memasukkan ke database kita cek dulu semua field apa ada yang kosong. Untuk itu kita gunakan fungsi *cek\_field()* yang akan mengecek semua field. Jika ada field yang belum terisi maka ubah isi dari variabel **\$pesan\_error** seperti berikut

```
$pesan_error = 'Error: masih ada field yang kosong
';
```

Jika email juga tidak valid maka tambah isi dari **\$pesan\_error** dengan menggunakan tanda **.=** (titik sama dengan). Agar isi dari error sebelumnya tidak tertindih. Seperti yang pernah kita

---

lakukan pada bab 1. Kita mengecek kata-kata kotor dan mengantinya dengan karakter !@#\$%. Anda dapat menambahkan daftar dari kata-kata kotor tersebut.

Selanjutnya kita mengecek isi dari variabel \$pesan\_error. Jika isinya tidak kosong (!=) maka ada error. Untuk itu isi dari \$page\_bt kita samakan dengan \$pesan\_error. Selain itu proses data dan masukkan ke database. Jika pada proses pemasukan data ke database terdapat error maka tampilkan pesan bahwa data gagal dimasukkan ke database. Selain itu tampilkan pesan berhasil.

File berikutnya yang akan kita buat adalah *daftar.php*. Tugas file ini adalah untuk mendaftarkan user menjadi member.

#### Script dari file *daftar.php*

---

```
<?php

include('template/template.php');
include('template/index_var.php');

/* variabel-variabel untuk halaman daftar */
/*****************/
koneksi_db(); // koneksi ke MySQL
$daftar = $_GET['daftar_id'];
if ($daftar == '')
 $daftar = 'form';
// cegah Cross site scripting
$daftar = filter_str($daftar);

switch ($daftar)
{
case 'form': // jika daftar_id = form eksekusi kode berikut
$page_daftar .= '<h3>Mohon isi Form berikut</h3>
.' '<form action="daftar.php?daftar_id=proses_form" method="post">
.' '<table border="0" cellpadding="4">
.' '<tr bgcolor="#7cb500" class="putih">
.' '<td colspan="2" align="center">Form Pendaftaran</td></tr>
.' '<tr><td>Username: </td>
.' '<td><input type="text" name="username" maxlength="16">
.' ' max. 16 karakter</td></tr>
.' '<tr><td>Password: </td>
.' '<td><input type="password" name="password" maxlength="16">
.' ' max. 16 karakter</td></tr>
.' '<tr><td>Nama Lengkap: </td>
.' '<td><input type="text" name="nama" maxlength="30"></td></tr>
.' '<tr><td>Email: </td>
.' '<td><input type="text" name="email" maxlength="30"></td></tr>
.' '<tr><td>Alamat: </td>
.' '<td><input type="text" name="alamat" size="50"></td></tr>
.' '<tr><td>Kode Pos: </td>
.' '<td><input type="text" name="kodepos" maxlength="6" size="6"></td></tr>
.' '<tr><td>Kota: </td>
.' '<td><input type="text" name="kota" maxlength="20"></td></tr>
.' '<tr><td><input type="submit" value="DAFTAR"></td></tr>
.' '<tr bgcolor="#7cb500" height="15"><td colspan="2"></td></tr>
.' '</table>
.' '</form>';
break; // akhir dari proses form

case 'proses_form': // jika daftar_id = proses_form eksekusi berikut
$username = $_POST['username'];
$password = $_POST['password'];
$nama = $_POST['nama'];
$email = $_POST['email'];
```

```

$alamat = $_POST['alamat'];
$kodepos = $_POST['kodepos'];
$kota = $_POST['kota'];

$pesan_error = '';
// sebelum memasukkan ke database cek setiap field dulu
// panggil fungsi cek_field() untuk mengecek field
if (!cek_field($_POST))
 $pesan_error = 'Error: masih ada field yang kosong
';

// cek username apakah mengandung karakter aneh
if (ereg('[^a-zA-Z0-9_]', $username))
 $pesan_error .= 'Error: username hanya boleh terdiri dari huruf, angka dan

';

// cek password
if (ereg('[^a-zA-Z0-9_]', $password))
 $pesan_error .= 'Error: password hanya boleh terdiri dari huruf, angka dan

';

// cek email
if (!cek_email($email))
 $pesan_error .= 'Error: email tidak valid
';

// cek apakah username sudah terpakai atau belum
$q_cek = mysql_query("SELECT * FROM member WHERE username='$username'");
if (mysql_num_rows($q_cek) > 0)
 $pesan_error .= 'Error: username sudah terpakai, silahkan pilih yang
lain
';

// cek isi dari $pesan_error jika tidak sama dengan '' maka ada error
if ($pesan_error != '')
 $page_daftar = $pesan_error.$kembali;
else
{
 $query = "INSERT INTO member VALUES ('$username', '$password', '$nama',
'$email', '$alamat', '$kodepos', '$kota')";
 $hasil = mysql_query($query);
 mysql_close(); // tutup koneksi

 if (!$hasil)
 $page_daftar = 'Error: gagal memasukkan data ke database. Mohon coba lagi
nanti!
'.$kembali;
 else
 {
 $page_daftar = 'Registrasi sukses. Data registrasi juga kami kirim ke email
anda,
 .'silahkan dicek.

Klik Login untuk '
 .'masuk ke member area.';

 // buat variabel untuk proses pengiriman email
 $to = $email; // email dari user yang baru mendaftar
 $subject = 'Account anda di javafoundation.com';
 $isi_email = "Administrator javafoundation.com\n\n"
 ."Saudara/i $nama, terima kasih telah menjadi member dari website
 kami.\n"
 ."Berikut ini adalah username dan password untuk login di member
 area\n\n"
 .". =====\n"
 .". Username: $username\n"
 .". Password: $password\n"
 .". =====\n"
 .". Mari bersama selamat masa depan anak-anak Indonesia.";
 $from = 'From: admin@javafoundation.com';

```

```

// jika tidak memiliki mail server atau belum di server sebenarnya
// beri komentar pada fungsi mail() dibawah ini
mail($to, $subject, $isi_email, $from);
} // akhir dari else ke dua
} // akhir dari else pertama
break; // akhir dari proses_proses_form
} // akhir dari switch

mysql_close(); // tutup koneksi

$skin = new skin;
$skin->ganti_skin('template/java_skin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{UTAMA}', $page_daftar);
$skin->ganti_tag('{NEWS}', $side_news);
$skin->ganti_tag('{FORUM}', $forum);
$skin->ganti_tampilan();

?>

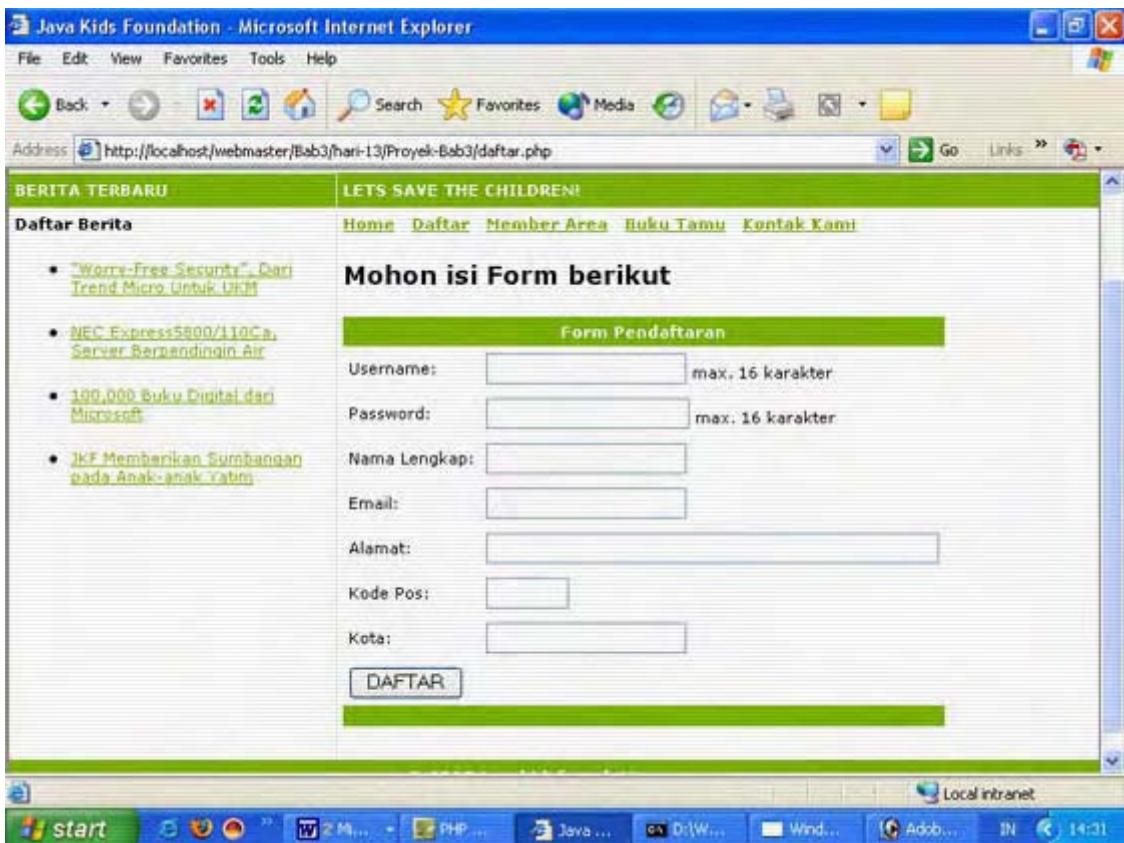
```

Simpan pada direktori **Proyek-Bab3** dengan nama **daftar.php**. Output dari file ini ditunjukkan oleh gambar 3.24.

### PENJELASAN SCRIPT

File *daftar.php* juga memiliki beberapa proses. Yaitu proses *form* untuk menampilkan form pengisian data pribadi. Dan *proses\_form* untuk memproses data yang dikirim dari form data pribadi. Untuk selebihnya tidak ada perbedaan dengan file-file sebelumnya.

Namun karena ini berhubungan dengan registrasi maka sudah selayaknya kita mengirimkan username dan password ke alamat si pendaftar. Jika kita belum terhubung ke internet kita belum bisa mengetes hasil yang sebenarnya. Oleh karena itu untuk sementara kita tangguhkan dulu fungsi *mail()* dengan memberi tanda komentar di depannya.



Gambar 3.24: tampilan daftar.php

---

File berikutnya adalah *kontak.php*. Fungsi dari file ini adalah untuk mengirimkan pesan ke email kita lewat form kontak kami. Untuk membuatnya klik **File-New-PHP** pada PHP DESIGNER 2006 lalu ketik kode berikut

Script dari file ***kontak.php***.

```
<?php

include('template/template.php');
include('template/index_var.php');

/* variabel-variabel untuk halaman kontak kami */
/************* */
$kontak = $_GET['kontak_id'];
if ($kontak == '')
 $kontak = 'form';
//cegah cross site scripting
$kontak = filter_str($kontak);

switch ($kontak)
{
case 'form':
 $page_kontak = '<h3>Kontak Kami</h3>
 .<p>Jika anda mempunyai pertanyaan seputar Java Kids Foundation.
Silahkan isi form kontak dibawah ini. Kami akan merespon pertanyaan anda secepat
mungkin.</p>'
 . '<form action="kontak.php?kontak_id=proses_form" method="post">'
 . '<table border="0" cellpadding="4">
 . '<tr bgcolor="#7cb500" class="putih">
 .<td colspan="2" align="center">Form Kontak</td> </tr>'
 . '<tr><td>Nama: </td>
 .<td><input type="text" name="name"></td></tr>'
 . '<tr><td>Email: </td>
 .<td><input type="text" name="email"></td></tr>'
 . '<tr><td>Subject: </td>
 .<td><input type="text" name="subject" size="50"> </td></tr>'
 . '<tr><td>Pertanyaan: </td>
 .<td><textarea name="pertanyaan" rows="8" cols="50">
 .</textarea></td></tr>
 . '<tr><td><input type="submit" value="KIRIM"> </td></tr>
 . '<tr bgcolor="#7cb500" height="15"><td colspan="2" > </td></tr>
 . '</table></form>';
break; // akhir dari proses form

case 'proses_form':
 $nama = $_POST['nama'];
 $email = $_POST['email'];
 $subject = $_POST['subject'];
 $pertanyaan = $_POST['pertanyaan'];

 $pesan_error = '';
 if (!cek_field($_POST))
 $pesan_error = 'Error: masih ada field yang kosong
';
 // cek email
 if (!cek_email($email))
 $pesan_error .= 'Error: email tidak valid

 if ($pesan_error != '')
 $page_kontak = $pesan_error.$kembali;
 else
 {
 $to = 'admin@rahasia-webmaster.com'; // email anda
 $from = 'From: '.$email; // email si pengirim
```

```

 // jika anda tidak memiliki mail server atau anda belum di server
 // sebenarnya beri komentar pada fungsi mail berikut
 mail($to, $subject, $pertanyaan, $from);
 $page_kontak = 'Pertanyaan anda telah terkirim. Terima Kasih.';
}
break; // akhir dari proses proses_from
}

$skin = new skin;
$skin->ganti_skin('template/java_skin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{UTAMA}', $page_kontak);
$skin->ganti_tag('{NEWS}', $side_news);
$skin->ganti_tag('{FORUM}', $forum);
$skin->ganti_tampilan();
?>

```

---

Simpan pada direktori **Proyek-Bab3** dengan nama ***kontak.php***. Tidak ada yang menarik dari file ini. Sama seperti sebelumnya untuk sementara fungsi `mail()` kita beri tanda komentar dulu agar tidak dieksekusi. Setelah terhubung ke internet hilangkan tanda komentar pada fungsi tersebut.

Untuk itu kita langsung saja membuat file berikutnya yaitu *news.php*. Fungsi dari file ini adalah untuk menampilkan berita secara lengkap sesuai dengan judul berita yang diklik user pada samping halaman. Klik **File-New-PHP** lalu ketik kode berikut

Script dari file ***news.php***

---

```

<?php

include('template/template.php');
include('template/index_var.php');
koneksi_db(); // koneksi ke MySQL server

/* variabel-variabe untuk menampilkan berita */
/**/

/* tampilkan berita secara lengkap */
$berita = $_GET['berita'];
//cegah Cross Site Scripting
$berita = filter_str($berita);
// lakukan query untuk mendapatkan berita selengkapnya
$hasil = mysql_query("SELECT * FROM news WHERE id_berita = '$berita'");
$data = mysql_fetch_array($hasil);
// nl2br() fungsi alternatif untuk mengganti \n menjadi

// gunakan stripslashes() untuk menghilangkan escaping character
$isi_berita = nl2br(stripslashes($data['isi_berita']));
$full_news = $data['tgl_berita']. '
' .
 ''.stripslashes($data['jdl_berita']).'
' .
 '<p>'.$isi_berita.'</p>';

mysql_close(); // tutup koneksi

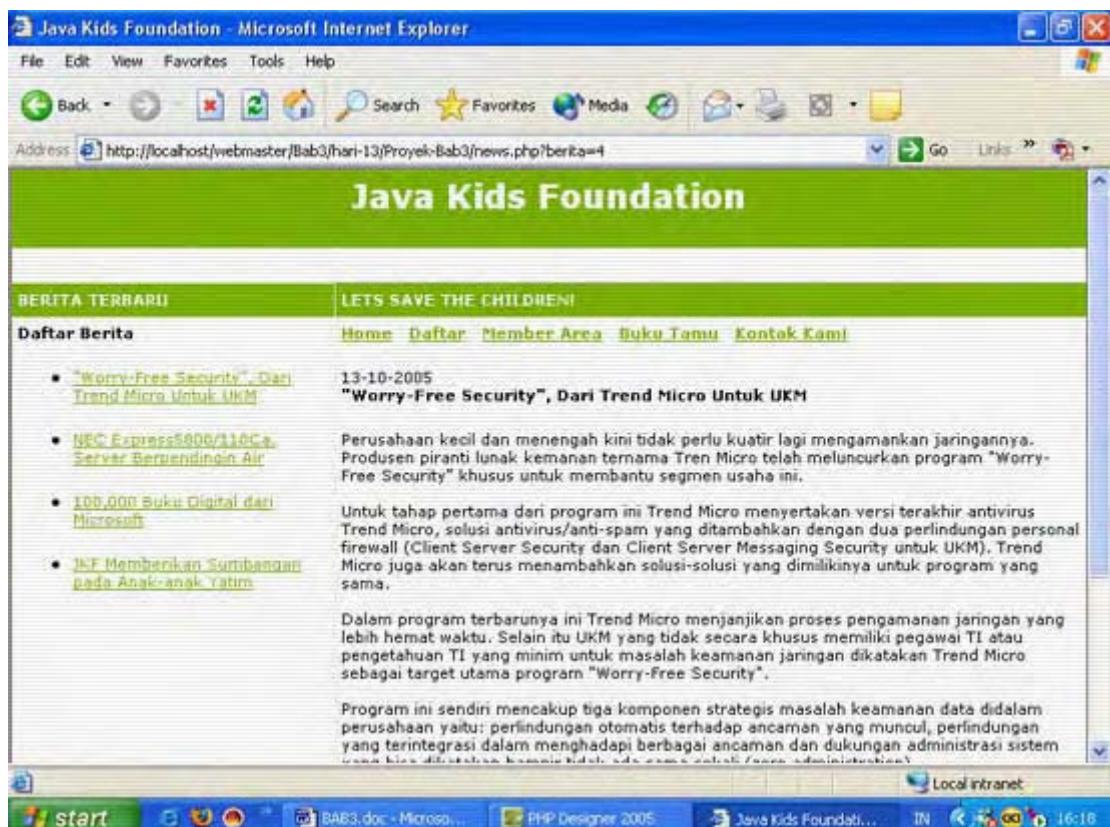
$skin = new skin;
$skin->ganti_skin('template/java_skin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{UTAMA}', $full_news);
$skin->ganti_tag('{NEWS}', $side_news);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **Proyek-Bab3** dengan nama **news.php**. Output dari file ini ditunjukkan oleh gambar 3.25.

Cara kerja file ini sama dengan yang lainnya yaitu mendapatkan data dari database lalu menampilkannya. Namun ada fungsi yang jarang kita pakai namun cukup berguna yaitu `n12br()`. Fungsi ini akan mengganti karakter garis baru `\n` menjadi format HTML yaitu `<br>`.



Gambar 3.25: Halaman berita secara lengkap

Selanjutnya kita akan membuat file untuk login ke member area. File ini memiliki dua proses, yaitu proses *form* dan *proses\_form*. Proses *form* digunakan untuk menampilkan form login. Sedangkan *proses\_form* untuk memproses form login dan membuat session baru. Klik **File-New-PHP** pada PHP DESIGNER 2006 lalu ketik kode berikut

Script dari file **login.php**

```
<?php
// panggil session_start() karena berhubungan dengan session
session_start();

include('template/template.php');
include('template/index_var.php');

/* variabel-variabel untuk halaman login */
/************* */
$login = $_GET['login_id'];
if ($login == '')
 $login = 'form';
// cegah Cross Site Scripting
$login = filter_str($login);

switch ($login)
{
```

```

case 'form': // jika login_id = form maka eksekusi kode berikut
 $page_login = '<h3>Login Page</h3>
 . '<p>Untuk masuk ke member area silahkan login terlebih dahulu.</p>'
 . '<form action="login.php?login_id=proses_login" method="post">
 .<table border="0" cellpadding="4">
 .<tr bgcolor="#7cb500">
 .<td colspan="2" class="putih" align="center">Login Di
 sini</td></tr>
 .<tr><td>Username: </td>
 .<td><input type="text" name="username"></td></tr>
 .<tr><td>Password: </td>
 .<td><input type="password" name="password"></td></tr>
 .<tr><td colspan="2" align="center">
 .<input type="submit" value="Login"></td></tr>
 .<tr bgcolor="#7cb500" height="15"><td colspan="2"></td></tr>
 .</table>
 .</form>
 .<p>Lupa password? klik di sini.</p>';
break; // akhir dari proses form

case 'proses_login': // jika login_id = proses_login eksekusi berikut
 // cegah SQL injection
 $username = filter_str($_POST['username']);
 $password = filter_str($_POST['password']);

 koneksi_db(); // koneksikan ke MySQL server

 // panggil fungsi login()
 if (!login('member', $username, $password))
 $page_login = 'Username atau password salah.
'.kembali;
 else
 {
 $_SESSION['member'] = $username; // buat session bernama member
 $page_login = 'Login berhasil. Silahkan klik '
 .'disini untuk masuk ke member area.';
 }
 mysql_close(); // tutup koneksi
break; // akhir dari proses proses_login
}

$skin = new skin;
$skin->ganti_skin('template/java_skin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{UTAMA}', $page_login);
$skin->ganti_tag('{NEWS}', $side_news);
$skin->ganti_tag('{FORUM}', $forum);
$skin->ganti_tampilan();
?>
```

Simpan pada direktori **Proyek-Bab3** dengan nama **login.php**. Tampilan dari file ini akan terlihat seperti gambar 3.26.

### PENJELASAN SCRIPT

Karena kita nantinya akan membuat session baru untuk pada file ini. Maka kita panggil fungsi **session\_start()**. Pada case *form* kita hanya menampilkan form login. Yang menampilkan dua textbox yaitu textbox untuk username dan textbox untuk password.

Pada case *proses\_form* kita mengambil data yang dipost dari form login. Lalu kita memanggil fungsi **login()**. 'member' pada parameter pertama pada fungsi **login()** yang kita tulis menunjukkan tabel yang kita gunakan yaitu tabel member. Jika tidak ada record yang cocok tampilkan pesan bahwa username atau password salah.

Selain itu berarti username dan password yang dimasukkan cocok dengan yang ada di database. Untuk itu buarkan session baru bernama *member* yang isinya adalah username dari user yang berhasil login.



Gambar 3.26: Form login untuk member

File terakhir pada direktori *Proyek-Bab3* adalah *lupa\_pass.php*. File berguna untuk mengirimkan password ke alamat user yang lupa akan passwordnya. Untuk membuatnya klik menu **File-New-PHP**, kemudian ketik kode berikut

Script dari file *lupa\_pass.php*

```
<?php

include('template/template.php');
include('template/index_var.php');

$id = $_GET['id'];
if ($id == '')
 $id = 'form';
// cegah cross site Scripting
$id = filter_str($id);

switch ($id)
{
case 'form': // titik dua bukan titik koma
 $page_lupa = '<h3>Lupa Password</h3>
 . '<p>Masukkan email anda yang ada pada member area '
 . 'javakidsfoundation.localhost</p>
 . '<form action="lupa_pass.php?id=kirim" method="post">'
 . '<table border="0" cellpadding="4">
 . '<tr bgcolor="#7cb500" class="putih">
 . '<td colspan="2" align="center">Form Lupa Password </td></tr>
 . '<tr><td>Email: </td>
 . '<td><input type="text" name="email" maxlength="30"> '
 . '</td></tr>'
```

---

```

 . '<tr><td><input type="submit" value="KIRIM"> </td></tr>'
 . '<tr bgcolor="#7cb500" height="15"><td colspan="2"> </td></tr>'
 . '</table></form>';
 break; // akhir dari proses form

case 'kirim':
 $email = $_POST['email'];
 // cek email
 if (!cek_email($email))
 $page_lupa = 'Error: email tidak valid
' . $kembali;
 else
 {
 //lakukan query untuk mengecek email apa ada di database
 koneksi(); // koneksi ke MySQL server
 $hasil = mysql_query("SELECT * FROM member WHERE email='$email'");
 mysql_close(); // tutup koneksi
 if (mysql_num_rows($hasil) == 0)
 $page_lupa = 'Error: email '.$email.' tidak ada di database.
' .
 . $kembali;
 else
 {
 $data = mysql_fetch_array($hasil);
 // buat variabel untuk digunakan pada fungsi mail()
 $to = $email;
 $subject = 'Password Anda di javakidsfoundation.localhost';
 $isi_email = "Berikut ini adalah username dan password anda: \n"
 . "===== \n\n"
 . "Username: ".$data['username']."\n"
 . "Password: ".$data['password']."\n\n"
 . "Harap ingat baik-baik password anda tersebut. Jika diperlukan
catatlah password tersebut.\n\n"
 . "www.jakidsfoundation.com";
 $from = 'From: admin@javakidsfoundation.localhost';

 // jika anda tidak memiliki mail server atau anda belum di server sebenarnya
 // beri komentar pada fungsi mail berikut
 mail($to, $subject, $isi_email, $from);
 $page_lupa = 'Password berhasil dikirim. Silahkan cek email anda';
 } // akhir dari else ke dua
 } // akhir dari else pertama

break; // akhir dari proses kirim
} // akhir dari switch

$skin = new skin;
$skin->ganti_skin('template/java_skin.php');
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{UTAMA}', $page_lupa);
$skin->ganti_tag('{NEWS}', $side_news);
$skin->ganti_tag('{FORUM}', $forum);
$skin->ganti_tampilan();
?>
```

---

Simpan pada direktori **Proyek-Bab3** dengan nama ***lupa\_pass.php***.

## PENJELASAN SCRIPT

File ini miliki dua proses yaitu *form* dan *kirim*. *form* untuk menampilkan textbox email. Dan *kirim* untuk memproses sekaligus mengirim email ke alamat email user tersebut.

---

Pada case *kirim* kita mengambil data yaitu email yang dimasukkan user pada textbox. Untuk kemudian kita cocokkan apakah email tersebut ada didatabase atau tidak. Kita mengeceknya menggunakan fungsi `mysql_num_rows()`.

```
if (mysql_num_rows($hasil) == 0)
```

Jika jumlah baris yang dikembalikan query SQL sama dengan nol berarti email tidak ada di database. Untuk itu tampilkan pesan error. Namun jika email ada di database pecah setiap field atau kolom pada record tersebut untuk mendapatkan username dan password.

Setelah itu kita masukkan username dan password pada isi dari email kita untuk dikirimkan. Jika anda tidak memiliki program mail server (contoh: hMailServer) atau anda belum mengupload ke server sebenarnya. Sebaiknya fungsi `mail()` kita tangguhkan agar PHP tidak memunculkan pesan error.

Akhir dari file yang berada pada direktori **Proyek-Bab3**

---

File-file yang akan kita buat berikutnya adalah untuk member area. File untuk member area hanya terdiri dari dua file. Yaitu file utama *index.php* dan file untuk mengupdate profil *profil.php*. Klik menu **File-New-PHP** pada PHP DESIGNER 2006 untuk membuat file PHP baru, kemudian ketik kode berikut

Script dari file *index.php*

---

```
<?php
// panggil fungsi session_start()
session_start();

include('../template/template.php');
include('../template/member_var.php');

if (!cek_session('member'))
{
 $menu = '';
 $utama = 'Anda belum login. Silahkan Login dulu.';
}
else
{

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';
// cegah cross site scripting
$proses = filter_str($proses);

switch ($proses)
{
case 'view':
 $utama = '<h3>Selamat Datang di Member Area</h3>
 .'.'Anda login sebagai: '.$_SESSION['member']
 .'<p> Halaman utama member area </p>';
break; // akhir dari proses view

case 'logout':
 if (!logout('member'))
 $utama = 'Tidak bisa logout. Login dulu.';
 else
 $utama = 'Anda telah logout dari sistem. Klik '
 .'di sini untuk login kembali.';
break; // akhir dari proses logout
} //akhir dari switch
```

---

```

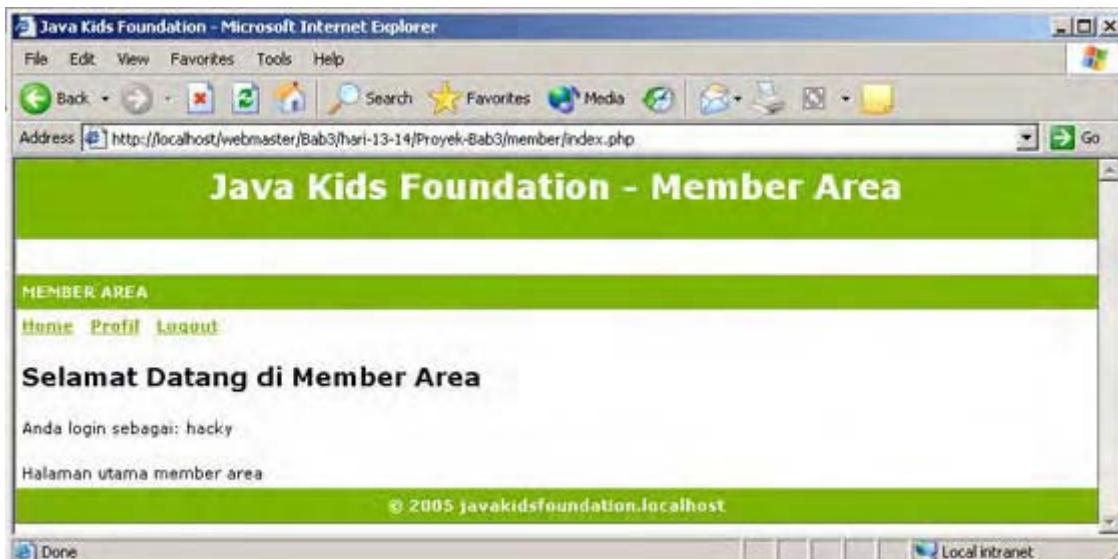
} // akhir dari else

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tampilan();
?>

```

---

Simpan pada direktori **Proyek-Ba3\member\** dengan nama **index.php**. Output dari file ini ditunjukkan oleh gambar 3.27.



Gambar 3.27: Halaman utama member area

### PENJELASAN SCRIPT

Pertama kita akan mengecek dulu apakah session member telah terisi atau belum dengan menggunakan fungsi `cek_session('member')`. Jika ternyata session member belum terisi ganti isi dari variabel `$menu` menjadi kosong. Dan variabel `$utama` kita isi dengan pesan bahwa user belum login.

Jika user telah login maka blok pada else yang akan dieksekusi. Pada blok kode else ini ada dua proses yaitu proses *view* dan *logout*. Pada case *view* kita menampilkan halaman utama member area. Pada *logout* kita menghapus isi dari session member dan menghendurkannya dengan `session_destroy()`.

File berikutnya yaitu file *profil.php*. File ini berfungsi untuk menyediakan form update dan memprosesnya. Klik File-New-PHP untuk membuat file baru kemudian ketik kode berikut

Script dari file ***profil.php***

---

```

<?php

// mulai session
session_start();

include('../template/template.php');
include('../template/member_var.php');

// cek apakah user sudah login atau belum
if (!cek_session('member'))

```

```

{
 $menu = '';
 $profil= 'Anda belum login. Silahkan Login dulu.';
}
else
{
 $proses = $_GET['proses'];
 if ($proses == '')
 $proses = 'view';
 // cegah cross site scripting
 $proses = filter_str($proses);

 switch ($proses)
 {
case 'view':
 // lakukan query database untuk mendapatkan profil
 $user = $_SESSION['member']; // user yang sedang login
 koneksi_db(); //koneksi ke MySQL Server
 $hasil = mysql_query("SELECT * FROM member WHERE username='$user'");
 $data = mysql_fetch_array($hasil);
 mysql_close(); // tutup koneksi

 $profil = '<h3>Update Profil</h3>
 '<p>Silahkan update data pribadi anda. Tekan UPDATE untuk
memproses.</p>
 '<form action="profil.php?proses=update" method="post">
 '<table border="0" cellpadding="4">
 '<tr bgcolor="#7cb500" class="putih">
 '<td colspan="2" align="center">Form Update</td>
 '<tr><td>Username: </td>
 '<td>'. $data['username']. '</td>
 '<tr><td>Password: </td>
 '<td><input type="password" maxlength="16" name="password"
value="'. $data['password']. '"></td></tr>
 '<tr><td>Nama Lengkap: </td>
 '<td><input type="text" name="nama"
value="'. $data['nama']. '"></td></tr>
 '<tr><td>Email: </td>
 '<td><input type="text" name="email"
value="'. $data['email']. '"></td></tr>
 '<tr><td>Alamat: </td>
 '<td><input type="text" name="alamat" value="'. $data['alamat']. '"'
size="50"></td></tr>
 '<tr><td>Kode Pos: </td>
 '<td><input type="text" name="kodepos" size="7" maxlength="7"
value="'. $data['kodepos']. '"></td></tr>
 '<tr><td>Kota: </td>
 '<td><input type="text" name="kota"
value="'. $data['kota']. '"></td></tr>
 '<tr><td><input type="submit" value="UPDATE"></td></tr>
 '<tr bgcolor="#7cb500"><td colspan="2" height="15"></td></tr>
 '</table></form>';
break; // akhir dari proses view

case 'update':
 $password = $_POST['password'];
 $nama = $_POST['nama'];
 $email = $_POST['email'];
 $alamat = $_POST['alamat'];
 $kodepos = $_POST['kodepos'];
 $kota = $_POST['kota'];

 $pesan_error = '';
 // cek setiap field sebelum memasukkan ke database

```

```

if (!cek_field($_POST))
 $pesan_error = 'Error: masih ada field yang kosong
';
// cek password selain alfabet, numerik dan _ maka Error
if (ereg('[^a-zA-Z0-9_]', $password))
 $pesan_error .= 'Error: password hanya boleh terdiri dari huruf, angka, dan

';
// cek email
if (!cek_email($email))
 $pesan_error .= 'Error: email tidak valid
';

if ($pesan_error != '')
 $profil = $pesan_error.$kembali;
else
{
 // masukkan data ke database
 koneksi_db(); // koneksikan ke MySQL server
 $user = $_SESSION['login']; // user yang sedang login
 $query = "UPDATE member SET password='$password', nama='$nama',
email='$email', alamat='$alamat', kodepos='$kodepos', kota='$kota' WHERE
username='$user'";
 $hasil = mysql_query($query);
 if (!$hasil)
 $profil = 'Error: tidak dapat mengupdate profil ke database.';
 else
 $profil = 'Profil berhasil diupdate. Lihat
Profil';
 } // akhir dari else ke dua
} // akhir dari switch
} // akhir dari else pertama

// ganti tampilan
$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $profil);
$skin->ganti_tampilan();
?>

```

Simpan pada direktori **Proyek-Bab3\member\** dengan nama **profil.php**. Gambar 3.28 menunjukkan output dari ini.

### PENJELASAN SCRIPT

Pada file ini kita memiliki dua proses yaitu proses *view* untuk menampilkan form yang sudah terisi data dari database. Proses yang ke dua yaitu *update*. Proses ini bertugas untuk mengupdate data yang ada didatabase berdasarkan form profil.

Pada case *view* kita mengambil data dari database dari user yang sedang login. User ini tidak lain adalah isi dari session member. Query yang kita gunakan adalah

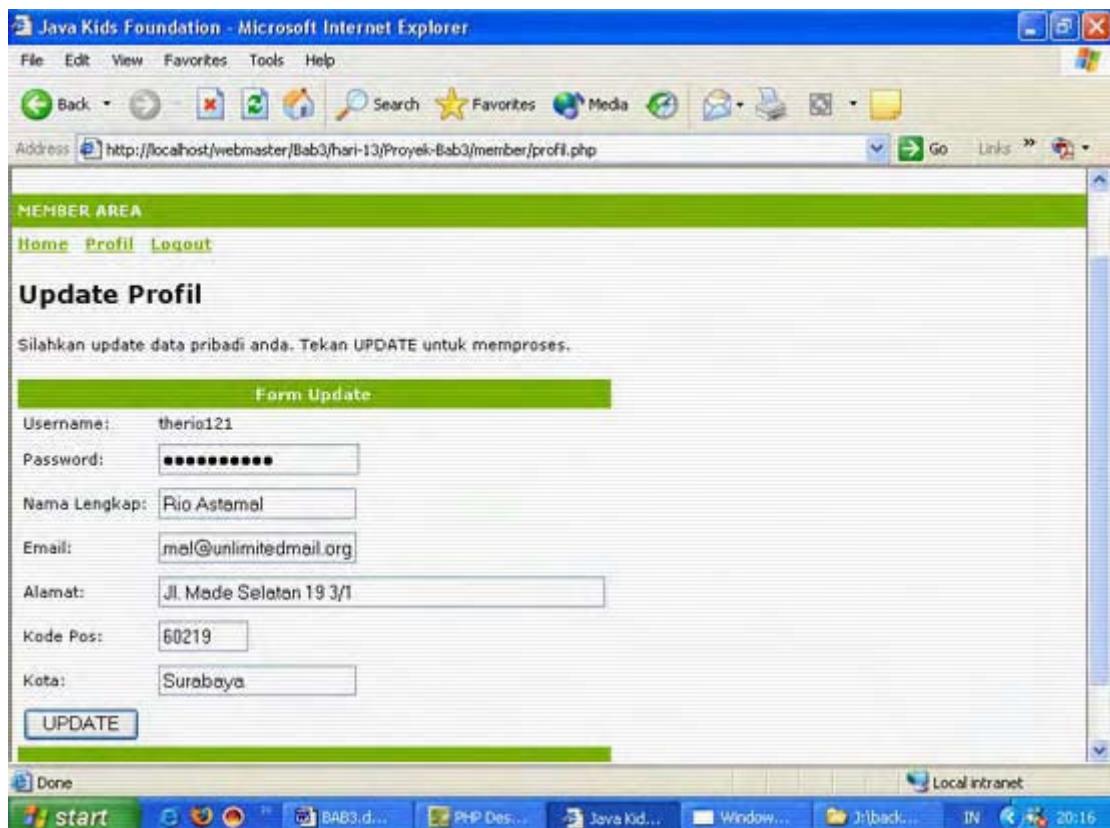
```
$hasil = mysql_query("SELECT * FROM member WHERE username='$user'");
```

Kemudian kita membuat form profil. Untuk memberikan nilai pada setiap field kita menambahkan atribut *value*. Isi dari atribut ini adalah data yang kita peroleh dari database.

Untuk case *update* yang perlu kita perhatikan adalah query untuk melakukan update ke database. Query yang kita gunakan pada kode diatas adalah

```
$query = "UPDATE member SET password='$password', nama='$nama', email='$email',
alamat='$alamat', kodepos='$kodepos', kota='$kota' WHERE username='$user'";
```

Untuk query ini memang cukup panjang karena yang kita update seluruh field. Record yang diupdate adalah record yang field usernamenya adalah username yang sedang login. Tanpa ada klausa WHERE maka seluruh record akan ikut terupdate.



Gambar 3.28: Halaman update profil pada member area.

Akhir dari file yang berada pada direktori **Proyek-Bab3\member\**.

\* \* \* \* \*

## Kalender

27	28	29	30	31	1	2	3
3	4	5	6	7	8	9	10
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

## Berikutnya

Pada bab berikutnya kita akan membahas materi tentang masalah keamanan dan contoh-contoh aplikasi.

# Bab

4

- Masalah Keamanan
- Error Handling
- Contoh-contoh Aplikasi

Sebelum melanjutkan ikuti langkah-langkah berikut:

- Buatlah folder baru pada direktori C:\Apache2\htdocs\webmaster\
- Berinama folder tersebut **Bab4**

\*\*\*\*\*

Seperti biasa karena sudah menginjak hari baru, untuk itu:

- Masuklah pada direktori C:\Apache2\htdocs\webmaster\Bab4\
- Buat folder baru dan beri nama **hari-15**

## Masalah Keamanan

Jika kita bicara masalah keamanan di bidang internet seakan tidak akan ada ujungnya. Ini terbukti jika suatu teknologi baru muncul pasti memiliki *bug* yang dapat dimanfaatkan para cracker untuk membobol sistem. Kali ini kita akan membahas beberapa point masalah keamanan diantaranya:

- Seberapa Penting Informasi Anda?
- Ancaman Keamanan
- Enkripsi Data
- Mengamankan Input dari User
- Secure Login
- Firewall

Materi diatas adalah materi seputar keamanan yang diharapkan anda kuasai dalam satu hari ke depan.

### Seberapa Penting Informasi Anda?

Ketika bicara tentang security, anda harus dapat mengevaluasi pentingnya data yang anda simpan. Anda harus mempertimbangkan hal itu baik untuk anda sendiri maupun si cracker yang mencoba membobol sistem anda.

Memang harus kita yakini tidak ada sistem yang tidak bisa ditembus. Namun sudah tugas kita untuk membuat sistem tersebut sesulit mungkin untuk ditembus. Kita dapat mengeluarkan biaya lebih untuk menjaga keamanan sistem kita. Namun kita harus mengetahui pentingkah informasi ini untuk saya lindungi dengan biaya yang saya keluarkan.

Misalkan data yang disimpan pada komputer seperti kumpulan hobi user, bisnis, bank atau militer tentu mempunyai kepentingan yang berbeda-beda. Cracker tentu memilih target yang menurutnya paling menguntungkan.

Kumpulan hobi user tentu bukan target yang ideal bagi para cracker. Karena informasi yang disimpan pada sistem tersebut tidak begitu penting. Cracker akan menghabiskan banyak usaha untuk menembus sedang informasi yang didapat tidak begitu berguna.

Sedangkan jika informasi yang disimpan menyangkut militer. Tentu nilai dari informasi yang disimpan sangatlah penting. Untuk itu sudah pasti pengamanan yang harus diterapkan pada sistem ini adalah super ketat.

Pentingnya informasi pada website komersial mungkin berada pada titik diantara kumpulan hobi dari user dan militer.

### Ancaman Keamanan (Security Threats)

Apakah ancaman yang mungkin mengancam website anda?. Berikut ini beberapa kemungkinan ancaman yang mengancam website anda:

- SQL Injection
- Cross Site Scripting
- Denial of Service (DoS)
- Error pada Software

Mari kita kupas satu per satu ancaman tersebut.

## SQL Injection

SQL Injection merupakan teknik untuk menembus sistem authentication dengan memanipulasi query SQL. Manipulasi ini biasanya dilakukan pada input pada form login dengan memasukkan beberapa karakter tertentu seperti ' (petik satu), #, dan sebagainya. Lihat query berikut



Gambar 4.0: Metode SQL Injection

Jika query yang kita gunakan untuk memproses login diatas adalah seperti berikut

```
$hasil = mysql_query("SELECT * FROM admin WHERE username='admin' AND password='pas123'");
```

Namun pada input username cracker memasukkan admin'#. Jadi query diatas akan menjadi seperti berikut

```
$hasil = mysql_query("SELECT * FROM admin WHERE username='admin'# AND password='pas123'");
```

Tanda # pada query MySQL sama dengan -- yang berarti komentar. Jadi setelah tanda # SQL menganggap text tersebut komentar sehingga tidak dieksekusi. Hasil dari query ke dua selalu mengembalikan satu baris artinya *true* karena sama saja dengan query berikut,

```
SELECT * FROM admin WHERE username='admin';
```

Sebenarnya masih banyak cara yang digunakan untuk memanipulasi query SQL. Tidak hanya terbatas dengan satu cara seperti diatas.

Cara mengatasinya adalah dengan mem-filter input dari user dengan addslashes() atau strip\_tags(). Hal ini sudah pernah kita lakukan pada proyek bab 3. Dimana kita membuat fungsi filter\_str() yang hanya mengijinkan alfabet, numerik, dan underscore. Source dari fungsi filter\_str() adalah sebagai berikut

```
function filter_str($string)
{
 $filter = ereg_replace('[^a-zA-Z0-9_]', '', $string);
```

```

 return $filter;
}

```

Namun kebanyakan server sekarang sudah diset untuk tidak mengijinkan karakter-karakter yang berbahaya. Tapi untuk berjaga-jaga sebaiknya kita membuat fungsi sendiri seperti diatas. Karena tidak semua server diset demikian.

## Cross Site Scripting (XSS)

Cross Site Scripting merupakan metode eksplorasi sistem dengan cara memanipulasi variabel pada URL. Biasanya pada URL terdapat variabel atau menu untuk masuk ke halaman lainnya. Contoh pada beberapa halaman yang pernah kita buat index.php?proses=proses\_form. Isi dari proses ini yang menjadi target XSS bagi para attacker.

Cara yang biasanya digunakan adalah dengan mengganti isi dari proses dengan kode tertentu. Contoh

<http://www.website.com/index.php?ref=user>

Jika kita mengganti isi dari ref menjadi seperti berikut apa yang akan terjadi?

<http://www.website.com/index.php?ref=<h1>Your%20SITE%20Is%20Vulnerable</h1>>  
Setiap kita membuka halaman web, website tersebut melakukan request ke server. Jika halaman yang dituju tidak ada maka browser akan menampilkan permintaan kita tersebut. Isi dari XSS kita diatas tidak berbahaya. Bagaimana kalau isinya adalah javascript yang bertugas mendapatkan cookie. Ini adalah masalah serius.

Untuk mencegahnya kita dapat mem-filter isi dari data dari proses `$_GET` pada URL. Kita dapat menggunakan fungsi `strip_tags()` dan fungsi yang kita buat sendiri seperti `filter_str()`.

## Denial of Service (DoS)

Salah satu serangan yang paling sulit untuk diatasi adalah *Denial of Service (DoS)*. DoS merupakan serangan yang membanjiri target dengan request-request yang sangat menguras resource CPU. Sehingga sistem target terkadang lumpuh total.

Pada awal tahun 2000, terjadi serangan *Distributed Denial of Service (DDoS)* terhadap situs-situs terkemuka seperti Yahoo!, eBay, Amazon, E-Trade dan Buy.com. Situs-situs diatas shutdown selama beberapa jam karena serangan DDoS ini. Serangan semacam ini memerlukan ratusan bahkan ribuan komputer untuk melakukan request secara bersamaan. Cracker umumnya mengontrol komputer-komputer yang ingin digunakan untuk menyerang sistem target melalui IRC.

Salah satu alasan mengapa serangan ini sangat sulit untuk dihindari adalah karena sistem target dibanjiri dengan request dengan jumlah sangat besar. Metode yang digunakan dapat berupa *menginstal program* pada mesin target yang menghabiskan banyak resource CPU. *Reserve Spam* dan menjalankan *Automated tools*.

*Reserve Spam* menyebabkan seseorang untuk mengirim email sampah dengan target yang ada pada daftar email yang ada pada address book. Sehingga target akan dibanjiri reply dari email yang dikirimkan.

*Automated tools* digunakan untuk melancarkan serangan DDoS pada target. Tanpa pengetahuan yang lebih seseorang dapat menscan banyak mesin untuk mengetahui mesin yang vulnerable(terdapat celah keamanan). Kemudian menginstal tool tersebut ke mesin target. Serangan yang berbasis DDoS umumnya menggunakan nomor port yang tinggi.

## Error pada Software

Ini masih tetap memungkinkan software yang berjalan disistem mengalami error. Error pada software dapat menyebabkan hal yang serius. Kita tahu semua bisnis yang memakai sistem komputer untuk memproses data mungkin saja memakai software yang vulnerable(memiliki celah keamanan).

Error pada software dapat menyebabkan sistem berlubang, resiko finansial, dan layanan yang buruk pada kostumer. Kebanyakan dari error ini karena spesifikasi yang buruk dan buruknya tes yang dilakukan.

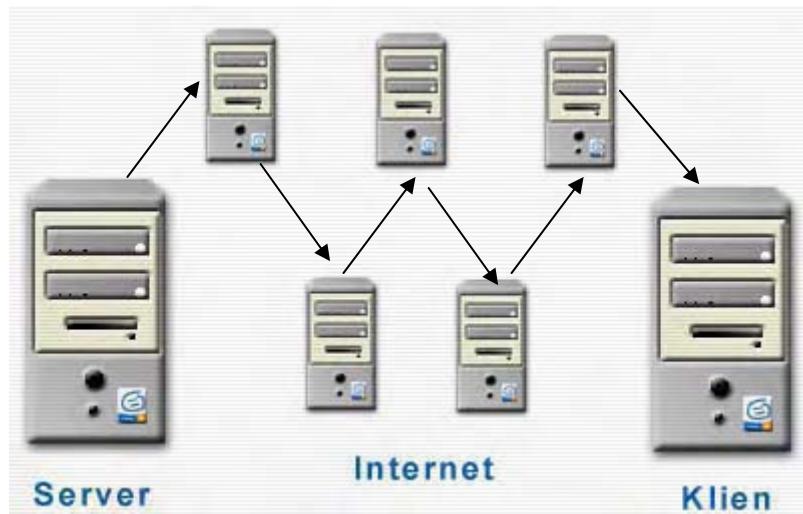
Sebelum melakukan pemasaran ada baiknya kita mengetes website kita pada segala kondisi. Seperti mengetes semua kemungkinan kondisi input, mengetes pada spesifikasi hardware yang berbeda, mengetes pada semua browser, mengetes pada semua sistem operasi dan sebagainya. Ini cukup penting dilakukan karena konsumen kita pasti memiliki spesifikasi yang berbeda-beda.

## Enkripsi Data

Enkripsi merupakan suatu proses algoritma yang merubah informasi awal menjadi data yang berupa string secara acak(random). Informasi yang lalu lalang di internet bagaikan sebuah papan pengumuman yang siap dibaca oleh siapa saja yang melihatnya. Tanpa di-enkripsi informasi dari server ke klien atau sebaliknya dapat dimonitor oleh seseorang dari jarak jauh.

Jika informasi yang dikirimkan hanya berupa teks lagu itu tidak masalah. Namun bagaimana jika yang dikirimkan adalah password kita untuk login ke suatu situs. Itu baru masalah, dan memang itu adalah masalah serius.

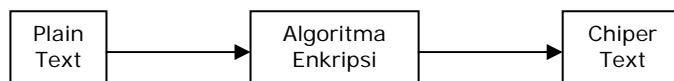
Ketika anda berinternet anda mungkin tidak sadar bahwa aktivitas anda sedang dimonitor oleh seseorang. Mereka mengetahui semua paket yang dikirimkan atau yang diterima komputer anda. Jika password anda tidak terenkripsi maka dengan mudah orang tersebut melihat password anda. Lihat gambar 4.1.



Gambar 4.1: Pengiriman data pada internet

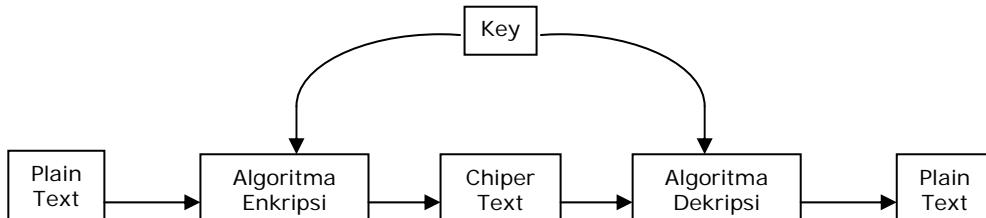
Seperti terlihat pada gambar diatas bahwa informasi yang kita terima sebenarnya melewati beberapa host terlebih dahulu sebelum akhirnya kita terima. Ini dikarenakan koneksi TCP/IP memecah data menjadi paket-paket kecil dan mengirimkan paket ini dari satu mesin ke mesin lainnya sebelum akhirnya sampai di sisi klien. Jadi sebelum data akhirnya sampai pada komputer anda mungkin saja data itu "mampir" ke host yang tidak terpercaya. Dari penggambaran diatas dapat kita simpulkan pentingnya enkripsi dalam suatu transmit di internet.

Pada enkripsi dikenal isitilah *Chiper Text* yang merupakan string acak hasil dari enkripsi sebuah text biasa(plain text). Lihat gambar 4.2 untuk memperjelas proses enkripsi



Gambar 4.2: proses enkripsi

Kebalikan dari enkripsi adalah dekripsi. Proses dekripsi adalah mengubah kembali chiper text yaitu string hasil enkripsi menjadi plain text. Untuk melakukan dekripsi kita memerlukan *key* atau kunci. Key yang digunakan untuk dekripsi harus dengan key yang digunakan sewaktu kita mengenkripsi text. Lihat gambar 4.3.



Gambar 4.3: Enkripsi mengubah plain text menjadi chiper text. Dekripsi mengubah chiper text menjadi plain text

Metode untuk melakukan enkripsi cukup beragam seperti MD5 dan SHA. Kedua metode ini didukung oleh PHP. Untuk algoritma MD5 kita dapat menggunakan fungsi `md5()`. Sedang untuk algoritma SHA kita menggunakan fungsi `sha1()`. Untuk mencobanya jalankan PHP DESIGNER 2005 klik menu **File-New-PHP** kemudian ketik kode berikut

```

<?php
// enkripsi menggunakan MD5 dan SHA

$string = 'rahasia-webmaster.com';
$chiper_md5 = md5($string);
echo "String awal: $string

 . Setelah dienkrip MD5: $chiper_md5

 . Panjang string setelah dienkrip: ".strlen($chiper_md5);
echo "<p></p>";
$chiper_sha = sha1($string);
echo "String awal: $string

 . Setelah dienkrip SHA: $chiper_sha

 . Panjang string setelah dienkrip: ".strlen($chiper_sha);
?>

```

Simpan pada direktori **hari-15** dengan nama **md5\_sha1.php**. Output yang ditampilkan browser dari kode diatas adalah

```

String awal: rahasia-webmaster.com
Setelah dienkrip MD5: 213ecd6efcb09078823634e18a25acf0
Panjang string setelah dienkrip: 32
String awal: rahasia-webmaster.com
Setelah dienkrip SHA: 1f403aaa8492129605ca27a4421b96895bb85681
Panjang string setelah dienkrip: 40

```

Perlu diingat bahwa baik proses MD5 dan SHA tidak dapat didekripsi. Artinya hasil enkripsi ini tidak bisa dikembalikan menjadi string asalnya. Metode ini cocok kita gunakan untuk sistem login. Namun hasil enkripsi dari kedua fungsi ini yaitu `md5()` dan `sha1()` akan selalu sama dengan string input yang digunakan.

Fungsi lain yang dapat kita gunakan untuk melakukan enkripsi adalah `crypt()`. Fungsi ini memiliki dua parameter satu bersifat wajib dan satu bersifat opsional. Struktur dasar dari `crypt()` adalah sebagai berikut

```
crypt(string yang dienkrip, [kunci]);
```

Meskipun memiliki kunci tetapi fungsi crypt tidak dapat mendekripsi hasil dari enkripsinya sendiri. Satu yang menarik dari fungsi ini adalah jika parameter kunci tidak diisi maka hasil enkripsi akan selalu berubah setiap kali fungsi tersebut dipanggil. Untuk mencobanya buat file baru pada PHP DESIGNER 2005 dengan mengklik **File-New-PHP**, lalu ketik kode berikut

```
<?php
// enkripsi menggunakan fungsi crypt
$string = 'rahasia-webmaster.com';
echo '<p>Hasil yang dienkrip dengan crypt() tanpa kunci:

.' . crypt($string) . ' <== selalu berubah.</p>
.<p>Hasil yang dienkrip dengan crypt() menggunakan kunci:

.' . crypt($string, 'xx') . ' <== tidak berubah.</p>';
?>
```

Simpan pada direktori **hari-15** dengan nama ***crypt.php***. Jalankan pada browser tekan tombol **Refresh** pada browser beberapa kali untuk melihat perubahannya. Hasilnya akan seperti berikut:

Hasil yang dienkrip dengan crypt() tanpa kunci:

**\$1\$jV5.wZ2.\$SyR6s4.NHCD1VgkZ3Fb28/** <== selalu berubah.

Hasil yang dienkrip dengan crypt() menggunakan kunci:

**xx5S2eXCOIs4w** <== tidak berubah.

Pada fungsi `crypt()` yang pertama hasil yang akan ditampilkan selalu berubah-ubah. Sedangkan pada fungsi `crypt()` yang kedua hasilnya akan selalu sama karena kita memberikan parameter kunci yaitu 'xx'. Panjang string untuk kunci maksimal adalah 2. Jika anda memasukkan string yang panjangnya lebih dari dua maka yang diambil adalah dua string pertama.

## Secure Login

Agar data baik yang diterima maupun dikirim klien tetap aman. Maka sudah seharusnya kita memperhatikan hal ini. Sistem login yang pernah kita buat pada tiga bab sebelumnya sangatlah risikan. Mengapa? Karena sistem login yang kita terapkan tanpa menggunakan enkripsi sama sekali. Sehingga informasi password dapat dengan mudah diketahui orang lain.

Kali ini kita akan membuat sistem login yang lebih aman, meskipun bukan jaminan 100%. Tetapi ini setidaknya akan mempersulit orang yang ingin mencuri password. Enkripsi yang akan kita gunakan pada sistem login kita berdasarkan algoritma enkripsi MD5. Metode enkripsi MD5 sekarang pun sudah ada tool untuk mengcracknya. Meskipun kemungkinan berhasilnya kecil, tapi kita tetap harus berhati-hati.

Tool semacam ini biasanya dinamakan **MD5 brute force**. Seperti namanya *brute force* artinya tool ini akan mencoba segala kemungkinan password yang menghasilkan string hasil enkripsi MD5. Jadi untuk mempersulit lagi kita harus memodifikasi enkripsi MD5 dengan cara-cara tertentu. Misalnya membalik urutan hasil enkripsi MD5, mengambil beberapa karakter tertentu, mengacak letak karakter dan sebagainya.

### PENERAPAN TEORI SECURE LOGIN

Jalankan PHP DESIGNER 2005 klik menu **File-New-SQL** untuk membuat file SQL baru. Kemudian ketik kode berikut

```
-- buat database secure_login
```

```
-- jika sudah ada hapus

DROP DATABASE IF EXISTS secure_login;
CREATE DATABASE secure_login;

-- aktifkan database secure_login
USE secure_login;

-- buat tabel user
-- panjang password 32 karena panjang enkripsi MD5 demikian
CREATE TABLE user
(
 username varchar(16) NOT NULL,
 password varchar(32) NOT NULL,
 PRIMARY KEY (username)
) TYPE='MyISAM';

-- buat privilege untuk user dengan nama secure
-- dan password secure123 untuk database ini
GRANT SELECT, UPDATE, INSERT, DELETE ON secure_login.*
TO secure@localhost IDENTIFIED BY 'secure123';
```

Simpan pada direktori **hari-15** dengan nama **secure.sql**. Eksekusi file ini dari phpMyAdmin atau dari MySQL Monitor.

Selanjutnya kita akan membuat file PHP baru klik **File-New-PHP**, lalu ketik kode berikut

```
<?php

/***** Kumpulan Fungsi untuk file secure_login.php ****

***** */

// cegah pengaksesan langsung dari browser
// isi dari $_SERVER['PHP_SELF'] adalah
// /webmaster/Bab4/hari-15/fungsi.php

if (eregi('fungsi.php', $_SERVER['PHP_SELF']))
 exit('Error: Akses ditolak');

// fungsi untuk melakukan koneksi ke MySQL server
function koneksi_db()
{
 // untuk username = secure dan password = secure123
 @$koneksi = mysql_connect('localhost', 'secure', 'secure123');
 // jika gagal melakukan koneksi kembalikan false
 if (!$koneksi)
 return false;
 else
 {
 // pilih database yang digunakan
 mysql_select_db('secure_login');
 return true;
 }
}

// fungsi untuk menfilter karakter
function filter_str($string)
{
 $filter = ereg_replace('[^a-zA-Z0-9_]', '', $string);
 return $filter;
}
```

```
// fungsi untuk mengenkripsi string dengan MD5 dan membaliknya
function balik_md5($string)
{
 // balik urutan string yang terenkripsi dengan md5()
 // menggunakan fungsi strrev()
 $chiper_text = strrev(md5($string));
 return $chiper_text;
}

?>
```

Simpan dengan nama ***fungsi.php***. Semua fungsi diatas saya rasa sudah jelas karena banyak komentar penjelasnya. Untuk alasan keamanan kita mencegah pengaksesan file-file konfigurasi secara langsung dari browser.

Pada penggerjaan proyek berskala besar. Tulislah kode anda untuk semudah mungkin dapat dibaca. Jangan terlalu pelit dalam memberi jarak ataupun memberi komentar. Ini akan mempermudah anda dalam menangani error.

Buat file PHP baru dengan mengklik **File-New-PHP**, kemudian salin listing kode berikut

```
<?php

/*
** Nama File.....: secure_login.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan...: untuk mengetes metode MD5 pada sistem login
***** */

// panggil file fungsi.php
include ('fungsi.php');

echo '<p>Hanya untuk testing...</p>';

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form_login';

/* Jika koneksi gagal tampilkan pesan */
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali parameter untuk host, username, dan password!.');

// tangani setiap proses dengan switch dan case
switch ($proses)
{
 /*
 case 'form_login':
 */

 echo '<h2>Form login</h2>';
 . '<form action="secure_login.php?proses=proses_login" method="post">'
 . 'Username: <input type="text" name="username">
'
 . 'Password: <input type="password" name="password">
'
 . '<input type="submit" value="LOGIN">'
 . '</form>'
 . '<p>Tambah User</p>';

 break;
}
```

```

/*************
case 'proses_login':
/************

// daftarkan variabel
$username = filter_str($_POST['username']);
$password = filter_str($_POST['password']);

// enkripsi password
$password = balik_md5($password);

// cek kecocokan data dengan yang ada didatabase
$hasil = mysql_query("SELECT * FROM user WHERE username='$username' AND password='$password'");

// jika jumlah baris lebih dari nol maka cocok
if (mysql_num_rows($hasil) > 0)
{
 echo '<p>Login Berhasil</p>' .
 'Halaman Login';
}
else
{
 echo '<p>Username atau Password salah!</p>' .
 'Login Kembali';
}

break;
/************

/*************
case 'tambah_user':
/************

echo '<h2>Tambah User</h2>' .
 '<form action="secure_login.php?proses=proses_tambah" method="post">' .
 'Username: <input type="text" name="username">
' .
 'Password: <input type="text" name="password">
' .
 '<input type="submit" value="TAMBAH USER">' .
 '</form>' .
 '<p>Halaman Login</p>';

break;
/************

/*************
case 'proses_tambah':
/************

// daftarkan variabel
$username = $_POST['username'];
$password = $_POST['password'];

// cek apakah data yang dipost mengandung karakter non alpanumerik
foreach ($_POST as $data)
{
 if (ereg('[^a-zA-Z0-9_]', $data))
 exit('Error: username atau password mengandung karakter bukan alpanumerik dan underscore.');
}

// jika tidak ada kesalahan maka enkripsi password
$password = balik_md5($password);

```

```

// masukkan data ke database
$hasil = mysql_query("INSERT INTO user VALUES ('$username', '$password')");
// jika ada kesalahan tampilkan
if (!$hasil)
 echo 'Error: Gagal memasukkan data ke database';
else
 echo 'Data berhasil dimasukkan ke database

' .
 'Tes Login';

break;

} // akhir switch

// tutup koneksi MySQL
mysql_close();

?>

```

Simpan pada direktori **hari-15** dengan nama **secure\_login.php**. Jalankan pada browser anda. Klik link *Tambah User* untuk menambah user baru. Pada username isikan **guest** dan pada password isikan **pas123**. Coba lihat isi dari tabel user pada phpMyAdmin atau MySQL monitor. Jika sudah masuk maka outputnya akan seperti berikut

username	password
guest	0da656c1352c7ec167b91122dd6d1aff

Anda lihat password dari field bukan *pas123*. Untuk login user tidak perlu mengetikkan password *0da656c1352c7ec167b91122dd6d1aff*. Tetapi tetap menggunakan *pas123* yang kemudian dienkrip menggunakan *balik\_md5()* dan menghasilkan string chiper yang sama dengan didatabase.

## Firewall

Firewall yang berarti dinding api, merupakan alat yang digunakan untuk melindungi komputer anda dari serangan pihak luar. Firewall menyaring paket dan menolak traffic yang tidak sesuai dengan ketentuan user. Firewall juga membatasi aktivitas seseorang dan mesin lain di luar firewall.

Firewall dapat membatasi jaringan protokol yang digunakan, membatasi host yang akan dikoneksi, menyembunyikan identitas komputer anda ketika online dan masih banyak lagi.

Firewall dapat berupa *hardware* atau *software*. Hardware firewall biasanya seperti *router* dengan fungsi yang dapat menfilter traffic yang masuk. Sedangkan firewall yang berbentuk software merupakan sebuah program yang berjalan pada komputer anda. Software ini memonitor semua traffic dari satu jaringan ke jaringan yang lain. Jika traffic tersebut tidak sesuai dengan ketentuan maka traffic tersebut akan dihentikan atau ditolak.

Untuk firewall yang berupa hardware harganya cukup mahal namun tingkat keamanannya lebih baik. Sedangkan untuk software banyak yang tersedia secara gratis namun cukup baik seperti <http://www.zonealarm.com/>.

# Error Handling

Pada materi kali ini kita akan membahas bagaimana menangani suatu error (Error Handling). Materi yang harus anda kuasai antara lain:

- Programming Error
  - Syntax Error
  - Runtime Error
  - Logic Error

Itulah beberapa materi yang harus anda kuasai.

## Programming Error

Bahasa pemrograman apapun yang anda gunakan, setidaknya bahasa itu pasti memiliki tiga tipe error pada pemrograman yaitu Syntax, Runtime dan Logic Error. Untuk yang pertama kita akan membahas syntax error.

### Syntax Error

Seperti namanya kesalahan ini disebabkan karena kita salah menuliskan syntax pada kode kita. Seperti yang kita tahu *Syntax* merupakan aturan baku yang harus diikuti agar kode yang kita tulis valid. Syntax error juga sering disebut *Parse Error*.

Contoh peraturan-peraturan yang harus kita ikuti pada PHP adalah akhir dari suatu statement perintah harus diberi tanda titik koma(;). String harus berada diantara tanda petik (petik satu atau petik dua). Jika ingin membuat fungsi maka kita harus memberi keyword *function* dan kodennya diapit kurung kurawal.

Contoh-contoh tersebut merupakan peraturan yang harus diikuti. Jika kita melanggarinya maka script kita tidak akan berkerja. Dan PHP menampilkan pesan error pada saat kita mencoba mengeksekusi script tersebut.

Saya harap anda tidak kesal pada PHP karena mengeluarkan pesan error tersebut. Karena pesan error tersebut sangat berguna bagi kita untuk mengetahui jenis error, pada file mana, dan baris keberapa. Contoh jika muncul error seperti ini

**Parse error: syntax error, unexpected T\_ECHO in C:\Apache2\htdocs\webmaster\Bab4\hari-15\error.php on line 3**

Pesan error diatas disebabkan oleh kode berikut

```
<?php
 $string = 'Apakah Saya Error?
 echo $string;
?>
```

Jika anda mengira error yang terjadi pada baris ke-3 maka anda salah. Pada pesan error tersebut ada kata **unexpected T\_ECHO** artinya perintah echo ini oleh PHP dianggap tidak terduga. **Error sebenarnya** yang menyebabkan munculnya pesan tersebut adalah pada baris **ke-2**. Dimana kita lupa tidak memberi tanda titik koma (;) pada akhir dari suatu statement perintah.

Error yang terjadi tidak selalu pada baris yang ditunjukkan oleh PHP. Error yang terjadi biasanya berada pada baris sebelumnya dan pasti bukan baris setelah nomor error yang disebutkan. Contoh lain jika pesan error seperti berikut

**Parse error:** syntax error, unexpected \$end in C:\Apache2\htdocs\webmaster\Bab4\hari-15\error.php on line 7

Kode yang menyebabkan error tersebut adalah

```
<?php
 $a = 1;
 while ($a < 6)
 {
 echo $a;
 $a++;
 }
?>
```

Error diatas terjadi karena kita tidak memberi tanda kurung penutup } pada looping while yang kita tulis. Sehingga tanda penutup script PHP ?> dianggap tidak terduga (unexpected).

Secara umum syntax error adalah yang paling mudah ditangani. Ini dikarenakan jika kita membuat syntax error PHP akan menampilkan pesan yang memberitahu kita dimana letak kesalahannya.

## Runtime Error

Rutime error lebih sulit untuk dideteksi dari pada syntax error. Jika syntax error begitu terdeteksi ketika script dieksekusi pertama kali. Lain halnya dengan runtime error. Munculnya runtime error biasanya disebabkan hubungan script anda dengan suatu aksi atau kondisi tertentu. Lihat script berikut

```
<?php
 include('tidak_ada.php');
?>
```

Script diatas sudah valid dan tidak memiliki syntax error. Namun ketika dijalankan muncul pesan error seperti berikut

**Warning:** main(tidak\_ada.php) [[function.main](#)]: failed to open stream: No such file or directory in I:\Apache2\htdocs\webmaster\Bab4\hari-15\error.php on line 2

**Warning:** main() [[function.include](#)]: Failed opening 'tidak\_ada.php' for inclusion (include\_path='.;i:\php5.04\PEAR') in I:\Apache2\htdocs\webmaster\Bab4\hari-15\error.php on line 2

Bagaimanapun tidak ada yang salah dengan kode kita, namun kita coba memanggil file yang tidak ada yaitu *tidak\_ada.php*. Sehingga PHP mengeluarkan error tersebut.

```
<?php
 $a = 5;
 $b = 0;
 $hasil = $a/$b;
?>
```

Script diatas sudah sempurna dan tidak memiliki syntax error sama sekali. Namun pada saat dijalankan muncul pesan error seperti berikut

**Warning:** Division by zero in I:\Apache2\htdocs\webmaster\Bab4\hari-15\error.php on line 4

Error diatas terjadi karena kita bagi(divide) bilangan dengan nol (zero). Secara matematika hasilnya tidak terdefinisi. Error tersebut dapat dihindari dengan cara menentukan kondisi sebelum melakukan penghitungan. Misalnya dengan statemen if.

Penyebab umum runtime error antara lain:

- Memanggil fungsi yang tidak ada
- Membaca atau menulis file
- Interaksi dengan MySQL dan database

Mari kita bahas satu per satu.

## Memanggil Fungsi Yang Tidak Ada

Anda mungkin sering salah menulis fungsi karena salah pengejaan atau kurang karakter tertentu. Misalnya fungsi `ereg_replace()` mempunyai tanda `_` sebagai pemisah antar kata sedangkan fungsi `addslashes()` tidak. Namun kesalahan tipe ini sangat mudah dicari. Lihat contoh berikut

**Fatal error: Call to undefined function fungsiku() in I:\Apache2\htdocs\webmaster\Bab4\hari-15\error.php on line 2**

Error diatas disebabkan oleh script berikut

```
<?php
 fungsiku();
?>
```

Fungsi `fungsiku()` tidak ada karena kita tidak membuatnya. Jadi secara otomatis PHP akan mengeluarkan pesan *undefined function* artinya fungsi ini tidak terdefinisi. Selain karena fungsi tersebut tidak ada. Ada satu kesalahan lagi yang berhubungan dengan fungsi. Yaitu kesalahan pada parameter fungsi yang kita tulis.

**Warning: Wrong parameter count for ereg() in I:\Apache2\htdocs\webmaster\Bab4\hari-15\error.php on line 2**

Error diatas disebabkan oleh script berikut

```
<?php
 ereg('string');
?>
```

Fungsi `ereg()` parameter minimal yang harus diisi ada dua yaitu pola string yang dicari dan string yang dicari. Namun pada kode diatas kita hanya mengisikan satu parameter saja. Sehingga muncul pesan error seperti diatas.

Error tipe ini sangat mudah ditangani karena PHP akan menampilkan pesan *undefined function* di file mana dan baris keberapa.

## Membaca atau Menulis File

Kesalahan tipe ini biasanya disebabkan karena kita tidak memiliki hak(permission) pada file tersebut. Misalnya permission untuk baca, eksekusi atau menulis. Jika anda menggunakan OS Windows maka kesalahan tipe ini mungkin tidak akan anda alami. Namun jika anda menggunakan OS yang berbasis UNIX seperti Linux, BSD dan lainnya. Masalah permission mungkin sering anda alami jika anda tidak bekerja pada direktori home anda.

**Warning: file(fileku.txt) [[function.file](#)]: failed to open stream: Permission denied in /home/proyek/error.php on line 113**

Error diatas terjadi karena kode berikut

```
file(fileku.txt);
```

Kode diatas sebenarnya sudah benar namun karena permission dari file fileku.txt diset untuk tidak dapat dibaca maka yang terjadi adalah error seperti diatas. Untuk itu pastikan jika anda membuat script PHP yang berinteraksi dengan file. Pastikan permission direktori atau filenya dapat kita baca, tulis dan eksekusi. Anda dapat menggunakan perintah chmod untuk merubah permission direktori atau file. Namun anda harus login sebagai root.

```
chmod 777 fileku.txt
```

## Interaksi dengan MySQL atau Database Lainnya

Berintaksi dengan MySQL akan banyak menghasilkan error. Fungsi `mysql_connect()` saja setidaknya dapat memunculkan error berikut

- MySQL Connection Failed: Can't Connect to MySQL server on 'namahost' (111) -> gagal melakukan koneksi
- MySQL Connection Failed: Can't connect to MySQL server through socket '/tmp/mysql.sock' (111) -> Masalah soket pada sistem UNIX
- MySQL Connection Failed: Unknow MySQL server host 'namahost' -> Nama host tidak diketahui atau tidak terdaftar pada tabel mysql
- MySQL Connection Failed: Access denied for user: 'namauser@localhost' (Using password: YES)

Seperti fungsi lainnya `mysql_connect()` juga akan mengembalikan nilai *false* jika terjadi error. Dengan demikian anda dapat dengan mudah menghandel kemungkinan terjadinya error.

Jika anda tidak menghandel error yang terjadi maka scrit anda akan terus dieksekusi dan terus berinteraksi dengan database. Dengan demikian hasil yang ditampilkan akan kacau dan terlihat tidak profesional. Seperti kode yang pernah kita tulis sebelumnya kita dapat menghandel error menggunakan statemen if.

```
@ $koneksi = mysql_connect("localhost", "user", "password");
if (!$koneksi)
 exit("Gagal melakukan koneksi ke MySQL server. Kontak Admin");
```

Dengan menggunakan kode seperti diatas maka pesan error yang dimunculkan pada user akan sesuai dengan keinginan kita. Tanda @ kita gunakan agar error internal yang diciptakan PHP karena kegagalan fungsi tidak ikut dicetak ke layar.

Namun jika anda kode anda pada tahap pengembangan (development). Sebaiknya jangan gunakan tanda @ karena pesan error secara spesifik tidak akan anda ketahui.

## Logic Error (Kesalahan Logika)

Tipe error inilah yang paling sulit untuk kita temukan dan perbaiki. Kode yang anda tulis sudah sepenuhnya benar dan tidak memiliki syntax error didalamnya. Namun hasilnya tidak seperti yang anda inginkan.

Contoh sederhana kode yang menampilkan logic error adalah seperti berikut

```
<?php
for ($i=0; $i<10; $i++); ←———— tanda ; menyebabkan logic error
{
 echo "Looping ke-$i
";
}
?>
```

Blok kode ini sudah sepenuhnya benar. Dan tidak memiliki syntax error. Anda sebagai programmer menginkan blok kode dieksekusi sepuluh kali. Sehingga hasilnya looping ke-0, looping ke-1 dan seterusnya. Akan tetapi yang ditampilkan pada layar adalah

Looping ke-10

Ini dikarenakan pada baris pertama yaitu anda memberi tanda titik koma (;) sehingga looping for tidak diulang sebanyak 10 kali. Melainkan hanya dieksekusi satu kali.

\*\*\*\*\*

## Kalender

27	28	29	30	31	1	2	3
4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27
28	29	30	31	1	2	3	4
5	6	7	8	9	10	11	12
14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29

Ikuti langkah-langkah berikut:

- Masuk pada direktori C:\Apache2\htdocs\webamster\Bab4\
- Buat folder baru, namai folder tersebut **hari-16-17**

## Contoh-contoh Aplikasi

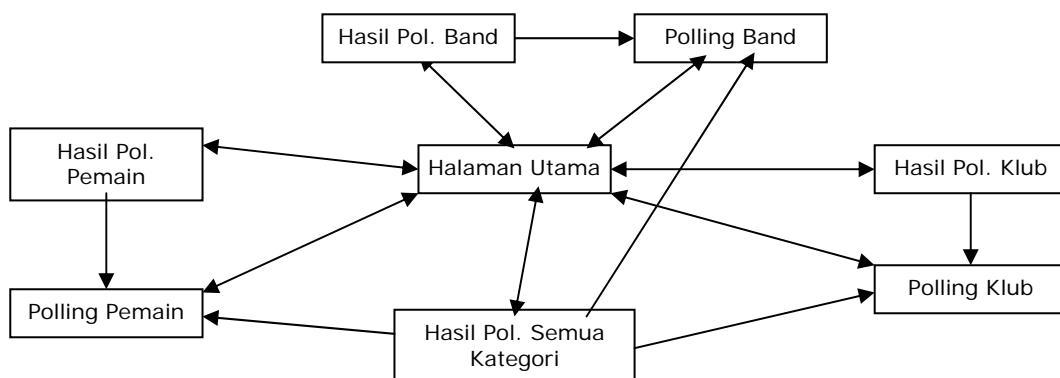
Pada materi kali ini kita akan mencoba membuat aplikasi-aplikasi yang biasanya terdapat pada suatu website. Aplikasi-aplikasi yang akan kita buat adalah

- NgawurPolling (website polling)
- Super Iklan (website iklan)
- Forum Bareng (website forum)

Ketiga website diatas sebenarnya lebih cocok untuk dijadikan satu kesatuan dalam satu website. Namun untuk mempermudah pemahaman, kita membuatnya secara terpisah. Setelah anda mengerti anda dapat menggabungkannya menjadi satu website. Kita mulai dari website yang pertama yaitu NgawurPolling.

## NgawurPolling.localhost

Dari namanya saja sudah terlihat bahwa polling ini hanya awur-awuran saja. Tapi itu untuk pollingnya, sedangkan untuk kodenya sama sekali tidak ngawur. Dan bahkan rasanya paling kompleks diantara proyek-proyek yang pernah kita kerjakan. Dalam dua hari ke depan kita akan mengerjakan aplikasi ini. Skema berikut menunjukkan hubungan antar halaman.



Gambar 4.4: Skema halaman NgawurPolling

## Sketsa Proyek

Sebelum melanjutkan ada beberapa masalah yang pemecahannya akan kita terapkan pada website NgawurPolling. Masalah-masalah yang ada pada proyek ini adalah

### Pada bagian pengunjung

- User dihadapkan pada tiga pilihan polling (polling pemain terfavorit, polling klub terfavorit dan polling band terfavorit).
- User hanya boleh mengikuti polling satu kali saja pada setiap kategori polling (jika admin memang menerapkannya).

- Untuk mengecek apakah user sudah melakukan polling atau belum dapat dicek dari cookie atau alamat IP.
- User dapat melihat hasil polling masing-masing kategori atau hasil polling semua kategori.
- Hasil polling ditunjukkan oleh warna grafik yang berbeda-beda pada masing-masing kategori.

#### Pada bagian administrator

- Untuk masuk ke halaman administrator, admin harus memasukkan username dan password.
- Admin dapat menghapus dan menambah daftar poling dari masing-masing kategori.
- Admin dapat melihat perolehan suara dari masing-masing kandidat.

IP(Internet Protocol) merupakan alamat sebuah komputer dalam suatu jaringan baik lokal maupun internet. Alamat IP satu komputer dengan komputer lain bersifat unik. Artinya tidak ada yang sama satu sama lain. Untuk alamat IP komputer yang berada dalam jaringan lokal misalnya 192.168.7.1. Sedangkan untuk komputer kita pada jaringan virtual yaitu localhost alamat IP-nya adalah 127.0.0.1.

Untuk mengaplikasikan proyek ini sebelumnya ikuti langkah-langkah berikut untuk konfigurasi direktori:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab4\hari-16-17\
- Buat folder baru namai folder tersebut **polling**
- Masuk pada direktori **polling** yang baru anda buat
- Buat beberapa folder baru berikut **admin, gambar, inc** dan **log**

Daftar file yang akan kita buat untuk membangun website NgawurPolling ditunjukkan pada tabel 4.1 berdasarkan letak direktorinya.

**Tabel 4.1** Daftar file untuk website NgawurPolling

Direktori <b>polling</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.php	Menampilkan halaman utama dan menampilkan form polling
poll_pemain.php	Menampilkan hasil polling pemain dan memproses polling pemain
poll_klub.php	Menampilkan hasil polling klub dan memproses polling klub
poll_band.php	Menampilkan hasil polling band dan memproses polling band
poll_semua.php	Menampilkan semua hasil polling
polling.sql	File SQL untuk membangun database polling
Direktori <b>polling\admin</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.php	Menampilkan form login atau menu utama untuk admin
pemain.php	Melihat, menambah, dan menghapus daftar polling pemain
klub.php	Melihat, menambah, dan menghapus daftar polling klub
band.php	Melihat, menambah, dan menghapus daftar polling band
Direktori <b>polling\inc</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.php	Me-redirect lokasi halaman
header.php	Menampilkan header HTML untuk semua halaman
fungsi.php	Berisi kumpulan fungsi untuk keperluan NgawurPolling
class_polling.php	Class untuk menggambar grafik dari masing-masing kandidat
konfig.php	Berisi konfigurasi untuk NgawurPolling
Direktori <b>polling\gambar</b>	
<b>Nama File</b>	<b>Fungsi</b>
red_bar.jpg	Grafik bar warna merah

green_bar.jpg	Grafik bar warna hijau
blue_bar.jpg	Grafik bar warna biru
index.php	Meredirect ke halaman utama
Direktori <b>polling\log</b>	
<b>Nama File</b>	<b>Fungsi</b>
ip.dat	Untuk menyimpan alamat IP dan jenis polling user
index.php	Meredirect ke halaman utama

Untuk file pertama kita akan membuat file gambar untuk grafik bar. Aplikasi yang kita gunakan adalah aplikasi bawaan Windows untuk mengolah gambar yaitu **Paint**. Jalankan Paint dengan cara klik **Start > Program > Accessories > Paint** atau klik **Start > Run** lalu ketikkan mspaint [ENTER] atau klik **OK**.

Setelah Paint terbuka akan nampak ruang kerja berupa kotak putih(lembar gambar), toolbox, dan color box. Klik menu **Image > Attributes...** setelah itu akan muncul window **Attributes**. Pada isian **Width** isikan **5** dan pada **Height** isikan **15**. Namun pastikan satuan pada **Units** adalah **pixel**. Setelah itu klik **OK**.

Lembar gambar akan berubah menjadi sangat kecil. Lalu klik tombol  **(Fill With Color)**.

Setelah tombol  aktif klik warna merah pada color box yang terletak dibagian bawah. Setelah itu arahkan cursor ke lembar gambar.Klik lembar gambar tersebut, kemudian warna lembar gambar akan berubah menjadi merah.

Setelah itu klik menu **File > Save As...** untuk menyimpan gambar. Pada window **Save As** pastikan isi dari *Save as type:* adalah **JPEG(\*.JPG; \*.JPEG; \*.JPE; \*.JFIF)**. Arahkan lokasinya ke **C:\Apache2\htdocs\webmaster\Bab4\polling\gambar\** Simpan dengan nama **red\_bar.jpg**.

Untuk file **green\_bar.jpg** langkahnya sama hanya saja ganti warnanya dengan hijau sedangkan untuk **blue\_bar.jpg** warnanya adalah biru.

---

Selanjutnya kita akan membuat file SQL untuk membangun database polling yang kita gunakan untuk menyimpan statistik polling. Jalankan PHP DESIGNER 2005 klik menu **File > New > SQL** lalu ketik kode SQL berikut

#### Script dari file **polling.sql**

---

```
-- buat database polling
-- jika sudah ada hapus
DROP DATABASE IF EXISTS polling;
CREATE DATABASE polling;

-- aktifkan database polling
USE polling;

-- buat tabel poll_pemain
CREATE TABLE poll_pemain
(
 pemain varchar(20) NOT NULL PRIMARY KEY,
 suara int(5) NOT NULL
) TYPE='MyISAM'
COMMENT='Tabel Polling untuk Pemain Sepak Bola Terfavorit';

-- buat tabel poll_band
CREATE TABLE poll_band
(
 band varchar(25) NOT NULL PRIMARY KEY,
 suara int(5) NOT NULL
) TYPE='MyISAM'
```

```

COMMENT='Tabel polling untuk Band Terfavorit';

-- buat tabel poll_klub
CREATE TABLE poll_klub
(
 klub varchar(25) NOT NULL PRIMARY KEY,
 suara int(5)
) TYPE='MyISAM'
COMMENT='Tabel Polling untuk Klub Terfavorit';

-- buat tabel untuk login admin
CREATE TABLE admin
(
 username varchar(16) NOT NULL PRIMARY KEY,
 password varchar(32) NOT NULL
) TYPE='MyISAM';

-- buat privilege untuk database polling
GRANT SELECT, UPDATE, INSERT, DELETE ON polling.*
TO userpoll IDENTIFIED BY 'poll123';

```

Simpan pada direktori **polling** dengan nama **polling.sql**. Eksekusi file berikut melalui MySQL atau phpMyAdmin.

Kita mulai proyek kita dengan membuat file yang berada pada direktori **inc** terlebih dahulu. Karena file ini yang akan kita panggil pada hampir setiap halaman pada proyek kita. Untuk membuatnya klik **File > New > PHP** pada PHP DESIGNER 2005 lalu ketik kode berikut

#### Script dari file **index.php**

---

```

<?php

// jika user mencoba masuk ke direktori
// pindahkan ia ke halaman utama

header ('Location: ../index.php');

?>

```

---

**Save As** tiga kali masing-masing pada direktori **polling\inc**, **polling\log**, dan **polling\gambar** dengan nama **index.php**. Tugas file ini hanyalah melakukan redirect ketika ada yang mencoba masuk ke direktori inc.

File berikutnya juga sangat sederhana karena hanya berisi kumpulan kode HTML. Fungsi file ini adalah untuk mencetak header HTML pada setiap halaman. Buat file PHP baru pada PHP DESIGNER 2005 lalu ketik kode berikut

#### Script dari file **header.php**

---

```

<?php

/*
** Nama File....: header.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan...: untuk menampilkan header halaman
*****/
?>

<html>
<head>
<style>

```

---

---

```

body{font-family: verdana; font-size: 11px}
td{font-family: verdana; font-size: 11px}
a{color: #076fd0}
a:visited{color: #076fd0}
</style>
</head>
<body>
```

---

Simpan pada direktori **polling\inc** dengan nama ***header.php***.

File berikutnya adalah file yang berisi konfigurasi untuk NgawurPolling. Konfigurasi ini meliputi pengecekan user, grafik bar yang digunakan dan sebagainya. Klik **File > New > PHP** untuk membuat file PHP baru lalu ketik kode berikut

Script dari file ***konfig.php***

---

```

<?php

/*
** Nama File.....: konfig.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan....: berisi konfigurasi untuk polling
***** */

// cegah pengaksesan langsung dari browser
if (eregi('konfig.php', $_SERVER['PHP_SELF']))
 exit('Error: Akses ditolak');

// tentukan gambar yang digunakan untuk masing2 polling
$gambar['pemain'] = 'gambar/green_bar.jpg'; // grafik untuk polling pemain
$gambar['klub'] = 'gambar/red_bar.jpg'; // grafik untuk polling klub
$gambar['band'] = 'gambar/blue_bar.jpg'; // grafik untuk polling band

// tentukan lebar maximal dari gambar grafik
$lebar_max = 400; // dalam pixel

// batasi user untuk memilih hanya satu kali dengan cookie
$cek_cookie = 'no'; // yes atau no

// tentukan masa aktif cookie dalam detik
$masa_aktif = time() + 60 * 60 * 24 * 7; // 1 minggu

// batasi user untuk memilih hanya satu kali dengan alamat IP
$cek_ip = 'no'; // yes atau no

?>
```

---

Simpan pada direktori **polling\inc** dengan nama ***konfig.php***. Untuk grafik bar pada masing-masing polling kita simpan pada array **\$gambar**.

File berikutnya yang akan kita buat adalah file yang menyimpan semua fungsi untuk yang digunakan pada setiap halaman NgawurPolling. Untuk membuatnya seperti biasa **File > New > PHP** lalu ketik kode berikut

Script dari file ***fungsi.php***

---

```

<?php

/*
** Nama File....: fungsi.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
***** */
```

---

```

** Penjelasan...: Kumpulan fungsi untuk polling ***
***** **** **** **** **** **** **** **** **** **** **** /

// cegah pengaksesan langsung dari browser
if (eregi('fungsi.php', $_SERVER['PHP_SELF']))
 exit('Error: Akses ditolak.');

// fungsi untuk mengkoneksikan dengan MySQL server
function koneksi_db()
{
 // untuk username = userpoll dan password = poll123
 @$koneksi = mysql_connect('localhost', 'userpoll', 'poll123');
 // jika gagal melakukan koneksi kembalikan false
 if (!$koneksi)
 return false;
 else
 {
 // pilih database yang digunakan
 mysql_select_db('polling');
 return true;
 }
}

// fungsi untuk menyaring string selain alfabet, numerik dan _
function filter_str($string)
{
 $filter = ereg_replace('[^a-zA-Z0-9_]', '', $string);
 return $filter;
}

// fungsi untuk mengenkripsi string dengan metode MD5
// dan membalik urutannya
function balik_md5($string)
{
 // untuk membalik urutan string digunakan fungsi strrev()
 $chiper_text = strrev(md5($string));
 return $chiper_text;
}

// fungsi untuk mengecek session
function cek_session($nama_ses)
{
 // jika session kosong
 if (!isset($_SESSION[$nama_ses]))
 return false; // kembalikan nilai false
 else
 // jika tidak kosong
 return true; // kembalikan nilai true
}

// fungsi untuk logout (menghapus session)
function logout($nama_ses)
{
 // jika session kosong
 if (!isset($_SESSION[$nama_ses]))
 return false; // kembalikan false
 else
 {
 // jika tidak kosong hancurkan session tersebut
 unset($_SESSION[$nama_ses]);
 session_destroy();
 return true; // kembalikan nilai true
 }
}

```

```

// fungsi untuk menampilkan daftar polling sesuai parameter
function daftar_poll($tabel, $field)
{
 // lakukan query untuk mendapatkan daftar
 $hasil = mysql_query("SELECT * FROM $tabel");

 // tampilkan daftar dengan looping
 while ($data = mysql_fetch_array($hasil))
 {
 $daftar .= '<input type="radio" name="'.$field.'" value="'
 . $data[0].'">' . $data[0]. '
';

 }

 $daftar .= '
<input type="submit" value="POLLING">';

 // kembalikan hasil dari daftar
 return $daftar;
}

// fungsi untuk mengecek kecocokan alamat IP dan jenis polling
function cek_ip($jenis_poll, $letak='log/ip.dat')
{
 $ip_user = $_SERVER['REMOTE_ADDR']; // dapatkan alamat ip user

 // buka dengan file() untuk mengubah setiap baris menjadi array
 $daftar_ip = file($letak);
 // hitung jumlah baris berguna untuk looping for
 $jml_baris = count($daftar_ip);

 // lakukan looping untuk mencocokan setiap ip dan polling
 for ($i=0; $i<$jml_baris; $i++)
 {
 $bagian = explode('#-#', $daftar_ip[$i]); // pecah setiap bagian
 $ip = $bagian[0]; // untuk alamat ip
 $nama_poll = $bagian[1]; // untuk jenis polling

 // jika alamat IP dan jenis polling cocok maka ia pernah
 // mengikuti polling pada kategori tersebut
 // jadi kembalikan false
 if (ereg($ip_user, $ip) && ereg($jenis_poll, $nama_poll))
 return false;
 }

 return true; // jika tidak ada kesalahan kembalikan true
}

?>

```

Simpan pada direktori **polling\inc** dengan nama ***fungsi.php***.

### PENJELASAN SCRIPT

Karena sebagian besar fungsi sudah pernah kita buat. Maka hanya fungsi yang baru saja yang akan saya jelaskan. Fungsi yang baru adalah **balik\_md5()**. Fungsi ini memerlukan satu parameter yaitu string yang akan dienkripsi. Enkripsi yang digunakan adalah MD5 setelah dienkripsi kemudian urutan dari string chiper kita balik dengan menggunakan **strrev()**.

Fungsi berikutnya adalah **daftar\_poll()**. Fungsi ini mempunyai dua parameter yang keduanya wajib diisi. Parameter pertama adalah tabel yang digunakan, sedangkan yang kedua adalah nama untuk setiap data yang dipost. Untuk mendapatkan semua daftar polling

didatabase kita gunakan looping while. Karena bersifat memilih maka input yang kita gunakan adalah radio button.

Fungsi yang terakhir adalah `cek_ip()`. Fungsi ini memiliki dua parameter satu wajib diisi dan satu lagi opsional. Parameter pertama adalah untuk mengidentifikasi jenis polling yang akan kita cek. Sebagaimana kita tahu jika kita memberi nilai yes pada `$cek_ip` di file konfig.php. Maka user hanya boleh memilih satu kali saja pada satu jenis polling.  
`$_SERVER['REMOTE_ADDR']` merupakan array super global yang berisi alamat ip user yang mengakses website.

Parameter ke dua adalah untuk file yang kita gunakan. Jika tidak diisi defaultnya adalah log/ip.txt. Format dari isi file ip.txt adalah  
`alamat_ip_user#-#jenis_polling`. Karena dibuka dengan file maka setiap baris akan menjadi array dari `$daftar_ip`. Pada blok looping kita memecah setiap bagian dengan `explode()` untuk mendapatkan alamat ip dan jenis polling.

Kita mencocokkan alamat ip dan jenis polling menggunakan ereg. Jika alamat ip dan jenis polling cocok maka ia pernah mengikuti polling. Untuk itu kembalikan nilai false.

File yang akan kita buat berikutnya adalah file yang berisi class untuk menggambar grafik bar. Untuk membuatnya klik menu **File > New > PHP** lalu ketik kode berikut ini

Script dari file ***class\_polling.php***

---

```
<?php

/*
** Nama File.....: class_polling.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan....: class untuk menggambar grafik

// cegah pengaksesan langsung dari browser
if (eregi('class_polling.php', $_SERVER['PHP_SELF']))
 exit('Error: Akses Ditolak');

// buat class bernama polling
class polling
{
 // deklarasikan properti
 var $gambar, $lebar_max, $tabel, $hasil_query, $jml_kandidat;
 var $suara_terbanyak, $total_pemilih, $persen, $grafik;

 // buat metode untuk mendapatkan lokasi gambar
 function get_gambar($lokasi)
 {
 $this->gambar = $lokasi;
 }

 // metode untuk menentukan lebar maximal grafik
 function get_lebar_max($angka)
 {
 $this->lebar_max = $angka;
 }

 // metode untuk memilih tabel yang digunakan
 function get_tabel($nama_tabel)
 {
 $this->tabel = $nama_tabel;
 }
}
```

```

// metode untuk melakukan query, berguna untuk mendapatkan
// jumlah kandidat, suara terbanyak, total pemilih atau
// sekedar melakukan query biasa
function query_SQL($q = '')
{
 // jika parameter tidak diisi lakukan query berikut
 if ($q == '')
 $query = mysql_query("SELECT * FROM $this->tabel");
 else
 $query = mysql_query($q);

 $this->hasil_query = $query;

 return $this->hasil_query; // kembalikan hasil dari query
}

// metode untuk menentukan jumlah kandidat
function get_jml_kandidat()
{
 $hasil = mysql_num_rows($this->hasil_query);
 $this->jml_kandidat = $hasil;
}

// metode untuk mendapatkan kandidat yang memiliki suara terbanyak
function get_suara_terbanyak()
{
 $cari_suara = mysql_fetch_array($this->hasil_query);
 $this->suara_terbanyak = $cari_suara['suara'];
}

// metode untuk mendapatkan jumlah total pemilih
function get_total_pemilih()
{
 // lakukan looping untuk menjumlah setiap isi dari kolom suara
 // looping diulng sebanyak jumlah kandidat yang ada
 for ($i=0; $i<$this->jml_kandidat; $i++)
 {
 $kolom = mysql_fetch_array($this->hasil_query);
 $total += $kolom['suara'];
 // sama dengan $total = $total + $kolom['suara']
 }

 $this->total_pemilih = $total;

 // kembalikan nilai karena akan dicetak ke layar
 return $this->total_pemilih;
}

// metode untuk mendapatkan persentase masing2 kandidat
function get_persen($suara)
{
 // jika total pemilih sama dengan nol tetapkan persen = 0
 // untuk menghindari pembagian dengan nol
 // MATEMATIKA-nya
 // ****
 ** % = (jumlah suara kandidat : total pemilih) x 100 *
 // ****
}

if ($this->total_pemilih == 0)
 $this->persen = 0;
else
 $this->persen = round((($suara / $this->total_pemilih) * 100, 2));
 // 2 menunjukkan jumlah digit dibelakang koma

```

```

 // kembalikan nilai karena akan dicetak ke layar
 return $this->persen;
}

// metode untuk meggambar grafik
function gambar_grafik($suara)
{
 // untuk menentukan lebar masing2 grafik MATEMATIKA-nya adalah
 //*****
 ** lebar = (lebar maximal x jumlah suara) : suara terbanyak **
 //*****

 // jika suara terbanyak 0 tetapkan lebar = 0 untuk menghindari
 // pembagian dengan 0
 if ($this->suara_terbanyak == 0)
 $lebar = 0;
 else
 $lebar = round(($this->lebar_max * $suara) / $this-> suara_terbanyak);

 $this->grafik= '';

 // kembalikan nilai karena akan dicetak ke layar
 return $this->grafik;
}

} // akhir dari class
?>

```

---

Simpan dengan pada direktori **polling\inc** dengan nama ***class\_polling.php***.

### PENJELASAN SCRIPT

Pada class polling diatas langkah pertama adalah kita mendeklarasikan properti. Sebenarnya tanpa deklarasipun tidak masalah. Hal ini hanya agar penggerjaan kita lebih terstruktur.

Pada class ini terdapat sembilan metode. Namun kedelapan metode tujuan akhirnya adalah metode ke sembilan yaitu **gambar\_grafik()**. Pada suatu metode untuk merujuk hasil dari metode itu sendiri kita gunakan variabel istimewa yaitu **\$this**. Kemudian diikuti oleh tanda **->** dan properti yang digunakan.

Kita ambil contoh pada metode pertama yaitu **get\_gambar()**. Pada metode ini memiliki satu parameter yaitu lokasi file gambar yang digunakan. Lihat cuplikan metode **get\_gambar()** berikut

```

function get_gambar($lokasi)
{
 $this->gambar = $lokasi;
}

```

Jika metode ini dipanggil pada script nilai dari metode ini adalah parameter yang dimasukkan user. Jika kita ingin mendapatkan nilai **get\_gambar()** pada class itu sendiri maka kita gunakan **\$this->gambar** bukan memanggil metodenya.

Perhatikan pada metode **get\_persen()**. Metode ini digunakan untuk mendapatkan persentase dari masing-masing kandidat polling.

```

function get_persen($suara)
{
 if ($this->total_pemilih == 0)
 $this->persen = 0;
 else

```

```

 $this->persen = round(($suara / $this->total_pemilih) * 100, 2);

 return $this->persen;
}

```

Perhatikan bahwa untuk mendapatkan persentase, metode `get_persen()` bergantung pada nilai dari metode `get_total_pemilih()` yaitu

`$this->total_pemilih`. Untuk melakukan perhitungan kita memanggil fungsi `round()`. Kita mengembalikan nilai dari metode ini dengan `return` karena akan dicetak ke layar.

Metode yang terakhir yaitu `gambar_grafik()` merupakan metode yang kita gunakan untuk menggambar grafik. Logika kerjanya adalah kita menggunakan tag HTML `<img>` yang isinya adalah `$this->gambar` dan untuk lebar maksimalnya adalah sesuai dengan isi dari metode `get_lebar_max()` yaitu `$this->lebar_max`. Untuk menentukan lebar grafik dari masing-masing kandidat digunakan perhitungan sebagai berikut

```
lebar = (lebar maximal x jumlah suara) : suara terbanyak
```

Misalnya ada tiga kandidat **A**, **B**, dan **C**. Si A memperoleh **4 suara**, Si B memperoleh **2 suara** sedangkan Si C **6 suara**. Jadi suara terbanyak dari polling diatas adalah **6 suara**. Kita asumsikan lebar maximal grafiknya adalah 400, maka kandidat yang memperoleh suara tertinggi pasti lebar grafiknya adalah 400 karena:

```

A => (400 x 4) : 6 = 266 pixel
B => (400 x 2) : 6 = 133 pixel
C => (400 x 6) : 6 = 400 pixel (Terbukti)

```

Bagaimana sudah mengerti bukan?.

#### Akhir dari file yang berada pada direktori **inc**

---

File yang akan kita buat berikutnya adalah file utama untuk halaman administrator. File ini berfungsi untuk login, menampilkan menu dan logout. Klik menu **File > New > PHP** pada PHP DESIGNER 2005 lalu ketik kode berikut

#### Script dari file **index.php**

---

```

<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan....: Untuk memanage polling
***** */

// panggil session_start() karena berhubungan dengan session
session_start();

// panggil file header.php untuk menampilkan header halaman
include ('../inc/header.php');

// panggil file fungsi.php
include ('../inc/fungsi.php');

// dapatkan proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'login';

```

```

// filter setiap isi dari proses
$proses = filter_str($proses);

// koneksikan ke database
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'login':
 ****/

 // cek apakah admin sudah login atau belum
 // jika sudah tampilkan menu, jika belum tampilkan form login
 if (!cek_session('admin'))
 {
 echo '<h2>Admin Area - Mohon Login</h2>
 .<form action="index.php?proses=proses_login" method="post">
 .Username: <input type="text" name="username">

 .Password: <input type="password" name="password">

 .<input type="submit" value="LOGIN">
 .</form>
 .<p><< Back To Polling Page</p>';

 }
 else
 {
 // jika sudah terisi maka tampilkan menu untuk memanage polling
 echo '<h2>Selamat Datang di Admin Area</h2>
 .<p>Manaje Polling Pemain</p>
 .<p>Manaje Polling Klub
</p>
 .<p>Manaje Polling Band
</p>
 .<p>Logout
</p>';

 }

 break;
 ****/
}

case 'proses_login':
****/

// ambil data yang di-post sekaligus filter
$username = filter_str($_POST['username']);
$password = filter_str($_POST['password']);

// enkripsi password agar sama dengan yang ada didatabase
$password = balik_md5($password);

// lakukan query untuk mencocokan data
$hasil = mysql_query("SELECT * FROM admin WHERE username='$username'
 AND password='$password'");

// jika jumlah baris yang dikembalikan lebih dari nol maka cocok
if (mysql_num_rows($hasil) > 0)
{
 // buatkan session bernama admin
 $_SESSION['admin'] = $username;
 echo 'Login berhasil. Klik di sini untuk
 masuk ke admin area.';
}
else

```

```

echo 'Username atau Password salah!. Login
 kembali.';

break;
***** */

***** */
case 'logout':
***** */

// panggil fungsi logout() untuk menghancurkan session
if (!logout('admin'))
 echo 'Logout gagal. Login dulu.';
else
 echo 'Anda telah logout dari sistem. Login
 kembali.';

break;
***** */

***** */
case '__add_admin_to_database__':
***** */

/*
case ini berfungsi untuk memasukkan account administrator ke database
ini dikarenakan fungsi yang kita gunakan login adalah balik_md5()
dan untuk menghasilkan string chiper ini hanya bisa dilakukan lewat
script PHP bukan pada MySQL

untuk memanggil fungsi ini harus diketikkan langsung pada address bar
index.php?proses=__add_admin_to_database__ lalu ENTER

untuk mencegah eksloitasi sistem, kita tidak menyediakan form untuk
menambahkan account admin ke database melainkan langsung
melakukan query
*/

// tentukan username dan password yang diinginkan
$username = 'admin';
$password = balik_md5('__pas123__');

// lakukan query INSERT untuk memasukkan account ke database
$hasil = mysql_query("INSERT INTO admin VALUES('$username', '$password')");

if (!$hasil)
 echo 'Error: Gagal memasukkan ke database. Mungkin account
 sudah dimasukkan.';
else
 echo 'Account untuk administrator berhasil dimasukkan ke database.

 .Login';

break;
***** /

} // akhir dari switch

mysql_close(); // tutup koneksi ke MySQL

?>

</body>
</html>

```

Simpan pada direktori **polling\admin** dengan nama ***index.php***. Perhatikan bahwa bukan hanya script ini saja, script-script lainnya akan kita tulis dengan cara demikian agar mudah dipahami.

### PENJELASAN SCRIPT

Yang perlu saya jelaskan disini adalah case `__add_admin_to_database__`. Case ini kita buat karena untuk menghasilkan string chiper text yang berisi algoritma md5 yang susunannya dibalik hanya bisa dilakukan dengan fungsi `balik_md5()`. Tentu untuk memanggil fungsi tersebut hanya dapat dilakukan lewat script PHP.

Jadi untuk membuat account untuk administrator kita harus memanggil case ini dulu dengan cara

`http://localhost/webmaster/Bab4/hari-16-17/admin/index.php?proses=__add_admin_to_database__`.

Jika anda memanggil ulang proses ini (tekan Refresh atau F5) maka akan muncul pesan error. Ini dikarenakan sifat dari kolom username pada tabel admin adalah **primary key**. Yang berarti tidak boleh ada yang sama.

Untuk login gunakan username **admin** dan password **pas123**. Secara otomatis **pas123** akan dienkrip dengan fungsi `balik_md5()` sehingga akan menghasilkan string chiper yang sama dengan yang ada didatabase.



Gambar 4.5: Halaman utama untuk admin

File berikutnya adalah file yang kita gunakan untuk memanage polling untuk pemain terfavorit. File ini berfungsi untuk menampilkan, menambah dan menghapus daftar pemain yang menjadi subjek polling. Pada PHP DESIGNER 2005 klik menu **File > New > PHP**, lalu ketik kode berikut

#### Script dari file **pemain.php**

---

```
<?php
```

```

** Nama File.....: pemain.php ***
** Penulis.....: Rio Astamal ***
** Tanggal.....: 16-10-2005 ***
** Penjelasan....: untuk memanage polling pemain ***
*****/
```

```

session_start();

// panggil file header.php
include ('../inc/header.php');

// panggil file fungsi.php
include ('../inc/fungsi.php');

// cek isi dari proses
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

// filter isi dari proses
$proses = filter_str($proses);

// koneksi ke MySQL server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL Server.

 Cek kembali settingan untuk host, username, dan password');

// cek apakah administrator sudah login atau belum
if (!cek_session('admin'))
 echo 'Anda belum login. Login dulu.';
else
{

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'view':
 ****/

 echo '<h3>Daftar Polling Untuk Pemain Terfavorit</h3>';

 // tampilkan semua pemain dan hasil suaranya
 $hasil = mysql_query("SELECT * FROM poll_pemain");

 // cek jumlah baris yang dikembalikan
 if (mysql_num_rows($hasil) == 0)
 echo 'Tidak ada daftar pemain untuk polling.

 Tambah Daftar.';
 else
 {
 echo '<form action="pemain.php?proses=hapus" method="post">
 <table border="0" cellpadding="4" cellspacing="0">
 <tr bgcolor="#d7e7f3">
 <td>Nama Pemain</td>
 <td>Jumlah Suara</td>
 <td>Hapus?</td>
 </tr>';

 // lakukan looping untuk mendapatkan semua record
 while ($data = mysql_fetch_array($hasil))
 {
 echo '<tr><td>'. $data[0]. '</td><td align="center">'. $data[1]. '</td>' .
 '<td align="center"><input type="checkbox" name="hapus[]" '
 .' value="'. $data[0]. '"></td>';
 // hapus kita beri tanda [] karena bertipe array
 }
 }
}

```

```

echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="2"></td>'
. '<td><input type="submit" value="HAPUS"></td></tr>'
. '</table>'
. '</form>'
. '<p>Halaman Utama &nbsp'
. 'Tambah Daftar</p>' ;
}

break;
******/

*****/
case 'tambah':
*****/

// ambil data yang di post untuk menetapkan jumlah field
$jml_field = $_POST['jml_field'];

// tetapkan maksimal untuk polling 10 pemain
if ($jml_field > 10)
 exit('Anda itu ingin membingukan pemilih dengan membuat pilihan
 sebanyak '.$jml_field.'!. Maksimal adalah 10.');

echo '<h3>Tambah Nama Pemain untuk Polling</h3>'
. '<form action="pemain.php?proses=tambah" method="post">'
. 'Isi jumlah field lalu tekan TAMPILKAN.

'
. 'Jumlah field: <input type="text" name="jml_field" value="1"> '
. '&nbsp &nbsp <input type="submit" value="TAMPILKAN">'
. '</form>';

echo '<p><form action="pemain.php?proses=proses_tambah" method="post">' ;

// buat field untuk menambah daftar pemain sebanyak $jml_field
for ($i=0; $i<$jml_field; $i++)
{
 echo intval($i + 1). ' '
 . 'Nama Pemain: <input type="text" name="pemain[]">

';
}

echo '<input type="submit" value="TAMBAH">'
. '</form></p>'
. 'Halaman Utama &nbsp'
. 'Polling Pemain';

break;
*****/

*****/
case 'proses_tambah':
*****/

// ambil data yang dipost
$pemain = $_POST['pemain'];
if (count($pemain) == 0) // hitung jumlah data yang dikirim
 exit('Error: Tidak ada pemain yang dimasukkan.

 Kembali');

// buat variabel error berguna untuk mengidentifikasi error
$error = 0;

// lakukan looping untuk memasukkan setiap field ke database
foreach ($pemain as $nama)
{
 if ($nama == '')

```

```

 $error += 1; // artinya $error = $error + 1;
 else
 $hasil = mysql_query("INSERT INTO poll_pemain VALUES ('$nama', 0)");
 }

// jika $error tidak sama dengan nol berarti ada error
if ($error != 0)
 echo '<p>Ada '.$error.' field yang tidak dimasukkan ke database</p>';

if (!$hasil)
 echo 'Error: Gagal memasukkan data ke database.';
else
 echo '<p>Data berhasil dimasukkan ke database.
 Lihat';

break;
***** */

***** */
case 'hapus':
***** */

// ambil data dari case view
$hapus = $_POST['hapus'];

// hitung jumlah dari hapus jika nol maka tidak ada yang dicek
if (count($hapus) == 0)
 exit('Error: Tidak ada item yang dipilih.

 Kembali');

// lakukan looping untuk menghapus setiap item
foreach ($hapus as $sampah)
{
 // lakukan query DELETE untuk menghapus record
 $hasil = mysql_query("DELETE FROM poll_pemain WHERE pemain='".$sampah"'");
}

// jika ada kesalahan tampilkan
if (!$hasil)
 echo 'Error: Gagal menghapus data dari database

 Kembali';
else
 echo 'Data berhasil dihapus dari database.

 Polling Pemain';

break;
***** */

} // akhir dari switch
} // akhir dari else

mysql_close(); // tutup koneksi ke MySQL

?>

</body>
</html>

```

Simpan pada direktori **polling\admin** dengan nama **pemain.php**.

## PENJELASAN SCRIPT

Struktur dari file **pemain.php**, **klub.php**, dan **band.php** adalah sama. Jadi penjelasan dari file ini akan sama artinya dengan kedua file lainnya. Untuk menghapus data cara yang kita terapkan disini berbeda dengan sebelumnya. Kita menggunakan checkbox untuk memilih data mana saja yang ingin kita hapus lalu tekan HAPUS. Cara ini tentu lebih efisien ketimbang memberi link hapus pada setiap data. Gambar 4.6 menunjukkan tampilan daftar polling untuk pemain.

Nama Pemain	Jumlah Suara	Hapus?
Frank Lampard	7	<input type="checkbox"/>
Steven Gerard	6	<input type="checkbox"/>
David Beckham	22	<input type="checkbox"/>
Samuel Eto'o	3	<input type="checkbox"/>
Ronaldinho	13	<input type="checkbox"/>

**HAPUS**

Gambar 4.6: Daftar pemain untuk polling pada admin area

Untuk nama dari setiap item adalah `hapus[]`. Tanda [] menunjukkan bahwa data yang kita post berupa array. Jadi untuk menanganinya juga harus dengan perlakuan array. Agar setiap data yang diberi tanda cek yang terhapus maka kita gunakan looping foreach seperti berikut

```
foreach ($hapus as $sampah)
{
 // lakukan query DELETE untuk menghapus record
 $hasil = mysql_query("DELETE FROM poll_pemain WHERE pemain='\$sampah'");
}
```

Arti perintah looping diatas adalah untuk setiap isi dari `$hapus` sebagai `$sampah`. Query `DELETE` pada blok kode diatas akan diluang sebanyak pemain yang ingin dihapus.

Yang cukup menarik adalah case *tambah*. Kita dapat menentukan jumlah pemain yang akan dimasukkan sekaligus. Kita cukup mengisi field jumlah field lalu menekan tombol TAMPILKAN. Maka field Nama Pemain akan dibuat sebanyak angka yang kita masukkan tadi. Maksimal field yang dapat kita buat adalah 10. Jika lebih dari 10 maka saat pemrosesan akan dimunculkan pesan error.

Sama dengan hapus karena pemain bertipe array. Maka untuk memasukkannya pun harus dengan cara array. Memasukkan setiap pemain ke database kita gunakan looping foreach seperti berikut

```
foreach ($pemain as $nama)
{
 if ($nama == '')
 $error += 1; // artinya $error = $error + 1;
 else
 $hasil = mysql_query("INSERT INTO poll_pemain VALUES ('$nama', 0)");
}
```

Jadi untuk setiap isi dari array \$emain kita masukkan pada variabel \$nama. Jika ada field yang masing masing kosong tambah nilai \$error dengan 1. Jika \$error nilainya tidak sama dengan nol maka terdapat field yang tidak diisi. Untuk itu tampilkan pesan jumlah field yang tidak dimasukkan ke database.

Misalnya ada 3 field dan admin hanya mengisi satu maka pesan error untuk pemasukan database tidak ditampilkan. Sebaliknya jika admin tidak mengisi satu pun field maka pesan error database akan dimunculkan. Karena tidak ada satupun data yang dimasukkan ke database.

File berikutnya adalah file yang kita gunakan untuk memanage polling klub. Fungsi file ini adalah menampilkan, menambah dan menghapus daftar dari polling untuk klub. Klik **File > New > PHP** untuk membuat file PHP baru lalu ketik kode berikut

**Tips:** Anda dapat menggunakan teknik **copy** dan **paste** dari file pemain.php untuk membuat file ini. Untuk mengcopy sebaiknya lakukan case demi case untuk menghindari kesalahan. Lalu ubah konfigurasinya sesuai dengan link dan tabel untuk klub polling.

#### Script dari file **klub.php**

```
<?php

/*
** Nama File.....: pemain.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan....: untuk memanage polling klub

session_start();

// panggil file header.php
include ('../inc/header.php');

// panggil file fungsi.php
include ('../inc/fungsi.php');

// cek isi dari proses
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

// filter isi dari proses
$proses = filter_str($proses);

// koneksi ke MySQL server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL Server.

 Cek kembali settingan untuk host, username, dan password');

// cek apakah administrator sudah login atau belum
if (!cek_session('admin'))
 echo 'Anda belum login. Login dulu.';
else
{

// handel setiap proses dengan switch dan case
switch ($proses)
{

}
```

```

case 'view':
/************/

echo '<h3>Daftar Polling untuk Klub Terfavorit</h3>';

// tampilkan semua daftar untuk polling klub
$hasil = mysql_query("SELECT * FROM poll_klub");

// cek jumlah baris yang dikembalikan
if (mysql_num_rows($hasil) == 0)
 echo 'Tidak ada daftar klub untuk polling.

 Tambah Daftar.';
else
{
 echo '<form action="klub.php?proses=hapus" method="post">
 <table border="0" cellpadding="4" cellspacing="0">
 <tr bgcolor="#d7e7f3">
 <td>Nama Klub</td>
 <td>Jumlah Suara</td>
 <td>Hapus?</td>
 </tr>';

 // lakukan looping untuk mendapatkan semua record
 while ($data = mysql_fetch_array($hasil))
 {
 echo '<tr><td>'.$data[0].'</td><td align="center">'.$data[1].'</td>
 <td align="center"><input type="checkbox" name="hapus[]" '
 .'value="'.$data[0].'"></td>';
 // hapus kita beri tanda [] karena bertipe array
 }

 echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="2"></td>
 <td><input type="submit" value="HAPUS"></td></tr>
 </table>
 </form>
 <p>Halaman Utama &nbsp
 Tambah Daftar</p>';
}

// akhir dari else

break;
/************/

```

---

```

/************/
case 'tambah':
/************/

// ambil data yang di post untuk menetapkan jumlah field
$jml_field = $_POST['jml_field'];

// tetapkan maksimal untuk polling 10 klub
if ($jml_field > 10)
 exit('Anda itu ingin membingungkan pemilih dengan membuat pilihan
 sebanyak '.$jml_field.'!. Maksimal adalah 10.');

echo '<h3>Tambah Nama Klub untuk Polling</h3>
 <form action="klub.php?proses=tambah" method="post">
 Isi jumlah field lalu tekan TAMPILKAN.

 Jumlah field: <input type="text" name="jml_field" value="0"> '
 .'&nbsp&nbsp<input type="submit" value="TAMPILKAN">
 </form>';

echo '<p><form action="klub.php?proses=proses_tambah" method="post">';


```

```

// buat field untuk menambah daftar klub sebanyak $jml_field
for ($i=0; $i<$jml_field; $i++)
{
 echo intval($i + 1). ' '
 .'Nama Klub: <input type="text" name="klub[]">

';
}

echo '<input type="submit" value="TAMBAH">'
 .'</form></p>'
 .'Halaman Utama &nbsp '
 .'Polling Klub';

break;
/************/

/************/
case 'proses_tambah':
/************/

// ambil data yang dipost
$klub = $_POST['klub'];
if (count($klub) == 0) // hitung jumlah data yang dikirim
 exit('Error: Tidak ada klub yang dimasukkan.

 Kembali');

// buat variabel error berguna untuk mengidentifikasi error
$error = 0;

// lakukan looping untuk memasukkan setiap field ke database
foreach ($klub as $tim)
{
 if ($tim == '')
 $error += 1; // artinya $error = $error + 1;
 else
 $hasil = mysql_query("INSERT INTO poll_klub VALUES ('$tim', 0)");
}

// jika $error tidak sama dengan nol berarti ada error
if ($error != 0)
 echo '<p>Ada '.$error.' field yang tidak dimasukkan ke database</p>';

if (!$hasil)
 echo 'Error: Gagal memasukkan data ke database.';
else
 echo '<p>Data berhasil dimasukkan ke database.
 Lihat';

break;
/************/

/************/
case 'hapus':
/************/

// ambil data dari case view
$hapus = $_POST['hapus'];

// hitung jumlah dari hapus jika nol maka tidak ada yang dicek
if (count($hapus) == 0)
 exit('Error: Tidak ada item yang dipilih.

 Kembali');

// lakukan looping untuk menghapus setiap item
foreach ($hapus as $sampah)

```

```

{
 // lakukan query DELETE untuk menghapus record
 $hasil = mysql_query("DELETE FROM poll_klub WHERE klub='\$sampah' ");
}

// jika ada kesalahan tampilkan
if (!$hasil)
 echo 'Error: Gagal menghapus data dari database

' .
 . 'Kembali';
else
 echo 'Data berhasil dihapus dari database.

' .
 . 'Polling Klub';

break;
***** */

} // akhir dari switch

} // akhir dari else

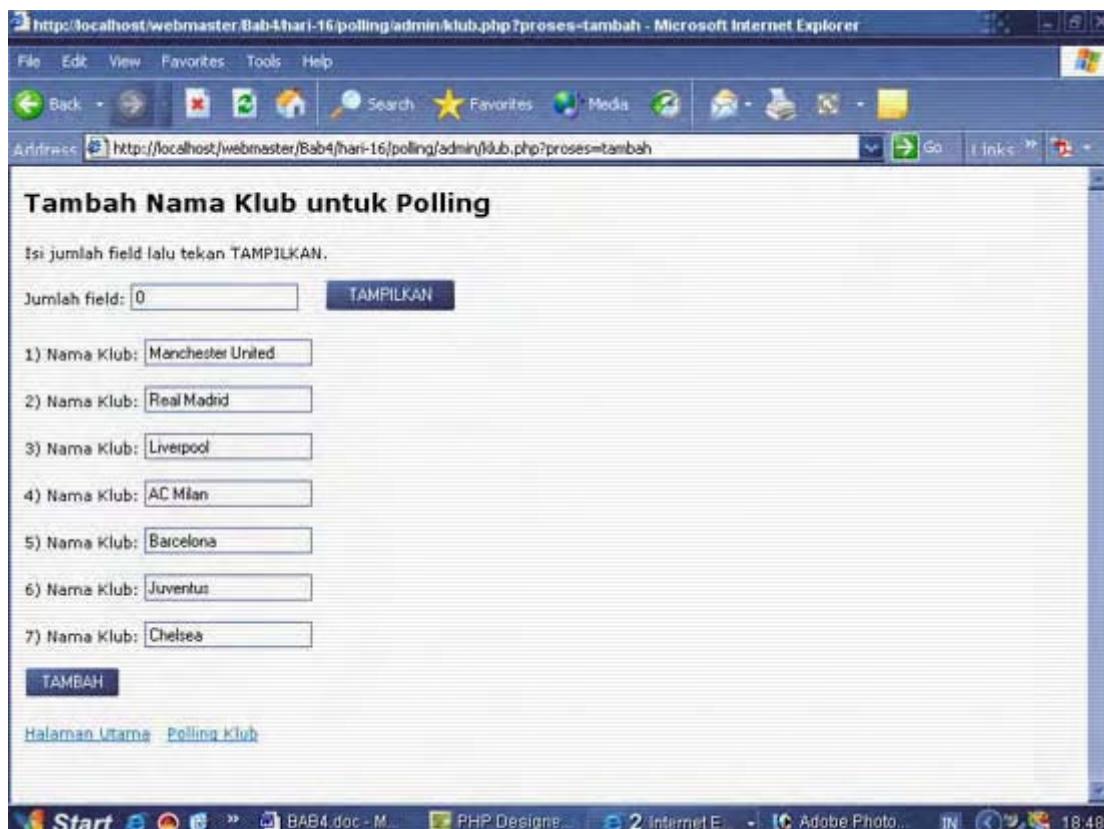
?>

</body>
</html>

```

Simpan pada direktori **polling\admin** dengan nama **klub.php**.

Struktur kode dari file *klub.php* sama persis dengan file. Jadi saya rasa tidak ada yang perlu dijelaskan lagi. Gambar 4.7 menunjukkan proses pemambahan daftar klub yang akan menjadi subjek polling.



Gambar 4.7: Proses saat menambah daftar polling

Sama dengan file **pemain.php** dan **klub.php**. Struktur dari file **band.php** sama persis dengan keduanya. Jadi untuk membuatnya gunakan teknik *copy* dan *paste* seperti pada Tips. Buat file baru pada PHP DESIGNER 2005 salin kode berikut

#### Script dari file **band.php**

---

```
<?php

/*
** Nama File.....: pemain.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan....: untuk memanage polling band
*/

session_start();

// panggil file header.php
include ('../inc/header.php');

// panggil file fungsi.php
include ('../inc/fungsi.php');

// cek isi dari proses
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

// filter isi dari proses
$proses = filter_str($proses);

// koneksi ke MySQL server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL Server.

 Cek kembali settingan untuk host, username, dan password');

// cek apakah administrator sudah login atau belum
if (!cek_session('admin'))
 echo 'Anda belum login. Login dulu.';
else
{
 // handel setiap proses dengan switch dan case
 switch ($proses)
 {

/*
*/
 case 'view':
/*
*/
 echo '<h3>Daftar Polling Untuk Band Terfavorit</h3>';

 // tampilkan semua pemain dan hasil suaranya
 $hasil = mysql_query("SELECT * FROM poll_band");

 // cek jumlah baris yang dikembalikan
 if (mysql_num_rows($hasil) == 0)
 echo 'Tidak ada daftar band untuk polling.

 Tambah Daftar.';
 else
 {
 echo '<form action="band.php?proses=hapus" method="post">
 .<table border="0" cellpadding="4" cellspacing="0">'
```

```

. '<tr bgcolor="#d7e7f3">'

. '<td>Nama Band</td>'

. '<td>Jumlah Suara</td>'

. '<td>Hapus?</td>'

. '</tr>' ;

// lakukan looping untuk mendapatkan semua record

while ($data = mysql_fetch_array($hasil))

{

 echo '<tr><td>'. $data[0]. '</td><td align="center">'. $data[1]. '</td>'

 . '<td align="center"><input type="checkbox" name="hapus[]" '

 . 'value="'. $data[0]. '"></td>' ;

 //hapus kita beri tanda [] karena bertipe array

}

echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="2"></td>'

. '<td><input type="submit" value="HAPUS"></td></tr>'

. '</table>'

. '</form>'

. '<p>Halaman Utama &nbsp

. 'Tambah Daftar</p>' ;

}

break;

/*************/

/*************/

case 'tambah':

/*************/

// ambil data yang di post untuk menetapkan jumlah field

$jml_field = $_POST['jml_field'];

// tetapkan maksimal untuk polling 10 pemain

if ($jml_field > 10)

 exit('Anda itu ingin membingungkan pemilih dengan membuat pilihan

 sebanyak '. $jml_field. '!. Maksimal adalah 10.');

echo '<h3>Tambah Nama Band untuk Polling</h3>'

. '<form action="band.php?proses=tambah" method="post">'

. 'Isi jumlah field lalu tekan TAMPILKAN.

'

. 'Jumlah field: <input type="text" name="jml_field" value="0"> '

. ' <input type="submit" value="TAMPILKAN">'

. '</form>' ;

echo '<p><form action="band.php?proses=proses_tambah" method="post">' ;

// buat field untuk menambah daftar pemain sebanyak $jml_field

for ($i=0; $i<$jml_field; $i++)

{

 echo intval($i + 1). '

 . 'Nama Band: <input type="text" name="band[]">

' ;

}

echo '<input type="submit" value="TAMBAH">'

. '</form></p>'

. 'Halaman Utama &nbsp

. 'Polling Band' ;

break;

/*************/

/*************/

case 'proses_tambah':

```

```

// ambil data yang dipost
$band = $_POST['band'];
if (count($band) == 0) // hitung jumlah data yang dikirim
 exit('Error: Tidak ada band yang dimasukkan.

 Kembali');

// buat variabel error berguna untuk mengidentifikasi error
$error = 0;

// lakukan looping untuk memasukkan setiap field ke database
foreach ($band as $artis)
{
 if ($artis == '')
 $error += 1; // artinya $error = $error + 1;
 else
 $hasil = mysql_query("INSERT INTO poll_band VALUES ('$artis', 0)");
}

// jika $error tidak sama dengan nol berarti ada error
if ($error != 0)
 echo '<p>Ada '.$error.' field yang tidak dimasukkan ke database</p>';

if (!$hasil)
 echo 'Error: Gagal memasukkan data ke database.';
else
 echo '<p>Data berhasil dimasukkan ke database.
 Lihat';

break;

case 'hapus':

// ambil data dari case view
$hapus = $_POST['hapus'];

// hitung jumlah dari hapus jika nol maka tidak ada yang dicek
if (count($hapus) == 0)
 exit('Error: Tidak ada item yang dipilih.

 Kembali');

// lakukan looping untuk menghapus setiap item
foreach ($hapus as $sampah)
{
 // lakukan query DELETE untuk menghapus record
 $hasil = mysql_query("DELETE FROM poll_band WHERE band='$sampah' ");
}

// jika ada kesalahan tampilkan
if (!$hasil)
 echo 'Error: Gagal menghapus data dari database

 .Kembali';
else
 echo 'Data berhasil dihapus dari database.

 .Polling Band';

break;

} // akhir dari switch

```

```

} // akhir dari else

?>

</body>
</html>

```

Simpan pada direktori **polling\admin** dengan nama ***band.php***.

Akhir dari file yang berada pada direktori **admin**.

File-file berikutnya yang akan kita bahas adalah file pada direktori **polling**. File pertama yang akan kita buat adalah file utama untuk website NgawurPolling. Untuk membuatnya klik **File > New > PHP** lalu ketik kode berikut

Script dari file ***index.php***

```

<?php

/*
** Nama File....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan...: halaman utama untuk ngawur polling

// panggil file header.php
include ('inc/header.php');

// panggil file fungsi.php
include ('inc/fungsi.php');

// dapatkan konfigurasi dari file konfig.php
include ('inc/konfig.php');

// panggil file class_polling.php
include ('inc/class_polling.php');

// dapatkan data dari proses pada URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'pilihan';

// filter isi dari proses
$proses = filter_str($proses);

// koneksi ke MySQL server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username, dan password');

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*
 case 'pilihan':
 */

echo '<h2>Selamat Datang di NgawurPolling.com</h2>
 .'<p>Silahkan pilih polling yang sesuai dengan keinginan anda.

';

```

```

.' Daftar Polling untuk Minggu ini: </p>
.' <p>Polling Pemain Terfavorit</p>
.' [Hasil Polling]</p>
.' <p>Polling Klub Terfavorit
.' [Hasil Polling]</p>
.' <p>Polling Band Terfavorit
.' [Hasil Polling]<p>
.' <p>[Hasil Polling Semua Kategori]</p>
.' <p align="center">© 2005 ngawurpolling.com</p>

break;
*****/
*****/
case 'pemain':
*****/

// siapkan form
echo '<h2>Polling Pemain Terfavorit</h2>
.' <p>Siapakah Pemain Favorit Anda?</p>
.' <form action="poll_pemain.php?proses=proses_poll" method="post">';

// panggil fungsi daftar_POLL() untuk menampilkan daftar pemain
echo daftar_POLL('poll_pemain', 'pemain');

echo '</form>
.' <p>Tekan tombol POLLING untuk memulai proses polling.</p>
.' Home';

break;
*****/
*****/
case 'klub':
*****/

// siapkan form
echo '<h2>Polling Klub Terfavorit</h2>
.' <p>Manakah Klub Favorit Anda?</p>
.' <form action="poll_klub.php?proses=proses_poll" method="post">';

// panggil fungsi daftar_POLL() untuk menampilkan daftar klub
echo daftar_POLL('poll_klub', 'klub');

echo '</form>
.' <p>Tekan tombol POLLING untuk memulai proses polling.</p>
.' Home';

break;
*****/
*****/
case 'band':
*****/

// siapkan form
echo '<h2>Polling Band Terfavorit</h2>
.' <p>Manakah Band Favorit Anda?</p>
.' <form action="poll_band.php?proses=proses_poll" method="post">';

// panggil fungsi daftar_POLL() untuk menampilkan daftar klub
echo daftar_POLL('poll_band', 'band');

echo '</form>

```

```

. '<p>Tekan tombol POLLING untuk memulai proses polling.</p>'
. 'Home';

break;

}
?>

```

Simpan pada direktori **polling** dengan nama **index.php**. Output dari file ini akan terlihat seperti gambar 4.8.



Gambar 4.8: Tampilan halaman utama NgawurPolling

### PENJELASAN SCRIPT

Tidak ada yang menarik dari script ini. Yang perlu anda perhatikan adalah case pemain, klub dan band. Pada case-case tersebut kita memanggil fungsi `daftar_poll()` untuk menampilkan daftar kandidat untuk polling. Cuplikan script berikut akan menampilkan daftar kandidat untuk polling pemain.

```
echo daftar_poll('poll_pemain', 'pemain');
```

Pada fungsi diatas kita menetapkan parameter untuk tabel yang diisi adalah `poll_pemain`. Isi dari parameter ke dua yaitu `pemain` menunjukkan nama dari data yang kita post.

File berikutnya adalah `poll_pemain.php`. Fungsi file ini adalah untuk memproses form polling pemain yang dipost dari file index.php. Selain itu file ini juga berfungsi untuk menampilkan hasil dari polling. Hasil yang ditampilkan berupa persentase dan grafik. Klik **File > New > PHP** untuk membuat file PHP baru lalu ketik kode berikut

#### Script dari file **poll\_pemain.php**

```

<?php

/*
** Nama File....: poll_pemain.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan...: untuk memproses polling dan menampilkan
** hasil polling pemain.
*/
// panggil fungsi output buffering karena berhubungan dengan
// cookie
ob_start();

```

```

// panggil file header.php
include ('inc/header.php');

// panggil file fungsi.php
include ('inc/fungsi.php');

// dapatkan konfigurasi dari file konfig.php
include ('inc/konfig.php');

// panggil file class_polling.php
include ('inc/class_polling.php');

// dapatkan data dari proses pada URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view_POLL';

// filter isi dari proses
$proses = filter_str($proses);

// koneksi ke MySQL server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username, dan password');

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'view_POLL':
 ****/

echo '<h2>Hasil Polling Pemain Terfavorit</h2>
 .<table border="0" cellpadding="4" cellspacing="4">
 .<tr bgcolor="#d7e7f3">
 .'<td>Nama Pemain<td>Grafik</td>
 .'<td>Persentase (%)</td><td>Jumlah Pemilih</td></tr>
 .</table>';

// panggil class polling untuk menggambar grafik
$poll = new polling();
$poll->get_gambar($gambar['pemain']); // letaknya di konfig.php
$poll->get_lebar_max($lebar_max); // letaknya di konfig.php
$poll->get_tabel('poll_pemain'); // pilih tabel yang digunakan

// lakukan query untuk mendapatkan jumlah kandidat dan suara terbanyak
// urutkan secara DESCENDING agar yang suaranya terbanyak berada
// pada record paling atas, digunakan untuk menghitung lebar grafik
$poll->query_SQL("SELECT * FROM poll_pemain ORDER BY suara DESC");
$poll->get_jml_kandidat(); // hitung jumlah kandidat
$poll->get_suara_terbanyak(); // dapatkan suara yang terbanyak

// lakukan query ulang
$poll->query_SQL();
$total_pml = $poll->get_total_pemilih(); // hitung jumlah total pemilih

// lakukan looping untuk menggambar setiap grafik
$hasil = $poll->query_SQL();
while ($data = mysql_fetch_array($hasil))
{
 $kandidat = $data[0]; // nama masing2 kandidat
 $suara = $data[1]; // suara masing2 kandidat
 $persen = $poll->get_persen($suara); // persentase masing2 kandidat
}

```

```

$grafik = $poll->gambar_grafik($suara); // grafik dari masing2 kandidat
echo '<tr><td>'.$kandidat.'</td><td>'.$grafik.'</td>' .
 '<td>'.$persen.'%</td><td>'.$suara.'</td></tr>';
}

// tutup table
echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="4"></td></tr>' .
 '</table>' .
 '<p>Total Pemilih: '.$total_pml.' orang.</p>' .
 '<p>Home &nbsp' .
 'Ikuti Polling</p>';

break;
***** */

***** */
case 'proses_poll':
***** */

$pesan_error = 'Maaf, Anda hanya boleh mengikuti polling pemain
terfavorit satu kali saja. Home';

// cek apakah user boleh memilih lebih dari satu kali
// settingan ada di konfig.php
// cek settingan cek_cookie pada konfig.php jika yes maka cek
if ($cek_cookie == 'yes')
{
// jika isi cookie poll_pemain telah diset maka user pernah polling
if (isset($_COOKIE['poll_pemain']))
 exit($pesan_error);
}

if ($cek_ip == 'yes') // jika yes cek alamat ip
{
 // cek ip dan jenis pollingnya adalah poll_pemain
 if (!cek_ip('poll_pemain')) // panggil fungsi cek_ip()
 exit($pesan_error);
}

// lanjutkan memproses jika user diperbolehkan polling
// ambil data dari form polling
$pemain = $_POST['pemain'];

if ($pemain == '')
 exit('Error: Anda belum memilih pemain.
 Kembali');

// lakukan query UPDATE untuk mengupdate isi dari pemain yang dipilih
$hasil = mysql_query("UPDATE poll_pemain SET suara=suara+1 WHERE
 pemain='".$pemain"'");
if (!$hasil)
 echo 'Error: Gagal memasukkan data ke database.

' .
 'Home';
else
{
 echo 'Terima Kasih karena telah mengikuti polling kami.

' .
 'Hasil Polling';

 // jika tidak ada cookie yang bername poll_pemain maka buat
 if (!isset($_COOKIE['poll_pemain']))
 setcookie('poll_pemain', '1', $masa_aktif);

 // catat IP dan jenis polling ke file ip.dat
}

```

```

$buka_file = fopen('log/ip.dat', 'a');
$isi = $_SERVER['REMOTE_ADDR'].'#-#poll_pemain'."\n";
fwrite($buka_file, $isi);
fclose($buka_file);

}

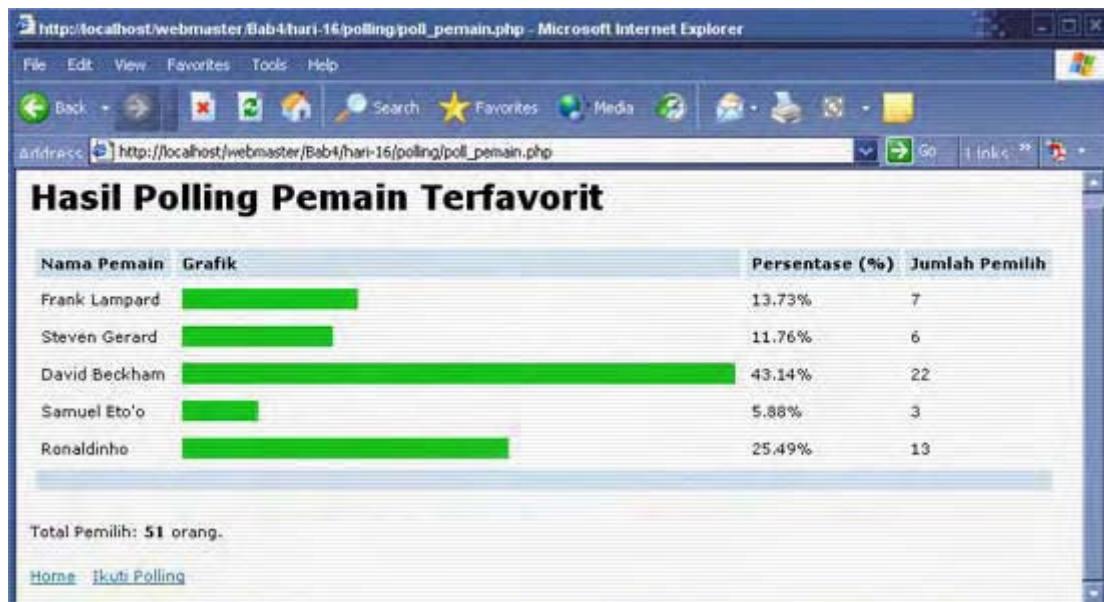
break;
***** */

}

?>

```

Simpan pada direktori **polling** dengan nama **poll\_pemain.php**. Hasil polling pemain ditunjukkan oleh gambar 4.9.



Gambar 4.9: Hasil polling pemain terfavorit

### PENJELASAN SCRIPT

Yang menarik adalah pada case pada view poll. Pada case ini kita membuat objek untuk membuat grafik. Objek ini merupakan implementasi dari class polling. Kita membuat objek ini dengan cara memanggil class polling seperti berikut

```
$poll = new polling;
```

Metode yang perlu anda perhatikan adalah metode untuk mendapatkan suara terbanyak. Untuk mendapatkan metode ini kita harus memanggil terlebih dahulu metode `query_SQL()` yang akan mengurut data secara descending yaitu mulai dari yang terbesar ke yang terekecil. Lihat cuplikan kode berikut

```
$poll->query_SQL("SELECT * FROM poll_pemain ORDER BY suara DESC");
$poll->get_jml_kandidat(); // hitung jumlah kandidat
$poll->get_suara_terbanyak(); // dapatkan suara yang terbanyak
```

Untuk mendapatkan suara terbanyak kita cukup mengambil data pada record pertama. Ini sudah dilakukan oleh metode `get_suara_terbanyak()`.

Untuk menampilkan gambar grafik, presentase dan jumlah pemilih masing-masing kandidat kita gunakan looping while. Lihat cuplikan script berikut

```
$hasil = $poll->query_SQL();
while ($data = mysql_fetch_array($hasil))
```

Arti \$poll->query\_SQL() diatas sama dengan `SELECT * FROM poll_pemain` karena pada metode `get_tabel()` kita isikan parameternya `poll_pemain`. Setelah itu kita gunakan looping while untuk mendapatkan daftar seluruh kandidat. Didalam blok kode looping while kita memanggil metode `get_persen()` dan `gambar_grafik()` untuk mendapatkan persentase dan grafik dari masing-masing kandidat.

Case berikutnya adalah `proses_poll`. Pada case ini kita memberikan ijin kepada user untuk mengikuti polling lebih dari satu kali atau tidak. Hal tersebut sesuai dengan settingan yang ada di file `konfig.php` yaitu isi dari `$cek_cookie` atau `$cek_ip`. Kelemahan dari pengecekan menggunakan cookie adalah user dapat menghapus cookie tersebut lewat browser sehingga ia dapat mengikuti polling ulang.

Akan tetapi jika menggunakan metode cek ip. User tidak dapat menghapus daftar ip-nya karena file tersebut berada di server. Sehingga jika user sudah pernah mengikuti polling maka ia tidak bisa lagi menikutinya ulang.

Pada kode diatas kita mengecek isi dari `$cek_cookie`. Jika yes maka cek isi cookie. Jika ternyata ada cookie bernama `poll_pemain` maka keluar dari rutin program dan tampilkan pesan error. Jika kita mengecek berdasarkan alamat IP. Maka kita memanggil fungsi `cek_ip()` yang pernah kita buat sebelumnya.

```
if (!cek_ip('poll_pemain')) // panggil fungsi cek_ip()
 exit($pesan_error);
```

Pada kode diatas secara default fungsi `cek_ip()` akan mengecek isi dari file ip.dat pada direktori `log` karena pada parameter kita tidak menentukannya. `poll_pemain` menunjukkan jenis polling yang kita cek. Jika ip dan jenis polling cocok maka user tersebut sudah pernah polling. Jadi tampilkan pesan error.

Jika tidak ada error maka lanjutkan memproses data. Jika data berhasil dimasukkan ke database buat cookie yang bernama `poll_pemain` yang isinya satu. Jika memang cookie tersebut belum ada. Kemudian catat alamat IP user dan jenis pollingnya ke file ip.dat dengan format IP#-#Jenis\_Polling.

File berikutnya yaitu `poll_klub.php` strukturnya sama dengan `poll_pemain.php`. Oleh karena itu penjelasannya pun sama. Anda dapat menggunakan trik copy – paste untuk membuat file `poll_klub.php`. Namun dengan catatan ubah konfigurasinya sesuai dengan polling klub. Klik **File > New > PHP DESIGNER** lalu ketik kode berikut

#### Script dari file `poll_klub.php`

---

```
<?php

/*
** Nama File....: poll_klub.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan...: untuk memproses polling dan menampilkan
** hasil polling klub.
*/

// panggil fungsi output buffering karena berhubungan dengan
// cookie
ob_start();

// panggil file header.php
include ('inc/header.php');

// panggil file fungsi.php
```

```

include ('inc/fungsi.php');

// dapatkan konfigurasi dari file konfig.php
include ('inc/konfig.php');

// panggil file class_polling.php
include ('inc/class_polling.php');

// dapatkan data dari proses pada URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view_POLL';

// filter isi dari proses
$proses = filter_str($proses);

// koneksi ke MySQL server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username, dan password');

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'view_POLL':
 ****/

echo '<h2>Hasil Polling Klub Terfavorit</h2>
 .<table border="0" cellpadding="4" cellspacing="4">
 .<tr bgcolor="#d7e7f3">
 .<td>Nama Klub<td>Grafik</td>
 .<td>Persentase (%)</td><td>Jumlah Pemilih</td>
 .</tr>';

// panggil class polling untuk menggambar grafik
$poll = new polling;
$poll->get_gambar($gambar['klub']); // letaknya di konfig.php
$poll->get_lebar_max($lebar_max); // letaknya di konfig.php
$poll->get_tabel('poll_klub'); // pilih tabel yang digunakan

// lakukan query untuk mendapatkan jumlah kandidat dan suara terbanyak
// urutkan secara DESCENDING agar yang suaranya terbanyak berada
// pada record paling atas, digunakan untuk menghitung lebar grafik
$poll->query_SQL("SELECT * FROM poll_klub ORDER BY suara DESC");
$poll->get_jml_kandidat(); // hitung jumlah kandidat
$poll->get_suara_terbanyak(); // dapatkan suara yang terbanyak

// lakukan query ulang
$poll->query_SQL();
$total_pml = $poll->get_total_pemilih(); // hitung jumlah total pemilih

// lakukan looping untuk menggambar setiap grafik
$hasil = $poll->query_SQL();
while ($data = mysql_fetch_array($hasil))
{
 $kandidat = $data[0]; // nama masing2 kandidat
 $suara = $data[1]; // suara masing2 kandidat
 $persen = $poll->get_persen($suara); // persentase masing2 kandidat
 $grafik = $poll->gambar_grafik($suara); // grafik dari masing2 kandidat

 echo '<tr><td>' . $kandidat . '</td><td>' . $grafik . '</td>
 .<td>' . $persen . '%</td><td>' . $suara . '</td></tr>';
}

```

```

}

// tutup table
echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="4"></td></tr>
. '</table>
. '<p>Total Pemilih: '.$total_pml.' orang.</p>'
. '<p>Home &nbsp '
. 'Ikuti Polling</p>';

break;
***** */

***** */
case 'proses_poll':
***** */

// cek apakah user boleh memilih lebih dari satu kali
// settingan ada di konfig.php
$pesan_error = 'Maaf, Anda hanya boleh mengikuti polling klub
terfavorit satu kali saja. Home';

// cek settingan cek_cookie pada konfig.php jika yes maka cek
if ($cek_cookie == 'yes')
{
// jika isi cookie poll_klub telah diset maka user pernah polling
if (isset($_COOKIE['poll_klub']))
exit($pesan_error);
}

if ($cek_ip == 'yes') // cek alamat IP
{
// cek ip dan jenis pollingnya adalah poll_klub
if (!cek_ip('poll_klub')) // panggil fungsi cek_ip()
exit($pesan_error);
}

// lanjutkan memproses jika user diperbolehkan polling
// ambil data dari form polling
$klub = $_POST['klub'];

if ($klub == '')
exit('Error: Anda belum memilih klub.
Kembali');

// lakukan query UPDATE untuk mengupdate isi dari pemain yang dipilih
$hasil = mysql_query("UPDATE poll_klub SET suara=suara+1 WHERE
klub='".$klub"'");
if (!$hasil)
echo 'Error: Gagal memasukkan data ke database.

.Home';
else
{
echo 'Terima Kasih karena telah mengikuti polling kami.

.Hasil Polling';

// jika tidak ada cookie yang bername poll_klub maka buat
if (!isset($_COOKIE['poll_klub']))
setcookie('poll_klub', '1', $masa_aktif);

// catat IP dan jenis polling ke file ip.dat
$buka_file = fopen('log/ip.dat', 'a');
$isi = $_SERVER['REMOTE_ADDR'].' #-#poll_klub'."\n";
fwrite($buka_file, $isi);
fclose($buka_file);
}

```

```

}
break;
***** */

} // akhir dari switch

?>

</body>
</html>

```

Simpan pada direktori **polling** dengan nama **poll\_klub.php**. Script ini secara struktur sama dengan file **poll\_pemain.php**. Output dari file ini ditunjukkan oleh gambar 4.10.



Gambar 4.10: Hasil polling klub terfavorit

File berikutnya adalah **poll\_band.php**. Fungsi file ini untuk memproses dan menampilkan hasil polling band terfavorit. Langsung saja, klik **File > New > PHP** lalu ketik kode berikut:

#### Script dari file **poll\_band.php**

```

<?php

/*
** Nama File....: poll_band.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan...: untuk memproses polling dan menampilkan
** hasil polling band.
*/

// panggil fungsi output buffering karena berhubungan dengan
// cookie
ob_start();

// panggil file header.php
include ('inc/header.php');

// panggil file fungsi.php

```

```

include ('inc/fungsi.php');

// dapatkan konfigurasi dari file konfig.php
include ('inc/konfig.php');

// panggil file class_polling.php
include ('inc/class_polling.php');

// dapatkan data dari proses pada URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view_POLL';

// filter isi dari proses
$proses = filter_str($proses);

// koneksi ke MySQL server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username, dan password');

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'view_POLL':
 ****/

echo '<h2>Hasil Polling Band Terfavorit</h2>';
 . '<table border="0" cellpadding="4" cellspacing="4">'
 . '<tr bgcolor="#d7e7f3">'
 . '<td>Nama Band<td>Grafik</td>'
 . '<td>Persentase (%)</td><td>Jumlah Pemilih</td>'
 . '</tr>';

// panggil class polling untuk menggambar grafik
$poll = new polling;
$poll->get_gambar($gambar['band']); // letaknya di konfig.php
$poll->get_lebar_max($lebar_max); // letaknya di konfig.php
$poll->get_tabel('poll_band'); // pilih tabel yang digunakan

// lakukan query untuk mendapatkan jumlah kandidat dan suara terbanyak
// urutkan secara DESCENDING agar yang suaranya terbanyak berada
// pada record paling atas, digunakan untuk menghitung lebar grafik
$poll->query_SQL("SELECT * FROM poll_band ORDER BY suara DESC");
$poll->get_jml_kandidat(); // hitung jumlah kandidat
$poll->get_suara_terbanyak(); // dapatkan suara yang terbanyak

// lakukan query ulang
$poll->query_SQL();
$total_pml = $poll->get_total_pemilih(); // hitung jumlah total pemilih

// lakukan looping untuk menggambar setiap grafik
$hasil = $poll->query_SQL();
while ($data = mysql_fetch_array($hasil))
{
 $kandidat = $data[0]; // nama masing2 kandidat
 $suara = $data[1]; // suara masing2 kandidat
 $persen = $poll->get_persen($suara); // persentase masing2 kandidat
 $grafik = $poll->gambar_grafik($suara); // grafik dari masing2 kandidat

 echo '<tr><td>' . $kandidat . '</td><td>' . $grafik . '</td>'
 . '<td>' . $persen . '%</td><td>' . $suara . '</td></tr>';
}

```

```

}

// tutup table
echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="4"></td></tr>
. '</table>
. '<p>Total Pemilih: ' . $total_pml . ' orang.</p>'
. '<p>Home &nbsp
. 'Ikuti Polling</p>';

break;
***** */

***** */
case 'proses_poll':
***** */

// cek apakah user boleh memilih lebih dari satu kali
// settingan ada di konfig.php
$pesan_error = 'Maaf, Anda hanya boleh mengikuti polling band
terfavorit satu kali saja. Home';

// cek settingan cek_cookie pada konfig.php jika yes maka cek
if ($cek_cookie == 'yes')
{
// jika isi cookie poll_klub telah diset maka user pernah polling
if (isset($_COOKIE['poll_band']))
exit($pesan_error);
}

if ($cek_ip == 'yes') // cek alamat IP
{
// cek ip dan jenis pollingnya adalah poll_band
if (!cek_ip('poll_band')) // panggil fungsi cek_ip()
exit($pesan_error);
}

// lanjutkan memproses jika user diperbolehkan polling
// ambil data dari form polling
$band = $_POST['band'];

if ($band == '')
exit('Error: Anda belum memilih band.
Kembali');

// lakukan query UPDATE untuk mengupdate isi dari band yang dipilih
$hasil = mysql_query("UPDATE poll_band SET suara=suara+1 WHERE
band='$band'");
if (!$hasil)
echo 'Error: Gagal memasukkan data ke database.

.Home';
else
{
echo 'Terima Kasih karena telah mengikuti polling kami.

.Hasil Polling';

// jika tidak ada cookie yang bernama poll_pemain maka buat
if (!isset($_COOKIE['poll_band']))
setcookie('poll_band', '1', $masa_aktif);

// catat IP dan jenis polling ke file ip.dat
$buka_file = fopen('log/ip.dat', 'a');
$isi = $_SERVER['REMOTE_ADDR']. '#-' . $band . "\n";
fwrite($buka_file, $isi);
fclose($buka_file);
}

```

---

```

}

break;
***** */

} // akhir dari switch

?>

</body>
</html>

```

---

Simpan pada direktori ***polling*** dengan nama ***poll\_band.php***.

File yang terakhir adalah *poll\_semua.php*. Fungsi file ini adalah untuk menampilkan seluruh hasil polling dalam satu halaman sekaligus. Jadi kita nantinya kita akan membuat tiga objek dari class polling. Untuk lebih memperjelas buat file baru pada PHP DESIGNER lalu ketik kode berikut

Script dari ***poll\_semua.php***

---

```

<?php

***** *****
** Nama File.....: poll_semua.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan....: untuk menampilkan seluruh polling
***** *****

// panggil file header.php
include ('inc/header.php');

// panggil file fungsi.php
include ('inc/fungsi.php');

// dapatkan konfigurasi dari file konfig.php
include ('inc/konfig.php');

// panggil file class_polling.php
include ('inc/class_polling.php');

// koneksi ke MySQL server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username, dan password');

// mulai tampilkan semua hasil polling

// hasil polling dari pemain terfavorit
echo '<h2>Hasil Polling Pemain Terfavorit</h2>
 .<table border="0" cellpadding="4" cellspacing="4">
 .<tr bgcolor="#d7e7f3">
 . .<td>Nama Pemain<td>Grafik</td>
 . .<td>Persentase (%)</td><td>Jumlah Pemilih</td></tr>
 .</table>';

// panggil class polling untuk menggambar grafik
$poll = new polling;
$poll->get_gambar($gambar['pemain']); // letaknya di konfig.php
$poll->get_lebar_max($lebar_max); // letaknya di konfig.php
$poll->get_tabel('poll_pemain'); // pilih tabel yang digunakan
$poll->query_SQL("SELECT * FROM poll_pemain ORDER BY suara DESC");

```

```

$poll->get_jml_kandidat(); // hitung jumlah kandidat
$poll->get_suara_terbanyak(); // dapatkan suara yang terbanyak

// lakukan query ulang
$poll->query_SQL();
$total_pml = $poll->get_total_pemilih(); // hitung jumlah total pemilih

// lakukan looping untuk menggambar setiap grafik
$hasil = $poll->query_SQL();
while ($data = mysql_fetch_array($hasil))
{
 $kandidat = $data[0]; // nama masing2 kandidat
 $suara = $data[1]; // suara masing2 kandidat
 $persen = $poll->get_persen($suara); // persentase masing2 kandidat
 $grafik = $poll->gambar_grafik($suara); // grafik dari masing2 kandidat

 echo '<tr><td>' . $kandidat . '</td><td>' . $grafik . '</td>'
 . '<td>' . $persen . '%' . '</td><td>' . $suara . '</td></tr>';
}

// tutup table
echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="4"></td></tr>'
 . '</table>'
 . '<p>Total Pemilih: ' . $total_pml . ' orang.</p>'
 . '<p>Home &nbsp '
 . 'Ikuti Polling</p>';

/**/

// hasil polling klub terfavorit
echo '<h2>Hasil Polling Klub Terfavorit</h2>'
 . '<table border="0" cellpadding="4" cellspacing="4">'
 . '<tr bgcolor="#d7e7f3">'
 . '<td>Nama Klub<td>Grafik</td>'
 . '<td>Persentase (%)<td><td>Jumlah Pemilih</td>'
 . '</tr>';

// panggil class polling untuk menggambar grafik
$poll = new polling;
$poll->get_gambar($gambar['klub']); // letaknya di konfig.php
$poll->get_lebar_max($lebar_max); // letaknya di konfig.php
$poll->get_tabel('poll_klub'); // pilih tabel yang digunakan
$poll->query_SQL("SELECT * FROM poll_klub ORDER BY suara DESC");
$poll->get_jml_kandidat(); // hitung jumlah kandidat
$poll->get_suara_terbanyak(); // dapatkan suara yang terbanyak

// lakukan query ulang
$poll->query_SQL();
$total_pml = $poll->get_total_pemilih(); // hitung jumlah total pemilih

// lakukan looping untuk menggambar setiap grafik
$hasil = $poll->query_SQL();
while ($data = mysql_fetch_array($hasil))
{
 $kandidat = $data[0]; // nama masing2 kandidat
 $suara = $data[1]; // suara masing2 kandidat
 $persen = $poll->get_persen($suara); // persentase masing2 kandidat
 $grafik = $poll->gambar_grafik($suara); // grafik dari masing2 kandidat

 echo '<tr><td>' . $kandidat . '</td><td>' . $grafik . '</td>'
 . '<td>' . $persen . '%' . '</td><td>' . $suara . '</td></tr>';
}

// tutup table

```

```

echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="4"></td></tr>
. '</table>
. '<p>Total Pemilih: ' . $total_pml . ' orang.</p>'
. '<p>Home '
. 'Ikuti Polling</p>';
. ****
// hasil pollind dari band terfavorit
echo '<h2>Hasil Polling Band Terfavorit</h2>
. '<table border="0" cellpadding="4" cellspacing="4">
. '<tr bgcolor="#d7e7f3">
. '<td>Nama Band<td>Grafik<td>
. '<td>Persentase (%)<td><td>Jumlah Pemilih</td>
. '</tr>';

// panggil class polling untuk menggambar grafik
$poll = new polling;
$poll->get_gambar($gambar['band']); // letaknya di konfig.php
$poll->get_lebar_max($lebar_max); // letaknya di konfig.php
$poll->get_tabel('poll_band'); // pilih tabel yang digunakan
$poll->query_SQL("SELECT * FROM poll_band ORDER BY suara DESC");
$poll->get_jml_kandidat(); // hitung jumlah kandidat
$poll->get_suara_terbanyak(); // dapatkan suara yang terbanyak

// lakukan query ulang
$poll->query_SQL();
$total_pml = $poll->get_total_pemilih(); // hitung jumlah total pemilih

// lakukan looping untuk menggambar setiap grafik
$hasil = $poll->query_SQL();
while ($data = mysql_fetch_array($hasil))
{
 $kandidat = $data[0]; // nama masing2 kandidat
 $suara = $data[1]; // suara masing2 kandidat
 $persen = $poll->get_persen($suara); // persentase masing2 kandidat
 $grafik = $poll->gambar_grafik($suara); // grafik dari masing2 kandidat

 echo '<tr><td>' . $kandidat . '</td><td>' . $grafik . '</td>
 . '<td>' . $persen . '%<td><td>' . $suara . '</td></tr>';
}

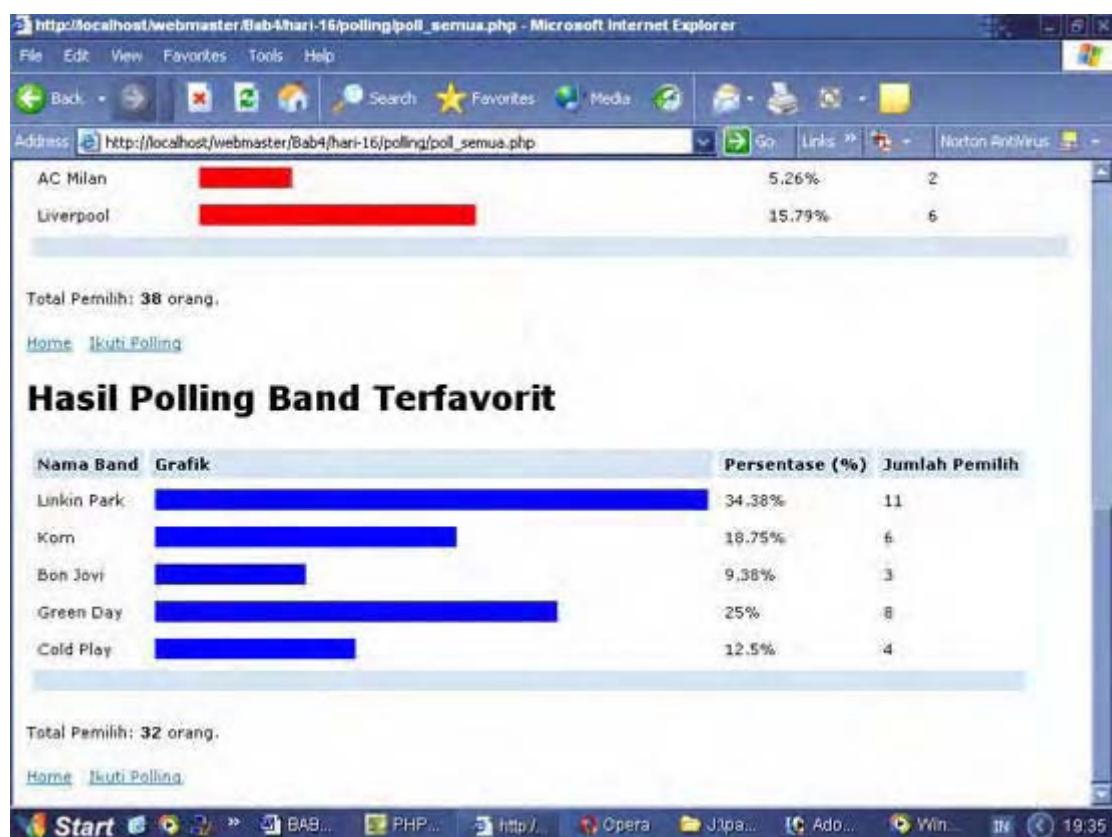
// tutup table
echo '<tr bgcolor="#d7e7f3" height="15"><td colspan="4"></td></tr>
. '</table>
. '<p>Total Pemilih: ' . $total_pml . ' orang.</p>'
. '<p>Home '
. 'Ikuti Polling</p>';

?>

</body>
</html>

```

Simpan pada direktori **polling** dengan nama **poll\_semua.php**. Tampilan file ditunjukkan oleh gambar 4.11.



Gambar 4.11: Semua hasil polling

Akhir dari file yang berada pada direktori ***polling***

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

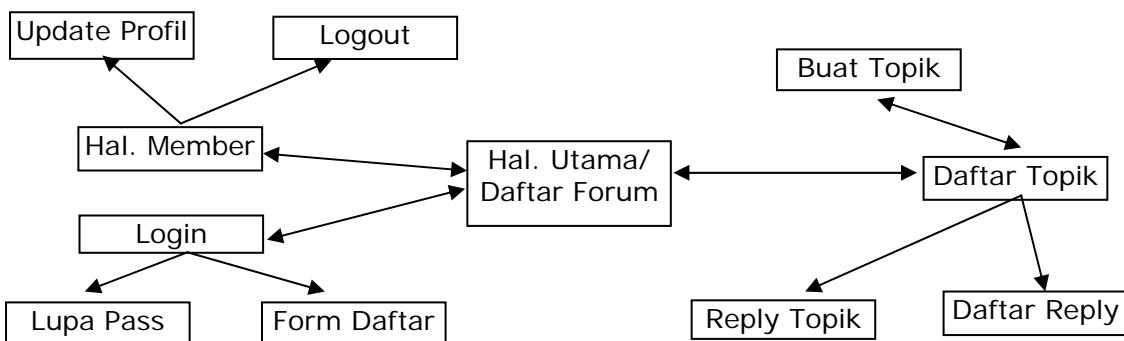
## Hari Ke-18 - 19 & 20

Seperti biasa karena kita sudah menginjak hari baru maka ikuti langkah-langkah berikut:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab4\
- Buat folder baru, namai folder tersebut hari-18-19-20

## ForumBARENG.localhost

Aplikasi ke dua yang kita buat adalah sebuah website forum. Pembuatan website kita bukan kita tekankan pada tampilan melainkan pada fungsionalnya. Untuk masalah tampilan akan dibahas pada bab khusus. Kita akan mengerjakan proyek ini dalam dua hari kedepan. Untuk lebih jelasnya lihat bagan sketsa proyek berikut.



Gambar 4.12: Sketsa halaman ForumBARENG

### Sketsa Proyek

Sketsa proyek kita bagi dalam dua bagian, yaitu bagian administrator dan bagian pengunjung. Untuk yang pertama kita bahas dulu bagian administrator.

### Bagian Administrator

- Untuk masuk ke halaman administrator / control panel admin harus memasukkan username dan password
- Admin dapat menambah, menghapus dan mengedit forum
- Admin dapat menambah dan menghapus topik pada forum
- Admin dapat menghapus reply dari user pada topik tertentu
- Admin dapat menghapus user/member
- Admin dapat mengatur berapa daftar reply, topik atau forum yang ditampilkan per halaman baik untuk control panel maupun untuk bagian user
- Admin dapat mengirim email ke semua member

### Bagian Pengunjung/Member

- User dapat menjadi member dari ForumBareng dengan cara mendaftar lewat form daftar
- Jika user belum login atau ia belum menjadi member maka user ID-nya adalah *guest*
- User dapat memilih forum yang ada pada forumBareng sesuai dengan keinginan
- User dapat membuat topik baru pada forum tertentu
- User dapat melakukan reply pada topik atau reply dari user yang lain
- User dapat melakukan reset pada passwordnya jika lupa dan password baru langsung dikirim ke alamat email user

Untuk Mengaplikasikan proyek ini ikuti beberapa langkah berikut untuk konfigurasi direktori:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab4\hari-18-19-20
- Buat folder baru namai folder tersebut **forum**
- Masuk pada direktori **forum** yang baru anda buat

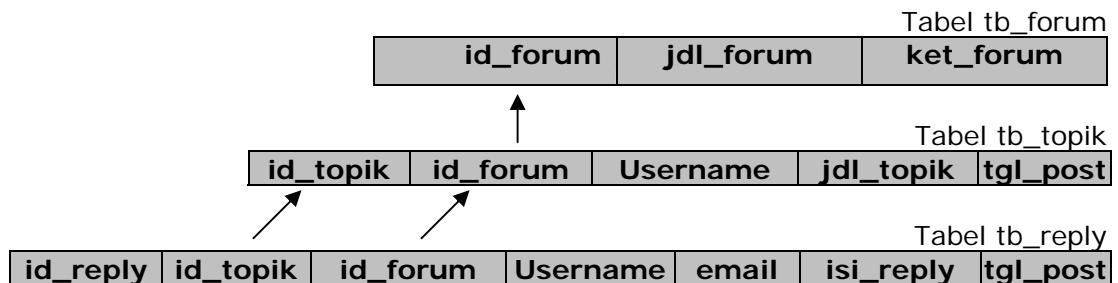
- Buat dua folder berikut **admin** dan **inc**

Daftar file yang akan kita buat untuk membangun website ForumBARENG ditunjukkan oleh tabel 4.2 berdasarkan letak direktoriannya.

**Tabel 4.2** Daftar file untuk website ForumBARENG

Direktori <b>forum</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.php	Menampilkan halaman utama dan daftar forum
topik.php	Menampilkan daftar topik dan mem-post topik
reply.php	Menampilkan daftar reply dan mem-post reply
member.php	Menampilkan form login, daftar menu, dan update profil
daftar.php	Menampilkan dan memproses form registrasi
lupa_pass.php	Menampilkan dan memproses form lupa password
db_forum.sql	Script SQL untuk membangun database forum
Direktori <b>forum\admin</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.php	Menampilkan form login atau menu administrator
forum.php	Menampilkan, mengedit, dan menghapus forum
topik.php	Menampilkan daftar topik dan menghapus topik
reply.php	Menampilkan daftar reply dan menghapus reply
member.php	Menampilkan daftar member dan menghapus member
Direktori <b>forum\inc</b>	
<b>Nama File</b>	<b>Fungsi</b>
index.php	Melakukan redirect ke halaman utama
header.php	Menyimpan tampilan header HTML
fungsi.php	Menyimpan kumpulan fungsi untuk semua halaman
class_forum.php	Berisi class halaman berguna untuk menampilkan nomor halaman
konfig.php	Menyimpan konfigurasi halaman
kata_kotor.txt	Menyimpan daftar kata-kata kotor

Berikut ini adalah sketsa hubungan(Relational) dari dari tabel tb\_forum, tb\_topik dan tb\_reply pada database forum



Gambar 4.13: Sketsa hubungan antar tabel forum, topik dan replay.

Untuk file yang pertama seperti biasa, kita akan membuat file untuk membangun databasenya terlebih dahulu. Untuk itu jalankan PHP DESIGNER 2005 klik menu **File > New > SQL** kemudian salin kode berikut ini

#### Script dari file **db\_forum.sql**

```
-- Buat database forum
-- jika sudah ada hapus

DROP DATABASE IF EXISTS forum;
CREATE DATABASE forum;
```

```
-- aktifkan database forum
USE forum;

-- buat tabel admin
CREATE TABLE admin
(
 username varchar(16) PRIMARY KEY,
 password varchar(32)
) TYPE='MyISAM';

-- buat tabel forum
CREATE TABLE tb_forum
(
 id_forum varchar(12) PRIMARY KEY,
 jdl_forum varchar(50),
 ket_forum varchar(255)
) TYPE='MyISAM';

-- buat tabel topik
-- id_topik bertipe auto_icrement artinya selalu ditambah satu
-- jika ada penambahan record baru
CREATE TABLE tb_topik
(
 id_topik int(4) PRIMARY KEY AUTO_INCREMENT,
 id_forum varchar(12),
 username varchar(16),
 jdl_topik varchar(100),
 tgl_post varchar(20)
) TYPE='MyISAM';

-- buat tabel reply
CREATE TABLE tb_reply
(
 id_reply int(4) PRIMARY KEY AUTO_INCREMENT,
 id_topik int(4),
 id_forum varchar(12),
 username varchar(16),
 email varchar(50),
 isi_reply text,
 tgl_post varchar(20)
) TYPE='MyISAM';

-- buat tabe member
CREATE TABLE member
(
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 email varchar(50),
 posting int(4),
 last_login varchar(30)
) TYPE='MyISAM';

-- buat privilege untuk database ini
GRANT SELECT, UPDATE, INSERT, DELETE ON forum.*
TO userforum@localhost IDENTIFIED BY 'forum123';

-- akhir dari file (EOF)
```

Simpan pada direktori **forum**dengan nama ***db\_forum.sql***. Untuk mengeksekusi file ini terdapat dua cara yaitu melalui MySQL Monitor atau lewat phpMyAdmin. Pada MySQL monitor ketikkan

---

mysql> source C:/Apache2/htdocs/webmaster/Bab4/hari-18-19-20/forum/db\_forum.sql

Pada phpMyAdmin, klik tombol **Browse** lalu arahkan ke alamat seperti diatas.

File-file berikutnya yang akan kita buat adalah file yang berada pada direktori **inc**. Kita membuat file-file pada direktori ini terlebih dahulu karena file-file pada direktori ini akan kita panggil hampir pada setiap halaman. Baik untuk halaman administrator ataupun halaman pengunjung.

Tanpa banyak gocek lagi langsung saja klik **File > New > PHP** pada PHP DESIGNER 2005 lalu ketik kode berikut

Script dari file ***index.php***

---

```
<?php
/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: melakukan redirect halaman
*/
// Jika user mencoba masuk ke direktori inc kembalikan ia
// ke halaman utama
header('Location: ../index.php');

?>
```

---

Simpan pada direktori **forum\inc** dengan nama ***index.php***. Tugas file ini hanyalah melakukan redirect halaman ketika ada yang mencoba masuk ke direktori inc.

Berikutnya kita akan membuat file yang menyimpan daftar kata-kata kotor. Untuk itu buka text editor kesayangan anda (notepad, wordpad, vi, gedit dan sebagainya). Ketik daftar kata berikut,

Isi dari file ***kata\_kotor.txt*** (maaf, tidak bermaksud pornografi atau kasar)

---

```
ass
asshole
shit
bullshit
fuck
fucker
keparat
bangsat
bajingan
brengsek
```

---

Simpan pada direktori **inc** dengan nama ***kata\_kotor.txt***. Anda dapat menambahkan kata-kata lain yang menurut anda perlu untuk difilter. Karena dalam pikiran saya yang terpikir hanya sepuluh kata tersebut.

Selanjutnya kita akan membuat file yang akan menampilkan header HTML pada setiap halaman. Buat file baru pada PHP DESIGNER 2005 lalu ketik kode berikut,

Script dari file ***header.php***

---

```
<?php
```

```

** Nama File.....: header.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk menampilkan HTML header
***** /

?>

<html>
<head>
<title>Forum BARENG - Mari Berbagi!</title>
<style>
 body{font-family: verdana; font-size: 11px}
 td{font-family: verdana; font-size: 11px}
 a{color: #076fd0}
 a:visited{color: #076fd0}
</style>
</head>
<body>
```

---

Simpan pada direktori **forum\inc** dengan nama **header.php**. File ini hanya bertugas untuk menampilkan header HTML pada setiap halaman. Kita nantinya memanggil file ini dengan include() atau require().

File selanjutnya adalah file yang berfungsi untuk menyimpan setting tampilan. Berapa topik atau reply yang harus ditampilkan per halaman kita simpan pada file ini. Klik **File > New > PHP** untuk membuat file PHP baru lalu ketik kode berikut,

Script dari file **konfig.php**

```
<?php

** Nama File.....: konfig.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: berisi konfigurasi untuk forum
***** /

// cegah pengaksesan langsung dari browser
if (eregi('class_forum.php', $_SERVER['PHP_SELF']))
{
 // kembalikan ke halaman utama
 header('Location: ../index.php');
 exit; // keluar dari rutin script
}

// Konfigurasi untuk administrator
***** /

// tentukan daftar topik yang ditampilkan per halaman
$a_tph = 3; // 3 topik per halaman

// tentukan daftar reply yang ditampilkan per halaman
$a_rph = 2; // 2 reply per halaman

// tentukan daftar member yang ditampilkan per halaman
$a_mph = 2; // 2 member per halaman

// tentukan letak file yang menyimpan daftar kata kotor
```

```
$daftar_kata = 'inc/kata_kotor.txt';

***** Konfigurasi untuk user *****
***** // tentukan daftar topik yang ditampilkan per halaman
$u_tph = 3; // 3 topik per halaman
***** // tentukan daftar reply yang ditampilkan per halaman
$u_rph = 2; // 2 reply per halaman
?>
```

---

Simpan pada direktori **forum\inc** dengan nama ***konfig.php***. Kita membagi konfigurasi menjadi dua bagian yaitu untuk administrator dan user. Untuk administrator nama variabelnya diawali dengan huruf a seperti `$a_tph`. Dan untuk user diawali huruf u seperti `$u_tph`. Pada file ini kita juga menentukan letak file yang menyimpan daftar kata kotor.

Selanjutnya kita akan membuat file yang berisi kumpulan fungsi yang kita pakai pada hampir setiap halaman pada website ForumBARENG. Buat file baru pada PHP DESIGNER 2005 dengan mengklik **File > New > PHP** lalu ketik kode berikut ini,

#### Script dari file ***fungsi.php***

---

```
<?php

***** ** Nama File....: fungsi.php ***
** Penulis.....: Rio Astamal ***
** Tanggal.....: 12-10-2005 ***
** Penjelasan...: Kumpulan fungsi untuk forum ***
***** /

// cegah pengaksesan langsung dari browser
if (eregi('fungsi.php', $_SERVER['PHP_SELF']))
 exit('Error: Akses ditolak.');

// fungsi untuk mengkoneksi dengan MySQL server
function konek_db()
{
 // untuk username = secure dan password = secure123
 @ $koneksi = mysql_connect('localhost', 'userforum', 'forum123');
 // jika gagal melakukan koneksi kembalikan false
 if (!$koneksi)
 return false;
 else
 {
 // pilih database yang digunakan
 mysql_select_db('forum');
 return true;
 }
}

// cek setiap field apa ada yang kosong
function cek_field($var)
{
 foreach ($var as $field)
 {
 if ($field == '' || !isset($field))
 return false;
 }
}
```

```

 return true;
 }

// cek kevalidan email
function cek_email($email)
{
 // fungsi untuk mengecek kevalidan email
 if (ereg('^[a-zA-Z0-9_\-]+@[a-zA-Z0-9\-\-]+\.[a-zA-Z0-9\-\-\.]+\$', $email))
 return true;
 else
 return false;
}

// fungsi untuk menyaring string selain alfabet, numerik dan _
function filter_str($string)
{
 $filter = ereg_replace('[^a-zA-Z0-9_]', '', $string);
 return $filter;
}

// fungsi untuk mengenkripsi string dengan metode MD5
// dan membalik urutannya
function balik_md5($string)
{
 // untuk membalik urutan string digunakan fungsi strrev()
 $chiper_text = strrev(md5($string));
 return $chiper_text;
}

// fungsi untuk mengecek session
function cek_session($nama_ses)
{
 // jika session kosong
 if (!isset($_SESSION[$nama_ses]))
 return false; // kembalikan nilai false
 else
 // jika tidak kosong
 return true; // kembalikan nilai true
}

// fungsi untuk login
function login($tabel, $username, $password)
{
 // lakukan query untuk mencocokkan data
 $hasil = mysql_query("SELECT * FROM $tabel WHERE username='$username'
 AND password='$password'");

 // cek baris yang dikembalikan
 if (mysql_num_rows($hasil) == 0)
 return false; // data tidak cocok
 else
 return true; // kembalikan nilai true
}

// fungsi untuk membuat password secara acak
// digunakan untuk mengirim password pada form lupa password
function pass_acak($panjang=8)
{
 $kar = "ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz";
 // acak karakter
 srand((double)microtime() * 1000000);

 // lakukan looping sebanyak $panjang(parameter)
}

```

```

for ($i=0; $i<$panjang; $i++) // default diulang sebanya 8x
{
 $nom_acak = rand() % 53; // untuk mendapatkan satu nomor acak
 $pass .= substr($kar, $nom_acak, 1); // ambil satu karakter
}

return $pass; // kembalikan hasil
}

// fungsi untuk mengecek kata-kata kotor
function filter_kata($string, $file)
{
 // buka file untuk daftar kata kotor
 $daftar_kata = file($file);

 // lakukan looping untuk mengganti setiap kata kotor dengan !@#$%
 foreach($daftar_kata as $kotor)
 {
 // hilangkan spasi diawal dan diakhir kata
 $kotor = trim($kotor);
 // ganti menggunakan fungsi eregi_replace()
 $string = eregi_replace($kotor, '!@#$%', $string);
 }

 return $string; // kembalikan hasil
}

// fungsi untuk logout (menghapus session)
function logout($nama_ses)
{
 // jika session kosong
 if (!isset($_SESSION[$nama_ses]))
 return false; // kembalikan false
 else
 {
 // jika tidak kosong hancurkan session tersebut
 unset($_SESSION[$nama_ses]);
 session_destroy();
 return true; // kembalikan nilai true
 }
}

?>

```

Simpan pada direktori **forum\inc** dengan nama ***fungsi.php***.

### PENJELASAN SCRIPT

Saya rasa fungsi-fungsi yang ada sudah pernah kita buat sebelumnya. Oh...ada dua yang belum pernah kita buat yaitu fungsi `pass_acak()` dan `filter_kata()`. Fungsi `pass_acak()` berguna untuk membuat karakter acak yang panjangnya dapat kita tentukan pada parameter. Namun jika tidak diisi panjang defaultnya adalah delapan karakter.

Pertama kita membuat variabel yang berisi karakter alfabets dan numerik yang kita simpan pada variabel `$kar`. Kemudian kita mengacak karakter menggunakan fungsi

```
 srand((double)microtime() * 1000000);
```

Untuk mendapatkan karakter sebanyak `$panjang` maka kita lakukan looping for sebanyak `$panjang` juga.

```
 $nom_acak = rand() % 53; // untuk mendapatkan satu nomor acak
 $pass .= substr($kar, $nom_acak, 1); // ambil satu karakter
```

Pada blok kode looping kita membuat variabel \$nom\_acak yang isinya adalah satu angka yang kita dapat secara acak. Kemudian kita mengambil satu karakter dari karakter-karakter yang ada di \$kar secara acak menggunakan substr(). Karena pada blok looping kita menuliskan \$pass dengan tanda .= maka isi dari \$pass sebelumnya akan digabung dengan isi \$pass berikutnya hingga delapan kali atau sesuai parameter.

Karakter yang dihasilkan fungsi pass\_acak() selalu berubah pada setiap fungsi itu dipanggil. Contoh karakter yang dihasilkan dari fungsi pass\_acak() tanpa mengisikan parameter adalah

XCYDn08R, 5uMwsrvj, 4bZHM3us, C60AS5hh dan sebagainya.

Fungsi filter\_kata() berfungsi untuk mengganti kata yang ada pada daftar kata kotor. Daftar kata kotor kita simpan pada file kata\_kotor.txt yang ada pada direktori inc. untuk mengganti setiap kata kotor kita menggunakan looping foreach dan fungsi eregi\_replace(). Karena eregi\_replace tidak membedakan huruf kecil atau huruf besar (tidak case sensitive).

File berikunya yang akan kita buat adalah file yang berisi class halaman. Fungsi file ini adalah untuk menampilkan link nomor halaman. Seperti first, before, last dan next. Untuk membuatnya klik **File > New > PHP** pada PHP DESIGNER 2005 kemudian ketik kode berikut,

#### Script dari file *class\_forum.php*

---

```
<?php

/*
** Nama File.....: class_forum.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Class untuk menampilkan link nomor
** halaman
***** */

// cegah pengaksesan langsung dari browser
if (eregi('class_forum.php', $_SERVER['PHP_SELF']))
{
 // kembalikan ke halaman utama
 header('Location: ../index.php');
 exit; // keluar dari rutin script
}

class halaman
{

 // deklarasikan properti untuk class
 var $tabel, $page, $file, $per_halaman, $jml_data, $jml_hal, $hasil_query;
 var $record, $last_page, $sebelumnya, $berikutnya, $tampilkan_hal;

 // metode untuk memilih tabel yang digunakan
 function set_tabel($nama_tabel)
 {
 $this->tabel = $nama_tabel;
 }

 // metode untuk menentukan data yang di-post
 function set_page($halaman)
 {
 $this->page = $halaman;
 }

 // metode untuk menentukan banyaknya pesan per halaman
}
```

```

function set_pph($angka)
{
 $this->per_halaman = $angka;

 return $this->per_halaman;
}

// metode untuk mendapatkan jumlah record yang harus ditampilkan
// pada query SQL. jumlah record = halaman x pesan per halaman
function get_record()
{
 $hasil = $this->page * $this->per_halaman;
 $this->record = $hasil;

 return $this->record;
}

// metode untuk melakukan query sql. digunakan untuk mendapatkan jumlah
// data, jumlah record dan sebagainya
function query_SQL($q = 1, $id='', $isi='', $lainnya='')
{
 // jika parameter pertama berisi 1 lakukan query berikut
 if ($q == 1)
 $query = mysql_query("SELECT * FROM $this->tabel");
 else if ($q == 2) // jika 2 lakukan query berikut
 $query = mysql_query("SELECT * FROM $this->tabel WHERE $id='$isi'");
 else if ($q == 3) // jika 3 lakukan query berikut
 $query = mysql_query("SELECT * FROM $this->tabel WHERE $id='$isi' ORDER BY
$lainnya DESC LIMIT $this->record, $this->per_halaman");
 else
 $query = mysql_query($q); // jika bukan 1,2 atau 3 lakukan query sesuai
parameter

 $this->hasil_query = $query;

 return $this->hasil_query; // kembalikan hasil dari query
}

// metode untuk mendapatkan jumlah data pada database
function get_jml_data()
{
 $jumlah = mysql_num_rows($this->hasil_query);
 $this->jml_data = $jumlah;

 return $this->jml_data;
}

// metode untuk mendapatkan jumlah halaman
function get_jml_hal()
{
 // untuk menghitung jumlah halaman digunakan fungsi ceil
 // dimana >> Jumlah data : pesan per halaman
 // jika hasilnya koma, maka dibulatkan ke atas
 $jumlah = ceil($this->jml_data / $this->per_halaman);

 $this->jml_hal = $jumlah;

 return $this->jml_hal;
}

// metode untuk mendapatkan halaman
// paling awal, sebelumnya, berikutnya dan paling akhir
function set_hal()
{

```

```

// halaman terakhir kita kurangi satu karena pada nomor
// link halaman kita menambahkan satu. Lihat pada bagian
// intval($i + 1)
$this->last_page = $this->jml_hal - 1;

// link halaman sebelumnya didapat dengan mengurangi nilai
// halaman sekarang(yang aktif) dengan satu
$this->sebelumnya = $this->page - 1;

// link halaman berikutnya didapat dengan menambahkan nilai
// halaman sekarang dengan satu
$this->berikutnya = $this->page + 1;
}

// metode untuk menampilkan nomor halaman
function show_page($URL)
{
 // jika jumlah halaman lebih dari satu tampilkan selain itu jangan
 if ($this->jml_hal > 1)
 {
 // jika halaman sekarang 0 (paling awal) jangan tampilkan link
 // first dan before ganti dengan tulisan biasa(warna abu-abu)
 if ($this->page == 0)
 {
 $first = "<< First";
 $back = "< Before";
 }
 else
 {
 $first = "<< First";
 $back = "sebelumnya'>< Before";
 }

 // jika halaman sekarang sama dengan nilai halaman terakhir
 // jangan tampilkan link last dan next
 if ($this->page == $this->last_page)
 {
 $last = "Last >>";
 $next = "Next >>";
 }
 else
 {
 $last = "last_page'>Last >>";
 $next = "berikutnya'>Next >>";
 }

 // tampilkan
 $halaman = "$first $back \n";

 // gunakan looping untuk menampilkan setiap nomor halaman
 for ($i=0; $i<$this->jml_hal; $i++)
 {
 // jika nomor halaman sama dengan halaman yang sedang
 // dibuka perbesar angka tersebut
 if ($i == $this->page)
 $halaman .= " " .intval($i + 1). " \n";
 else
 $halaman .= " ".intval($i + 1). "\n";
 }

 $halaman .= " $next $last\n";
 }
}

```

```

 // kurang lebih seperti berikut, dimana x adalah nomor halaman
 // <> First < Before x x x Next > Last >>
}
else
 $halaman = ''; // artinya halaman yang ada hanya satu

$this->tampilkan_hal = $halaman;

// kembalikan nilai karena akan dicetak ke layar
return $this->tampilkan_hal;
}

} // akhir dari class halaman

?>

```

---

Simpan pada direktori **forum\inc** dengan nama ***class\_forum.php***.

### PENJELASAN SCRIPT

Sebenarnya pada bab 3 kita pernah membuat nomor link halaman untuk bukutamu dan berita. Namun pada bab tiga kita membuatnya dengan cara tradisional. Sekarang kita coba membuatnya namun menggunakan metode class. Sehingga kode kita akan lebih efisien.

Untuk menampilkan link nomor halaman ada beberapa nilai yang harus ditentukan yaitu, data halaman yang dihandel, pesan per halaman, jumlah record yang harus ditampilkan, jumlah baris/data dan jumlah halaman.

Jumlah record berhubungan dengan jumlah data yang harus ditampilkan pada halaman. Nilai dari record kita dapat dengan mengalikan yang halaman yang aktif dengan banyaknya pesan per halaman.

Record = halaman x pesan per halaman

Pada metode query\_SQL() kita membuat empat parameter. Satu wajib diisi dan tiga lainnya bersifat opsional. Query 1, 2, 3 pada metode query\_SQL kita sesuaikan dengan kebutuhan kita. Query 1 dan dua biasanya kita gunakan untuk mendapatkan jumlah baris dan jumlah halaman. Sedangkan untuk query 3 kita untuk menampilkan data pada database.

Khusus query 3 terdapat klausa LIMIT yang berguna untuk menampilkan sejumlah data per halaman. Dengan mengetahui jumlah record kita dapat memberitahukan MySQL harus dimulai dari baris keberapa data yang harus ditampilkan.

Misalnya sekarang halaman 2 dan jumlah data per halaman yang ditetapkan adalah 3. maka nilai recordnya adalah  $2 * 3 = 6$ . Jadi pada query SQL akan seperti berikut

```
SELECT * FROM nama_tb WHERE id='xxx' ORDER BY id_xxx DESC LIMIT 6, 3
```

Jadi data yang ditampilkan akan dimulai dari record/baris ke-6 tampilkan sebanyak 3 baris.

Untuk metode get\_jml\_data() kita hanya menghitung jumlah baris yang dikembalikan oleh metode query\_SQL(). Sedangkan untuk metode get\_jml\_hal() kita menghitung jumlah halaman dengan cara membagi jumlah data/baris dengan pesan per halaman.

Jumlah halaman = jumlah baris : pesan per halaman

Untuk menghitung kita gunakan fungsi ceil() agar hasilnya dibulatkan ke atas. Misalnya jumlah baris 5 dan pesan per halaman 2. Maka jumlah halamannya

```
$jml_hal = ceil(5/2);
// hasilnya 2.5 dibulatkan menjadi 3. Halaman 0, 1, 2
```

Metode berikutnya adalah `set_hal()`. Fungsi metode ini adalah untuk menentukan isi dari link halaman Before, Next, dan Last. Untuk first nilainya tidak perlu kita set karena sudah pasti NOL karena merupakan halaman pertama. Nilai before kita peroleh dengan mengurangi nilai halaman sekarang dengan 1. Sebaliknya untuk nilai next dengan menambahkan nilai halaman sekarang dengan 1. Untuk halaman terakhir nilainya kita tentukan dengan mengurangi jumlah halaman dengan 1.

Sebagaimana kita tahu jika jumlah halamannya dua maka halamannya adalah 0 dan 1. Sehingga halaman terakhir adalah 1. Tanpa mengurangi satu maka nilai last page adalah dua. Sedangkan halaman dua tidak ada.

Metode yang terakhir adalah `show_page()`. Metode ini memiliki satu parameter yaitu URL halaman. Sebelum menampilkan halaman ada satu kondisi yang harus dipenuhi yaitu jumlah halaman harus lebih dari satu. Jika jumlah halaman hanya satu maka link nomor halaman tidak akan ditampilkan.

Jika kondisi terpenuhi artinya jumlah halaman lebih dari satu. Cek kembali beberapa kondisi yang mungkin terjadi. Yang pertama jika halaman sekarang sama dengan NOL maka jangan tampilkan link << First dan < Before.

```
if ($this->page == 0)
```

Begin juga jika halaman sekarang sama dengan nilai dari `$this->last_page` (halaman terakhir). Maka jangan tampilkan link Next > dan Last >>.

```
if ($this->page == $this->last_page)
```

Selanjutnya untuk menampilkan nomor link halaman kita gunakan looping for yang diulang sebanyak jumlah halaman.

```
for ($i=0; $i<$this->jml_hal; $i++)
```

Yang menarik juga adalah kita memberi tanda pada user halaman mana yang sedang ia buka dengan cara memperbesar ukuran hurufnya. Untuk itu kita gunakan kondisi sebagai berikut

```
if ($i == $this->page)
```

Artinya jika nomor halaman sama dengan halaman sekarang maka perbesar ukuran huruf tersebut. Selain itu tetap cetak nomor halaman dengan ukuran normal. Langkah terakhir adalah dengan membuat nilai dari properti `$this->tampilkan_hal` yaitu sama dengan `$halaman`. Karena akan dicetak ke layar kembalikan hasil dengan cara

```
return $this->tampilkan_hal;
```

Akhir dari file-file yang berada pada direktori **forum\inc**.

---

File-file berikutnya yang akan kita buat adalah file-file yang ada pada direktori admin. File-file yang ada pada direktori ini kita gunakan untuk memanage website forum. Baik menambah, mengedit atau menghapus forum. Melihat daftar topik atau reply dan menghapusnya. Memanage member yang telah bergabung dengan ForumBARENG.

File pertama yang akan kita buat adalah file untuk login dan menampilkan daftar menu untuk halaman control panel. Klik **File > New > PHP** untuk membuat file baru kemudian ketik kode berikut ini

---

Script dari file ***index.php***

---

```
<?php
```

```

*****+
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk login admin dan menu utama
*****+/

// panggil session_start() karena berhubungan dengan session
session_start();

// panggil file header.php
include('../inc/header.php');

// panggil file fungsi.php
include('../inc/fungsi.php');

// koneksi ke MySQL Server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'login';

// filter karakter pada proses
$proses = filter_str($proses);

// sebagai alternatif kita gunakan \n agar source kode HTML-nya sesuai
// dengan struktur kode PHP kita

// handel setiap proses dengan switch dan case
switch ($proses)
{
 *****/
 case 'login':
 /***/

 // jika session admin kosong maka admin belum login
 // jadi tampilkan form login selain itu tampilkan menu utama
 if (!cek_session('admin'))
 {

 // sebagai alternatif kita gunakan \n agar source HTML-nya
 // sesuai dengan struktur kode. dengan catatan untuk mencetak
 // string kita menggunakan petik dua
 echo "<h2>ForumBARENG Admin Area - Please login</h2>\n"
 . "<form action='index.php?proses=proses_login' method='post'>\n"
 . "Username: <input type='text' name='username'>

\n"
 . "Password: <input type='password' name='password'>

\n"
 . "<input type='submit' value='LOGIN'>\n"
 . "</form>\n";
 }
 else
 {
 echo "<h2>Selamat Datang di Control Panel ForumBARENG</h2>\n"
 . "<p>Silahkan pilih link dibawah ini untuk memanage forum</p>\n"
 . "<p>Manaje Forum</p>\n"
 . "<p>Manaje Topik</p>\n"
 . "<p>Manaje Reply</p>\n"
 . "<p>Manaje Member</p>\n"
 . "<p>Logout</p>\n";
 }
}

```

```

} // akhir dari else

break;
/************/

/************/
case 'proses_login':
/************/

// ambil data yang dipost sekaligus filter karakter
$username = filter_str($_POST['username']);
$password = filter_str($_POST['password']);

// enkripsi password
$password = balik_md5($password);

// gunakan fungsi login() untuk mengecek kecocokan data
if (!login('admin', $username, $password))
{
 // jika gagal jangan buatkan session
 echo 'Username atau password salah. Kembali';
}
else
{
 // buatkan session karena berhasil login
 $_SESSION['admin'] = $username;
 echo 'Login berhasil. Klik disini untuk ' .
 '.masuk ke admin area';
}

break;
/************/

/************/
case 'logout':
/************/

if (!logout('admin'))
 echo 'Tidak bisa logout. Login dulu.';
else
 echo 'Anda telah logout dari sistem. Klik ' .
 '.disini untuk login kembali.';

break;
/************/

/************/
case '__add_admin_to_database__':
/************/

/*
case ini berfungsi untuk memasukkan account administrator ke database
ini dikarenakan fungsi yang kita gunakan login adalah balik_md5()
dan untuk menghasilkan string chiper ini hanya bisa dilakukan lewat
script PHP bukan pada MySQL

untuk memanggil fungsi ini harus diketikkan langsung pada address bar
index.php?proses=__add_admin_to_database__ lalu ENTER

untuk mencegah eksloitasi sistem, kita tidak menyediakan form untuk
menambahkan account admin ke database melainkan langsung
melakukan query
*/

```

```

// tentukan username dan password yang diinginkan
$username = 'admin';
$password = balik_md5('__pas123__');

// lakukan query INSERT untuk memasukkan account ke database
$hasil = mysql_query("INSERT INTO admin VALUES('{$username}', '{$password}')");
if (!$hasil)
 echo 'Error: Gagal memasukkan ke database. Mungkin account

 sudah dimasukkan.';
else
 echo 'Account untuk administrator berhasil dimasukkan ke database.
' .
 'Login';

break;

} // akhir dari switch

mysql_close(); // tutup koneksi ke MySQL

?>

</body>
</html>

```

---

Simpan pada direktori **forum\admin** dengan nama **index.php**. Output dari menu utama admin area ditunjukkan oleh gambar 4.12

### PENJELASAN SCRIPT

Struktur file ini sama dengan yang biasa kita gunakan untuk halaman administrator pada proyek-proyek sebelumnya. Pada halaman utama jika session admin belum terisi maka ia(admin) belum login. Untuk itu tampilkan form login sebaliknya jika sudah terisi tampilkan menu untuk manajemen forum.

Karena pada login kita menggunakan fungsi `balik_md5()` maka untuk memasukkan account untuk administrator harus dilakukan dari script PHP. Mengingat string chiper hasil fungsi `balik_md5()` tidak bisa dihasilkan lewat MySQL. Untuk memasukkan account administrator ke database kita dapat mengetikkan langsung pada URL

[http://..../Bab4/hari-18-19-20/admin/index.php?proses=\\_\\_add\\_admin\\_to\\_database\\_\\_](http://..../Bab4/hari-18-19-20/admin/index.php?proses=__add_admin_to_database__)

Jika isi dari `$username` dari case ini tidak dirubah case ini hanya bisa dieksekusi satu kali saja. Ini dikarenakan tipe kolom `username` primary key artinya tidak boleh ada data yang sama. Setelah mengeksekusi URL diatas anda dapat login ke admin area menggunakan `username admin` dan `password __pas123__`.



Gambar 4.12: Halaman utama admin area.

File berikutnya yang akan kita buat adalah file yang berfungsi untuk menambah, mengedit, dan menghapus forum. Untuk membuatnya klik **File > New > PHP** pada PHP DESIGNER 2005 lalu ketik kode berikut,

#### Script dari file **forum.php**

---

```
<?php

/*
** Nama File....: forum.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan...: halaman untuk memanage forum
***** */

// panggil session_start() karena berhubungan dengan session
session_start();

// panggil file header.php
include('../inc/header.php');

// panggil file fungsi.php
include('../inc/fungsi.php');

/// panggil file class_forum.php
include('../inc/class_forum.php');

// koneksi ke MySQL Server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

// filter isi dari proses dan page
$proses = filter_str($proses);

// cek isi session admin, untuk mengetahui sudah login atau belum
if (!cek_session('admin'))
 echo "<p>Anda belum login. Login dulu.</p>\n";

```

```

else
{
 // handel setiap proses dengan switch dan case
 switch ($proses)
 {
 /*****
 case 'view':
 *****/
 echo "<h2>Daftar Forum yang ada pada ForumBARENG</h2>\n";
 // lakukan query
 $hasil = mysql_query("SELECT * FROM tb_forum");
 // cek jumlah data/baris yang dikembalikan
 if (mysql_num_rows($hasil) == 0)
 exit("<p>Masih belum ada forum yang dibuat.</p>\n"
 " Buat Forum\n");
 // buat header tabel
 echo "<table border='0' cellpadding='4'>\n"
 . " <tr bgcolor='#d7e7f3'>\n"
 . " <td>Judul Forum</td><td>Edit?</td>\n"
 . " <td>Topik</td>\n"
 . " <td>Hapus?</td>\n"
 . " </tr>\n";
 // tampilkan hasil dengan looping
 while ($data = mysql_fetch_array($hasil))
 {
 echo "<tr>\n"
 . "<td>".$data['jdl_forum']."</td>\n"
 . "<td>EDIT</td>\n"
 . "<td>TOPIK</td>\n"
 . "<td>HAPUS</td>\n"
 . "</tr>\n";
 } // akhir dari while
 // tutup tabel
 echo "<tr bgcolor='#d7e7f3' height='15'> \n"
 . "<td colspan='4'></td></tr>\n"
 . "</table>\n\n"
 . "<p>Home &nbsp "
 . "Tambah Forum</p>\n";
 break;
 *****/
}

// ambil data dari URL
$id_forum = $_GET['forum'];

// lakukan query untuk mendapat forum yang diedit
$hasil = mysql_query("SELECT * FROM tb_forum WHERE id_forum='".$id_forum"'");
$data = mysql_fetch_array($hasil);

```

```

// buat form
echo "<h2>Form Edit Forum</h2>\n\n"
. "<form action='forum.php?proses=proses_edit' method='post'>\n"
. "Judul Forum:
\n"
. "<input type='text' name='jdl_forum' "
. "value='".$data['jdl_forum']."' size='30'>

\n"
. "Keterangan Forum:
\n"
. "<input type='text' name='ket_forum' "
. "value='".$data['ket_forum']."' size='75'>

\n"
. "<input type='hidden' name='id_forum' value='".$id_forum'>\n"
. "<input type='submit' value='E D I T'>\n"
. "</form>\n\n"
. "<p>Home &nbsp "
. "Daftar Forum</p>\n";

break;
***** */

***** */
case 'proses_edit':
***** */

// ambil data yang di-post
$id = $_POST['id_forum'];
$judul = $_POST['jdl_forum'];
$ket = $_POST['ket_forum'];

// masukkan perubahan ke database
$hasil = mysql_query("UPDATE tb_forum SET jdl_forum='".$judul', ket_forum=
'$ket' WHERE id_forum='".$id"'");
// cek berhasil atau tidak
if (!$hasil)
 echo "Error: Gagal mengupdate database. Kembali\n";
else
 echo "Forum berhasil diupdate. Forum\n";

break;
***** */

***** */
case 'tambah':
***** */

// buat form untuk menambah forum
echo "<h2>Tambah Forum</h2>\n"
. "<form action='forum.php?proses=proses_tambah' method='post'>\n"
. "ID Forum:
<input type='text' name='id'>

\n"
. "Judul Forum:
<input type='text' name='judul' size='30'>

\n"
. "Keterangan:
<input type='text' name='ket' size='100'>

\n"
. "<input type='submit' value='T A M B A H'>\n"
. "</form>\n\n"
. "<p>Home &nbsp "
. "Daftar Forum</p>\n";

break;
***** */

***** */
case 'proses_tambah':
***** */

// ambil data yang di-post
$id = $_POST['id'];

```

```

$judul = $_POST['judul'];
$ket = $_POST['ket'];

// cek jika masih ada field yang kosong
if (!cek_field($_POST))
 exit("Error: Masih ada field yang kosong. "
 ."Kembali");

// masukkan data ke database
$hasil = mysql_query("INSERT INTO tb_forum VALUES ('$id', '$judul', '$ket')");
if (!$hasil)
 echo "Error: Gagal memasukkan data ke database.\n";
else
 echo "Data berhasil dimasukkan ke database. "
 ."Forum";

break;
***** */

***** */
case 'hapus':
***** */

// ambil data dari URL
$id_forum = $_GET['forum'];

// lakukan query untuk menghapus data
$hasil = mysql_query("DELETE FROM tb_forum WHERE id_forum='$id_forum'");
if (!$hasil)
 echo "Error: Gagal menghapus data dari database.";
else
 echo "Data berhasil dihapus dari database. "
 ."Forum\n";

break;
***** */

} // akhir dari switch
} //akhir dari else

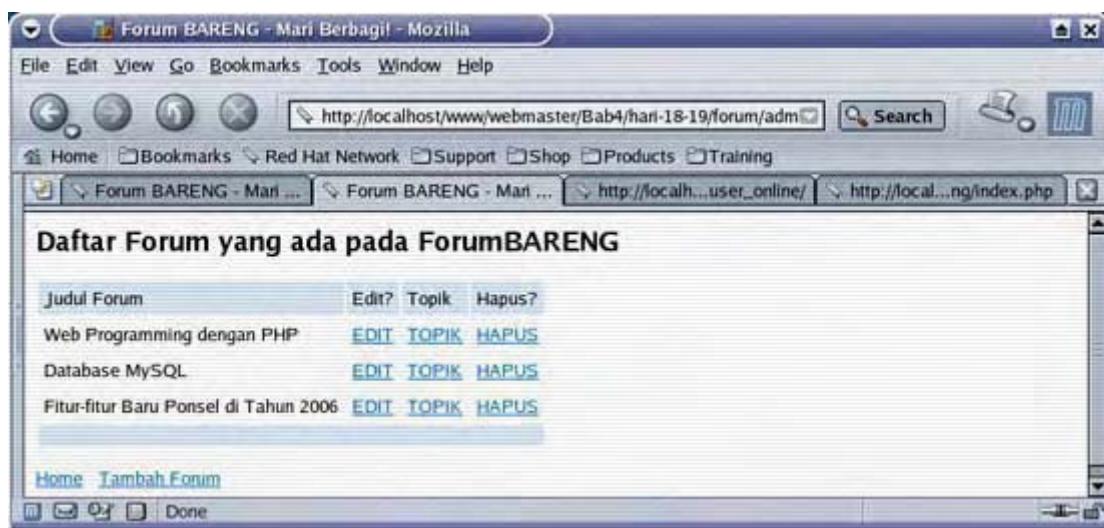
mysql_close(); // tutup koneksi ke MySQL

?>

</body>
</html>

```

Simpan pada direktori **forum\admin** dengan nama **forum.php**. Coba buat forum baru misalnya dengan id forum PHP, judul forum Web programming dengan PHP. Keterangan forum Pada forum ini anda dapat berdiskusi bla bla bla ... Output dari file ini akan terlihat seperti gambar 4.13.



Gambar 4.13: Tampilan halaman forum pada control panel

### PENJELASAN SCRIPT

Pada halaman forum.php terdapat beberapa case yaitu case *view* untuk menampilkan daftar forum. Case *edit* untuk mengedit forum yang ada. Case *proses\_edit* untuk memproses data yang dikirim dari form edit. Case *tambah* untuk menambah topik dan case *proses\_tambah* untuk memproses data dari case tambah. Dan case yang terakhir adalah case *hapus* yang berguna untuk menghapus forum.

Saya rasa tanpa penjelasan anda sudah tahu maksud dari kode masing-masing case. Namun saya coba menjelaskannya secara singkat. Pada case *view* kita mendaftar semua forum yang ada menggunakan query SQL seperti berikut

```
$hasil = mysql_query("SELECT * FROM tb_forum");
```

Untuk menampilkannya kita gunakan looping while seperti biasa yaitu

```
while ($data = mysql_fetch_array($hasil))
```

Pada setiap link baik EDIT, TOPIK, dan HAPUS kita sertakan id dari forum-forum tersebut agar saat mengedit, melihat, topik atau menghapus yang dituju forum yang kita klik.

Pada proses *edit* kita mengambil id dari forum yang akan edit menggunakan metode `$_POST`. Selanjutnya kita melakukan query untuk mendapatkan informasi judul dan keterangan forum yang bersangkutan. Kita membuat tiga type input yaitu dua berupa text dan satu berupa hidden. ID forum yang akan kita edit kita masukkan pada input type hidden.

Pada *proses\_edit* kita mengambil data yang di-post dari form edit menggunakan metode post. Kita nantinya melakukan update terhadap forum menggunakan query SQL UPDATE. Dimana forum yang diupdate adalah forum yang ber-id berdasarkan id forum yang dipost.

Case selanjutnya yaitu *tambah* pada proses kita hanya menyediakan form untuk menambah forum. Form terdiri dari tiga input text yaitu textbox untuk id forum, judul forum dan keterangan forum. Case *proses\_tambah* berfungsi untuk memasukkan data yang di-post dari case *tambah* ke database.

Case yang terakhir adalah *hapus*. Case ini berfungsi menghapus forum yang ber-id sesuai dengan forum yang diklik admin.

File berikutnya yang kita buat adalah file yang berfungsi untuk menampilkan daftar topik dan menghapus topik. Pada halaman ini nantinya kita menerapkan class halaman yang pernah kita buat. Untuk membuatnya klik **File > New > PHP** pada PHP DESIGNER 2005, kemudian ketik kode berikut ini

---

## Script dari file ***topik.php***

---

```
<?php

/*
** Nama File.....: topik.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk memanage topik

// panggil session_start() karena berhubungan dengan session
session_start();

// panggil file header.php
include('../inc/header.php');

// panggil file fungsi.php
include('../inc/fungsi.php');

// panggil file class_forum.php
include('../inc/class_forum.php');

// panggil file konfig.php
include('../inc/konfig.php');

// koneksi ke MySQL Server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

// filter isi dari proses
$proses = filter_str($proses);

// cek isi session admin, untuk mengetahui sudah login atau belum
if (!cek_session('admin'))
 echo "<p>Anda belum login. Login
 . "dulu.</p>\n";
else
{

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*
 case 'view':
 */

echo "<h2>Daftar Topik untuk masing-masing Forum</h2>\n";

// lakukan query untuk mendaftar forum
$hasil = mysql_query("SELECT * FROM tb_forum");

// cek jumlah yang dikembalikan
if (mysql_num_rows($hasil) == 0)
 exit("Masih belum ada forum.

\n
 Buat Forum\n");
}
}

```

```

// buat header tabel
echo "<table border='0' cellpadding='4'>\n"
 . " <tr bgcolor='#d7e7f3'>\n"
 . " <td>Forum</td><td>Jumlah Topik</td>"
 . " </tr>";

// tampilkan data dengan looping
while ($data = mysql_fetch_array($hasil))
{
 // buat variabel
 $id = $data['id_forum'];
 $judul = $data['jdl_forum'];

 // buat link untuk setiap forum
 $link = "Lihat";

 // lakukan query untuk menghitung jumlah topik
 $query = mysql_query("SELECT * FROM tb_topik WHERE id_forum='$id'");
 $jml_topik = mysql_num_rows($query);

 echo "<tr>\n"
 . "<td>$judul</td><td>$jml_topik [$link]</td>\n"
 . "</tr>\n";
}

// tutup table
echo "<tr bgcolor='#d7e7f3' height='15'><td colspan='2'></td></tr>\n"
 . "</table>\n"
 . "<p>Home &nbsp "
 . "Tambah Topik\n";

break;
*****/

*****/
case 'tambah':
*****/

// buat form
echo "<h2>Tambah Topik</h2>\n"
 . "<form action='topik.php?proses=proses_tambah' method='post'>\n"

// buat pilihan Forum dengan type select
 . "Pilih Forum: <select name='forum'>\n";

// lakukan looping untuk mendapatkan seluruh pilihan forum
$hasil = mysql_query("SELECT * FROM tb_forum");
while ($data = mysql_fetch_array($hasil))
{
 $id = $data[0]; // sama dengan $data['id_forum']
 echo "<option value='id'id</option>\n";
}

echo "</select>

\n" // tutup select
 . "Username:
<input type='text' name='username'>

\n"
 . "Judul Topik:
<input type='text' name='judul' size='50'>

\n"
 . "<input type='submit' value='P O S T'>\n"
 . "</form>\n\n"
 . "<p>Home &nbsp "
 . "Topik</p>\n";

break;
*****/

```

```

/*************
case 'proses_tambah':
/************

// ambil data yang di-post
$forum = $_POST['forum'];
$user = $_POST['username'];
$judul = $_POST['judul'];
// buat tanggal
$tgl = date('d-m-Y, H:i');

// cek apakah masih ada field yang kosong
if (!cek_field($_POST))
 exit("Error: Masih ada field yang kosong.

\n
 Kembali\n");

// jika tidak ada kesalahan masukkan ke database
$hasil = mysql_query("INSERT INTO tb_topik (id_forum, username, jdl_topik,
 tgl_post) VALUES ('$forum', '$user', '$judul', '$tgl')");

// cek status
if (!$hasil)
 echo "Error: Gagal memasukkan data ke database.
\n";
else
 echo "Data berhasil dimasukkan ke database.

\n"
 ."Topik\n";

break;
/************

/*************
case 'topik':
/************

// ambil data dari URL
$forum = filter_str($_GET['forum']);
$page = filter_str($_GET['page']);

if ($page == '')
 $page = 0;

echo "<h2>Daftar Topik</h2>\n";

// panggil class halaman untuk menampilkan link halaman
$hal = new halaman; // buat objek halaman
$hal->set_tabel('tb_topik'); // tentukan tabel yang digunakan
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph($a_tph); // tampilan per halaman (konfig.php)

$hal->query_SQL(2, 'id_forum', $forum); // lakukan query SQL
$jml_data = $hal->get_jml_data(); // dapatkan jumlah data/baris
$hal->get_jml_hal(); // dapatkan jumlah halaman

// cek jumlah data/baris yang dikembalikan
if ($jml_data == 0)
 exit("Tidak ada topik untuk forum ini. Topik");

// buat form header tabel
echo "<form action='topik.php?proses=hapus' method='post'>\n"
 . "<table border='0' cellpadding='4'>\n"
 . " <tr bgcolor='#d7e7f3'>\n"
 . " <td>Posted By</td><td>Topik</td>\n"

```

```

. " <td>Posted On</td><td>Reply</td>\n"
. " <td align='center'>Hapus?</td>\n"
. " </tr>\n";

$record = $hal->get_record(); // dapatkan jumlah record yang harus ditampilkan
// query ulang untuk mengurutkan data
$hasil = $hal->query_SQL(3, 'id_forum', $forum, 'id_topik');

// lakukan looping untuk menambilkan semua entry
while ($data = mysql_fetch_array($hasil))
{
 //buat variabel
 $id = $data['id_topik'];
 $user = $data['username'];
 $judul = $data['jdl_topik'];
 $tgl = $data['tgl_post'];
 $checkbox = "<input type='checkbox' name='id[]' value='$id'>\n";
 $reply = "Reply\n";

 echo "<tr>\n"
 . " <td>$user</td><td>$judul</td><td>$tgl</td>"
 . " <td>$reply</td><td align='center'>$checkbox</td>\n</tr>\n";
}

echo "<tr bgcolor='#d7e7f3' height='15'>\n <td colspan=4></td>\n"
 . " <td><input type='submit' value='HAPUS'></td></tr>\n"
 . "</table>\n"
 . "</form>\n\n";

$hal->set_hal(); // untuk memberi nilai Last, before dan next
// tampilkan link nomor halaman
echo $hal->show_page("topik.php?proses=topik&forum=$forum")
. "
\n"
. "<p>Home &nbsp "
. "Topik</p>\n";

break;
***** */

***** */
case 'hapus':
***** */

// ambil data yang di-post (berupa array)
$id = $_POST['id'];

// cek jumlah id jika 0 maka tidak ada item yang dicek
if (count($id) == 0)
 exit("Error: Anda belum memilih item satu pun.

\n"
 . "Kembali\n");

// lakukan looping untuk menghapus setiap data
foreach ($id as $sampah)
{
 // gunakan query DELETE untuk menghapus
 $hasil = mysql_query("DELETE FROM tb_topik WHERE id_topik='$sampah'");

 // tampilkan pesan berhasil atau tidaknya data dihapus
 if (!$hasil)
 echo "Error: Gagal menghapus topik dengan id: $sampah.
\n";
 else
 echo "Topik dengan id: $sampah berhasil dihapus.
\n";
}

```

```

echo "<p>Topik</p>\n";

break;
***** //

} // akhir dari switch

} // akhir dari else

mysql_close(); // tutup koneksi ke MySQL

?>

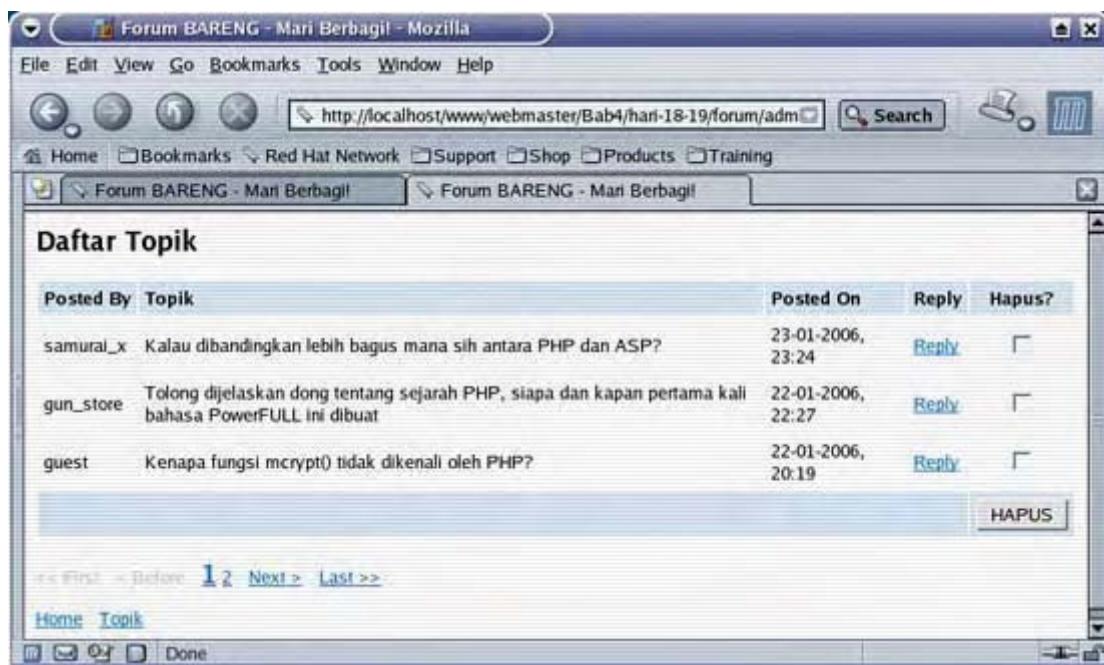
</body>
</html>

```

Simpan pada direktori **forum\admin** dengan nama ***topik.php***. Coba tambahkan topik baru pada halaman ini dengan cara mengklik link [Tambah Topik](#). Output dari file forum.php ditunjukkan oleh gambar 4.14

### PENJELASAN SCRIPT

Jika kita mengklik link [Manaje Topik](#) maka secara otomatis yang kita tuju adalah proses view. Pada case ini kita mendaftar forum yang ada dan menampilkan berapa banyak topik yang ada pada masing-masing forum. Kita juga menyediakan link untuk menuju daftar topik disebelah kanan dari nama forum.



Gambar 4.14: Daftar topic pada halaman control panel

Jika belum ada topik kita bisa membuat topik baru dengan cara mengklik link [Tambah Topik](#). Untuk urusan ini blok kode pada case tambah yang dijalankan. Untuk membuat topik tentu kita harus memilih forumnya. Untuk itu daftar forum kita peroleh dengan melakukan query SQL berikut

```

$hasil = mysql_query("SELECT * FROM tb_forum");
while ($data = mysql_fetch_array($hasil))
{
 $id = $data[0]; // sama dengan $data['id_forum']
 echo "<option value='$id'>$id</option>\n";
}

```

Selanjutnya kita lakukan looping while untuk membuat pilihan menggunakan type input select. Tipe input ini akan memberikan tampilan berupa combobox. Selanjutnya form pada case ini akan diproses oleh case *proses\_tambah*. Blok kode pada *proses\_tambah* menampung data yang dipost oleh form tambah topik untuk dimasukkan ke database.

Case selanjutnya adalah case *topik*. Saya rasa case ini yang paling menarik karena kita menerapkan objek dari class halaman untuk menampilkan link nomor halaman. Fungsi dari case ini adalah untuk menampilkan daftar topik. Diantaranya siapa yang mem-post topik, kapan di-post juga kita menghapus topik dari halaman ini.

Untuk menampilkan daftar topik kita tidak menggunakan query seperti biasa tetapi menggunakan metode query yang kita panggil dari class halaman. Ini dikarenakan jika kita menggunakan query biasa maka metode-metode lainnya seperti jumlah halaman dan sebagainya tidak mempunyai nilai karena nilai dari metode tersebut tergantung dari hasil metode *query\_SQL()*.

```
// panggil class halaman untuk menampilkan link halaman
$hal = new halaman; // buat objek halaman
$hal->set_tabel('tb_topik'); // tentukan tabel yang digunakan
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph($a_tph); // tampilan per halaman (konfig.php)

$hal->query_SQL(2, 'id_forum', $forum); // lakukan query SQL
$jml_data = $hal->get_jml_data(); // dapatkan jumlah data (topik)
$hal->get_jml_hal(); // dapatkan jumlah halaman
```

Untuk dapat menampilkan daftar topik beserta nomor link halaman. Hal pertama yang kita lakukan adalah membuat objek halaman dalam hal ini adalah *\$hal*. Selanjutnya kita menentukan tabel yang kita gunakan menggunakan metode *set\_tabel()*. Karena akan menampilkan topik maka tabel yang kita gunakan adalah *tb\_topik*.

Selanjutnya kita memanggil metode *set\_page()* untuk membentuk variabel halaman yang akan kita handel. Kemudian kita menentukan berapa topik yang harus ditampilkan per halaman menggunakan metode *set\_pph(\$a\_tph)*. Nilai dari *\$a\_tph* ada di *konfig.php* anda dapat merubahnya sesuai keinginan.

Langkah berikutnya adalah melakukan query SQL untuk mendapatkan nilai yang digunakan untuk menghitung jumlah data dan jumlah halaman. Metode yang kita gunakan adalah *query\_SQL(2, 'id\_forum', \$forum)* arti dari metode ini sama dengan query berikut

```
SELECT * FROM tb_topik WHERE id_forum='$forum' ;
```

Lihat kembali kode dari class halaman pada bagian metode *query\_SQL* jika anda lupa. Setelah melakukan query SQL maka kita sudah mendapatkan nilai untuk digunakan menghitung jumlah baris/data yang ada. Jadi kita gunakan metode *get\_jml\_data()* untuk mendapatkan jumlah baris/data. Nilai dari jumlah data sudah kita dapatkan maka kita dapat menghitung jumlah halaman menggunakan metode *get\_jml\_hal()*. Sekedar mengingatkan nilai dari jumlah halaman kita dapat dengan cara

Jumlah Halaman = jumlah data/baris : topik per halaman

Jika hasilnya koma maka dibulatkan karena kita menggunakan fungsi *ceil()*. Sebelum menampilkan hasil kita cek dulu jumlah data yang dikembalikan jika 0 maka tidak ada topik maka keluar dari rutin script.

Jika *\$jml\_data* lebih dari nol maka tampilkan seluruh topik menggunakan looping while. Namun sebelumnya kita perlu mendapatkan berapa jumlah record yang harus ditampilkan. Untuk itu kita gunakan metode *get\_record()*. Setelah jumlah record kita dapatkan kita dapat melakukan query menggunakan metode

---

```
$hasil = $hal->query_SQL(3, 'id_forum', $forum, 'id_topik');
```

Arti dari metode query\_SQL() diatas sama dengan query SQL berikut

```
SELECT * FROM tb_topik WHERE id_forum='$forum' ORDER BY id_topik DESC
LIMIT $record, $pph;
```

Selanjutnya kita menampilkan hasil dengan looping while. Pada blok kode looping while kita membuat checkbox yang berisi id dari masing-masing topik. Ini berguna untuk melakukan penghapusan pada topik tertentu yang dianggap tidak perlu.

Setelah looping kita menutup tabel lalu menampilkan link nomor halaman. Sebelumnya kita memanggil terlebih dahulu metode set\_hal() untuk memberi nilai pada before, next dan last. Langkah yang terakhir adalah menampilkan link nomor halaman dengan memanggil metode show\_page("topik.php?proses=topik&forum=\$forum"). Metode ini memerlukan satu parameter yaitu halaman yang dibuka dan data-data lainnya jika diperlukan. Pada kasus diatas isi dari parameter kita adalah "topik.php?proses=topik&forum=\$forum". Ini karena halaman yang ingin kita tampilkan daftar topik maka isinya adalah topik.php?proses=topik dan karena daftar topik yang ingin kita tampilkan pada forum tertentu. Maka kita juga memasukkan forum pada URL.

Contoh hasil setelah halaman dijalankan adalah

```

```

Case yang terakhir adalah *hapus*. Pada case ini kita mengambil data yang berupa array dari form daftar topik. Isinya adalah id dari topik yang akan kita hapus. Untuk menghapus setiap id topik yang dikirim kita gunakan looping foreach seperti berikut

```
foreach ($id as $sampah)
```

Pada blok kode looping foreach kita melakukan query DELETE sebagai berikut

```
$hasil = mysql_query("DELETE FROM tb_topik WHERE id_topik='$sampah'");
```

File berikutnya yang akan kita buat adalah *reply.php*. File ini berfungsi untuk menampilkan daftar reply dan menghapusnya. Prinsip kerja file ini sama dengan file sebelumnya yaitu *topik.php*. Untuk membuat file ini pada PHP DESIGNER 2005 klik menu **File > New > PHP** lalu ketik kode berikut

#### Script dari file *reply.php*

```
<?php
```

```

** Nama File.....: reply.php ***
** Penulis.....: Rio Astamal ***
** Tanggal.....: 12-10-2005 ***
** Penjelasan....: Untuk memanage reply ***

// panggil session_start() karena berhubungan dengan session
session_start();

// panggil file header.php
include('../inc/header.php');

// panggil file fungsi.php
include('../inc/fungsi.php');

// panggil file class_forum.php
```

```

include('../inc/class_forum.php');

// panggil file konfig.php
include('../inc/konfig.php');

// koneksi ke MySQL Server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

// filter isi dari proses
$proses = filter_str($proses);

// cek isi session admin, untuk mengetahui sudah login atau belum
if (!cek_session('admin'))
 echo "<p>Anda belum login. Login dulu.</p>\n";
else
{

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'view':
 ****/

 // dapatkan data dari URL
 $page = $_GET['page'];
 if ($page == '')
 $page = 0;

 $page = filter_str($page);

 // panggil class halaman
 $hal = new halaman; // buat objek halaman
 $hal->set_tabel('tb_topik'); // tentukan tabel yang digunakan
 $hal->set_page($page); // data halaman yang dihandel
 $pph = $hal->set_pph($a_tph); // topik per halaman (di konfig.php)

 $hal->query_SQL(1); // lakukan query SQL
 $jml_data = $hal->get_jml_data(); // dapatkan jumlah data (topik)
 $hal->get_jml_hal(); // dapatkan jumlah halaman

 // cek jumlah data/baris yang dikembalikan
 if ($jml_data == 0)
 exit("Tidak ada topik. Topik");

 // buat header tabel
 echo "<h2>Daftar Jumlah reply untuk masing-masing Topik</h2>\n"
 . "<table border='0' cellpadding='4'>\n"
 . " <tr bgcolor='#d7e7f3'>\n"
 . " <td>Topik</td><td>ID Forum</td>\n"
 . " <td>Jumlah Reply</td>\n"
 . " </tr>\n";

 $record = $hal->get_record(); // dapatkan jumlah record yang harus ditampilkan
 $hasil = $hal->query_SQL("SELECT * FROM tb_topik ORDER BY id_topik DESC LIMIT
 $record, $pph");
}

```

```

// tampilkan semua topik dengan looping
while ($data = mysql_fetch_array($hasil))
{
 //buat variabel
 $id = $data['id_topik'];
 $judul = $data['jdl_topik'];
 $forum = $data['id_forum'];

 // hitung jumlah reply masing-masing topik, query dibawah ini
 $query = mysql_query("SELECT * FROM tb_reply WHERE id_topik='$id'");

 // hitung jumlah baris query diatas
 $jml_reply = mysql_num_rows($query);

 // buat link reply untuk masing-masing topik
 $link = "Lihat";

 echo "<tr>\n".
 . "<td>$judul</td>\n<td>$forum</td><td>$jml_reply [$link] </td>\n";
}

// tutup tabel dan tampilkan link nomor halaman jika ada
echo "<tr bgcolor='#d7e7f3' height='15'><td colspan='3'></td></tr>\n".
 . "</table>\n\n";

$hal->set_hal(); // untuk memberi nilai Last, before dan next
// tampilkan link halaman
echo $hal->show_page('reply.php?proses=view')
. "<p>Home</p>\n";

break;
***** */

***** */
case 'reply':
***** */

// ambil data dari URL
$topik = filter_str($_GET['topik']);
$page = filter_str($_GET['page']);

if ($page == '')
 $page = 0;

echo "<h2>Daftar Reply</h2>\n";

// panggil class halaman untuk menampilkan link halaman
$hal = new halaman; // buat objek halaman
$hal->set_tabel('tb_reply'); // tentukan tael yang digunakan
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph($a_rph); // reply per halaman (di konfig.php)

$hal->query_SQL(2, 'id_topik', $topik); // lakukan query SQL
$jml_data = $hal->get_jml_data(); // dapatkan jumlah data (topik)
$hal->get_jml_hal(); // dapatkan jumlah halaman

// cek jumlah data/baris yang dikembalikan
if ($jml_data == 0)
 exit("Tidak ada reply untuk topik ini. Reply");

// buat form dan header tabel

```

```

echo "<form action='reply.php?proses=hapus' method='post'>\n"
 . "<table border='0' cellpadding='4'>\n";

$record = $hal->get_record(); // dapatkan jumlah record
$hasil = $hal->query_SQL(3, 'id_topik', $topik, 'id_reply');

// lakukan looping untuk menampilkan semua entry
while ($data = mysql_fetch_array($hasil))
{
 // buat variabel
 $id_rep = $data['id_reply'];
 $id_top = $data['id_topik'];
 $id_for = $data['id_forum'];
 $user = $data['username'];
 $email = $data['email'];
 // gunakan htmlspecialchars() agar tag-tag HTML tidak diproses
 $reply = htmlspecialchars($data['isi_reply']);
 $tgl = $data['tgl_post'];

 // tampilkan masing-masing reply
 echo "<tr bgcolor='#d7e7f3'><td colspan='2'>$tgl

 . | ID Topik: $id_top | ID Forum: $id_for</td></tr>\n"
 . "<tr><td>Pengirim:</td><td>$user</td></tr>\n"
 . "<tr><td>Email:</td><td>$email</td></tr>\n"
 . "<tr><td>Reply:</td>\n"
 . "<td><textarea name='reply' cols='70' rows='4'$reply </textarea>\n"
 . "</td></tr>\n"
 . "<tr><td colspan='2'><input type='checkbox' name='hapus[]'
value='$id_rep'>
 . "Hapus</td></tr>\n";
}

echo "<tr bgcolor='#d7e7f3'><td colspan='2' align='right'>\n"
 . "<input type='submit' value='HAPUS'></td>\n"
 . "</table>\n"
 . "</form>\n\n";

$hal->set_hal(); // beri nilai pada before, next dan last
// tampilkan link nomor halaman
echo $hal->show_page("reply.php?proses=reply&topik=$topik")
 . "<p>Home &nbsp "
 . "Reply</p>\n";

break;
***** */

***** */
case 'hapus':
***** */

// ambil data yang di-post
$hapus = $_POST['hapus']; // bertipe array

// cek apakah user telah memilih item
if (count($hapus) == 0)
 exit("Error: Anda belum memilih item.

 Kembali\n");

// gunakan looping foreach untuk menghapus setiap item
// yang dicek
foreach ($hapus as $sampah)
{
 // gunakan query DELETE untuk menghapus dari database
 $hasil = mysql_query("DELETE FROM tb_reply WHERE id_reply=' $sampah '");
}

```

```

// cek status
if (!$hasil)
 echo "</p>Error: Gagal menghapus reply dengan id: $sampah\n";
else
 echo "Reply dengan id: $sampah berhasil dihapus.
\n";
}

echo "<p>Reply</p>\n";

break;
***** */

} // akhir dari switch

} // akhir dari else

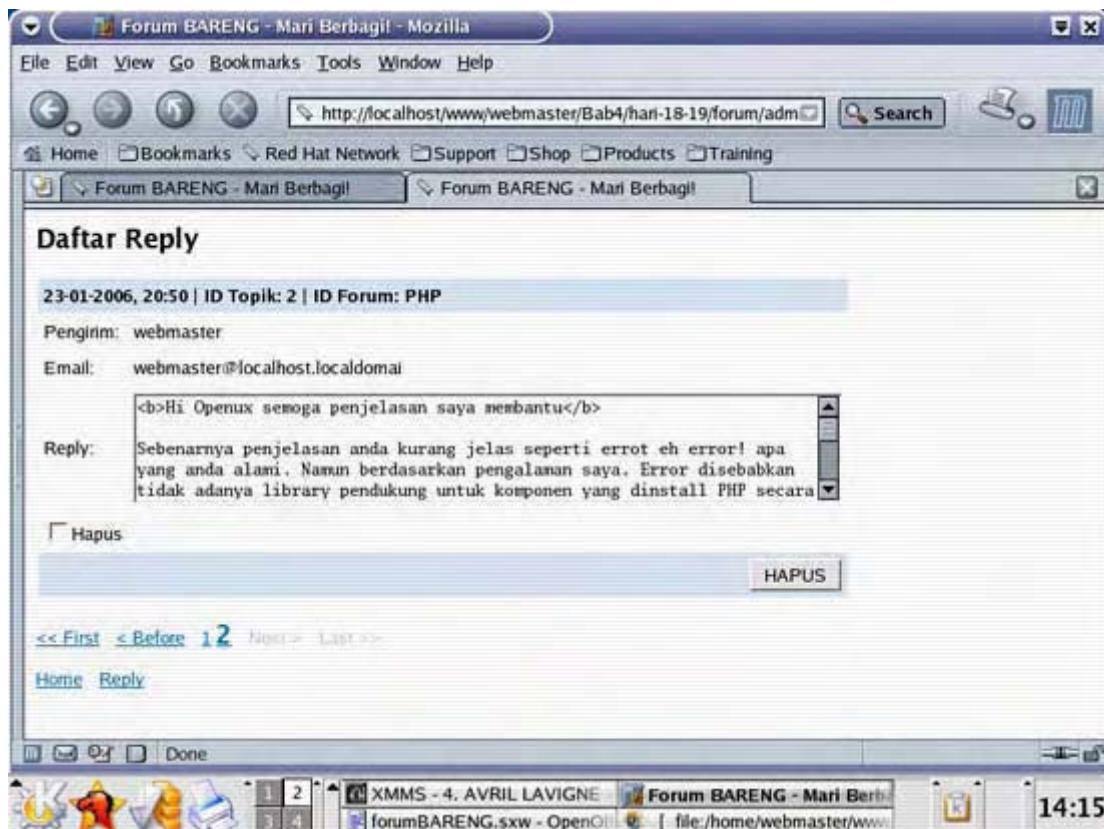
mysql_close(); // tutup koneksi ke MySQL

?>

</body>
</html>

```

Simpan pada direktori **forum\admin** dengan nama **reply.php**. Output dari file ini ditunjukkan oleh gambar 4.15



Gambar 4.15: Daftar reply saat ditampilkan pada control panel

Yang perlu anda ketahui di sini adalah kita menggunakan fungsi baru yaitu `htmlspecialchars()`. Kita menggunakan fungsi ini karena kita membuat forum tentang web. Jadi kemungkinan besar mereka akan mempost source codenya di forum.

Agar source ini tidak diproses oleh PHP maka kita menggunakan fungsi tersebut. Fungsi `htmlspecialchars()` secara otomatis mengubah tag-tag HTML menjadi karakter khusus. Misalnya karakter < diganti dengan &lt; petik " diganti dengan &quot; dan sebagainya.

File terakhir yang akan kita buat pada direktori admin adalah file *member.php*. Tugas file ini adalah untuk menampilkan daftar member dan menghapus member. Langsung saja klik **File > New > PHP** pada PHP DESIGNER 2005 kemudian ketik kode berikut

Script dari file ***member.php***

```
<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk memanage member
***** */

// panggil session_start() karena berhubungan dengan session
session_start();

// panggil file header.php
include('../inc/header.php');

// panggil file fungsi.php
include('../inc/fungsi.php');

// panggil file class_forum.php
include('../inc/class_forum.php');

// panggil file konfig.php
include('../inc/konfig.php');

// koneksi ke MySQL Server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

// filter isi dari proses
$proses = filter_str($proses);

// cek isi session admin, untuk mengetahui sudah login atau belum
if (!cek_session('admin'))
 echo "<p>Anda belum login. Login dulu.</p>\n";
else
{

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*
 case 'view':
 */

// dapatkan data dari URL
$page = $_GET['page'];
if ($page == '')
 $page = 0;


```

```

$page = filter_str($page); // filter isi dari $page

// panggil class halaman
$hal = new halaman;
$tabel = $hal->set_tabel('member'); // pilih tabel member
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph($a_mph); // member per halaman (di konfig.php)

$hal->query_SQL(1); // lakukan query SQL
$jml_member = $hal->get_jml_data(); // dapatkan jumlah member
$hal->get_jml_hal(); // dapatkan jumlah halaman

echo "<h2>Daftar Member untuk ForumBARENG</h2>\n"
. "<p>Jumlah Member: $jml_member</p>\n";

// cek jumlah baris
if ($jml_member == 0)
exit(" <p>Masih belum ada member yang mendaftar.</p>\n");

// buat header tabel dan form
echo "<form action='member.php?proses=hapus' method='post'>\n"
. "<table border='0' cellpadding='4'>\n"
. " <tr bgcolor='#d7e7f3'>\n"
. " <td>Username</td><td>Nama</td>\n"
. " <td>Email</td><td>Posting</td>\n"
. " <td>Login Terakhir</td><td>Hapus?</td>\n"
. " </tr>\n";

$record = $hal->get_record(); // dapatkan jumlah record yang harus ditampilkan
// lakukan query ulang
$hasil = $hal->query_SQL("SELECT * FROM member LIMIT $record, $pph");

// tampilkan hasil dengan looping while
while ($data = mysql_fetch_array($hasil))
{
 //buat checkbox untuk setiap member
 $checkbox = "<input type='checkbox' name='hapus[]' value='".$data[0]."'>\n";
 // $data[0] = $data['username']

 echo "<tr>\n<td>".$data[0]."</td>\n<td>".$data[2]."</td>\n"
 . "<td>".$data[3]."</td>\n<td>".$data[4]."kali</td>\n"
 . "<td>".$data[5]."</td><td align='center'>$checkbox</td>\n</tr>\n";
}

// tutup tabel
echo "<tr bgcolor='#d7e7f3' height='20'>\n"
. "<td colspan='6' align='right'><input type='submit' value='HAPUS'></td>\n"
. "</tr>\n</table>\n</form>\n\n";

$hal->set_hal(); // untuk memberi nilai Last, before dan next

// tampilkan nomor link halaman
echo $hal->show_page('member.php?proses=view')
. "<p>Home &nbsp "
. "Email Semua Member </p>\n";

break;
/*********/
/*********/
case 'hapus':
/*********/
// ambil data yang di-post

```

```

$hapus = $_POST['hapus']; // data ini bertipe array

// cek jumlah item yang dicek
if (count($hapus) == 0)
 exit("<p>Error: Tidak ada member yang dipilih.

\n
 Kembali</p>\n");

// gunakan foreach untuk menghapus setiap item yang dicek
foreach ($hapus as $sampah)
{
 // lakukan query DELETE untuk menghapus data dari database
 $hasil = mysql_query("DELETE FROM member WHERE username='$sampah'");

 // cek status
 if (!$hasil)
 echo "Error: Gagal menghapus $sampah data dari database.
\n";
 else
 echo "Username: $sampah berhasil dihapus dari database.
\n";
}

echo "
Member\n";

break;
***** */

***** */
case 'mail_member':
***** */

// buat form untuk mengirim email
echo "<h2>Kirim Email ke Semua Member</h2>\n"
 . "<form action='member.php?proses=kirim_email' method='post'>\n"
 . "Subject:
\n<input type='text' name='subject' size='50'>

\n"
 . "Isi Email:
\n"
 . "<textarea name='body' cols='75' rows='10'></textarea>

\n"
 . "<input type='submit' value='KIRIM'>\n"
 . "</form>\n\n"
 . "<p>Home &nbsp "
 . "Member</p>\n";

break;
***** */

***** */
case 'kirim_email':
***** */

// ambil data yang di-post
$subject = $_POST['subject'];
$body = $_POST['body'];

// buat header email
$from = 'From: root@localhost'; // coba ganti dengan alamat email anda

// cek apakah masih ada field yang kosong
if (!cek_field($_POST))
 exit("Error: Masih ada field yang kosong.

\n
 Kembali\n");

// lakukan query untuk mendapatkan semua email
$hasil = mysql_query("SELECT email FROM member");

// lakukan looping untuk mengirim ke semua alamat email
while ($data = mysql_fetch_array($hasil))

```

```

{
 // buat variabel
 $to = $data[0]; // alamt email masing-masing member

 // gunakan fungsi mail untuk mengirim email
 mail($to, $subject, $body, $from);
}

echo "<p>Email berhasil dikirim. Member</p>\n";
} // akhir dari switch

} // akhir dari else

mysql_close(); // tutup koneksi ke MySQL

?>

</body>
</html>

```

---

Simpan pada direktori **forum\admin** dengan nama ***member.php***. Output dari file *member.php* ditunjukkan oleh gambar 4.16

#### PENJELASAN SCRIPT

Sebagian isi dari script ini juga sama dengan sebelumnya hanya saja ada beberapa case yang baru. Yaitu case *mail\_member* dan case *kirim\_email*. Case *mail\_member* hanya bertugas menyediakan form untuk subject dan isi. Sedangkan case *kirim\_email* bertugas memproses form dan mengirimnya ke semua alamat email member. Untuk mendapatkan semua alamat email member kita menggunakan query SQL seperti berikut

```
$hasil = mysql_query("SELECT email FROM member");
```

Query diatas hanya mengembalikan satu field/kolom saja yaitu field email. Jadi array 0 merupakan field email bukan username. Untuk mengirim ke semua alamat kita gunakan looping while seperti berikut

```

while ($data = mysql_fetch_array($hasil))
{
 // buat variabel
 $to = $data[0]; // alamt email masing-masing member

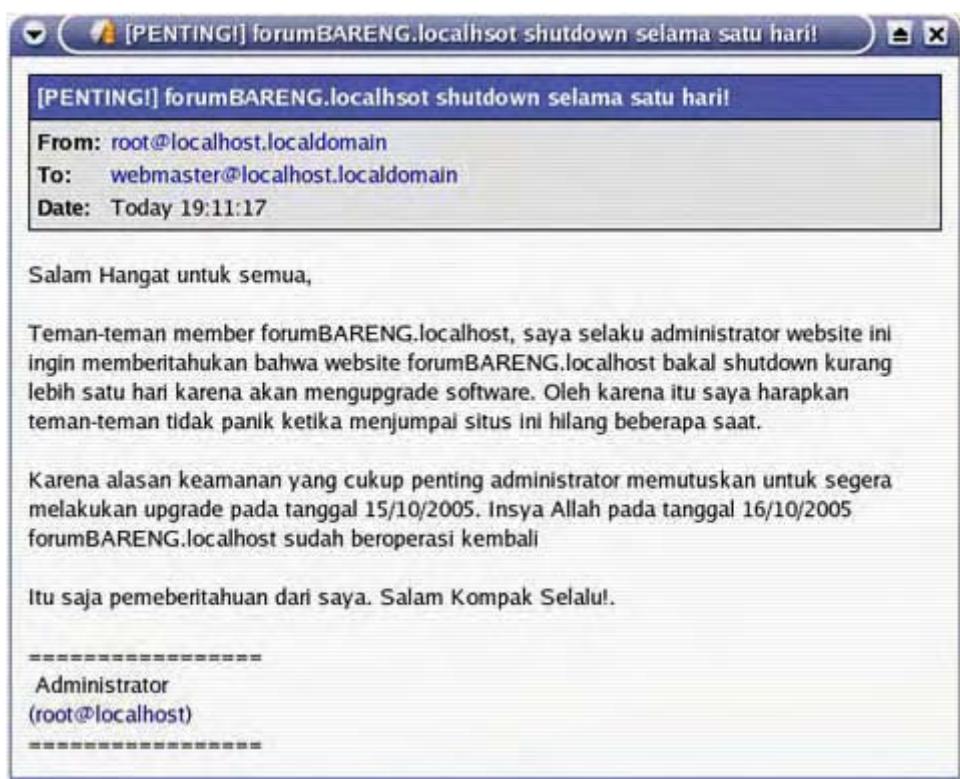
 // gunakan fungsi mail untuk mengirim email
 // hilangkan tanda komentar jika sudah terhubung ke internet
 // mail($to, $subject, $body, $from);
}

```



Gambar 4.16: Halaman daftar member pada control panel

Untuk \$from disini saya isikan dengan [root@localhost](mailto:root@localhost). Anda dapat menggantinya dengan alamat email anda. Gambar 4.17 menunjukkan email yang diterima oleh salah satu member.



Gambar 4.17: Email pemberitahuan yang diterima member dari administrator

Akhir dari file-file yang berada pada direktori **admin**.

File-file berikutnya yang akan kita buat adalah file yang berada pada direktori **forum**. File-file pada direktori ini merupakan halaman-halaman yang dapat diakses oleh pengunjung forum bareng baik yang belum menjadi member (*guest*) atau yang sudah menjadi member.

Tujuan kita membuat sistem membership adalah agar kita dapat mengidentifikasi siapa-siapa yang sering posting, mengirim email pemberitahuan dan sebagainya. Jika user belum login atau ia belum mendaftar menjadi member dari forumBARENG.localhsot maka kita anggap user ID-nya adalah *guest*. Setelah user melakukan login maka user ID-nya kita set menjadi username dari user itu sendiri.

Untuk file yang pertama yang kita buat adalah file untuk halaman utama dari forumBARENG.localhsot. Pada halaman ini terdapat daftar forum yang ada serta jumlah topik dan reply pada masing-masing forum.

Jalankan PHP DESIGNER 2005 klik menu **File > New > PHP** untuk membuat file PHP baru, kemudian ketik kode berikut,

#### Script dari file *index.php*

---

```
<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk menampilkan halaman utama
*****/
```

```
// panggil fungsi session_start() karena berhubungan dengan session
session_start();

// panggil file-file yang diperlukan
include('inc/header.php');
include('inc/fungsi.php');
include('inc/class_forum.php');
include('inc/konfig.php');

// koneksi ke MySQL Server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// keterangan singkat tentang forumBARENG
echo "<center><h2>Selamat Datang di ForumBARENG.localhost</h2>\n"
 . "<p>ForumBARENG tempat dimana pengetahuan adalah "
 . "milik kita bersama. Mari Berbagi dengan yang lain!!!.</p>"
 . "</center>\n"
 . "Pada ForumBareng anda dapat memilih forum yang sesuai dengan "
 . "atau keinginan anda. Kami juga mengharap anda mau menjadi "
 . "member kami. Dengan menjadi member anda akan mendapat banyak "
 . "keuntungan diantaranya informasi terbaru seputar ForumBareng"
 . " dan berbagai newsletter bermanfaat. "
 . "Klik link Daftar untuk mendaftar "
 . "menjadi member kami. ";

// gunakan fungsi cek_session() untuk mengecek id user
if (!cek_session('member'))
 $user_id = 'guest'; // user tersebut belum login atau bukan member
else
 $user_id = $_SESSION['member']; // user telah login

// jika ID-nya guest tampilkan link untuk mendaftar
if ($user_id == 'guest')
{
 echo "<p>User ID anda: $user_id.
\nMohon daftar menjadi member "
 . "kami. Daftar</p>\n";
}
else
```

```

echo "<p>User ID anda: $user_id</p>\n";

// buat tabel untuk menu
echo "<table border='0' cellspacing='0' width='100%>\n"
. "<tr bgcolor='#d7e7f3' height='25'>\n"
. " <td>Home \n"
. " Member Login \n"
. "</tr>\n</table>\n";

// buat tabel untuk daftar forum
echo "<h2>Daftar Forum</h2>\n"
. "<table border='0' cellpadding='4' cellspacing='2' width='100%'> \n"
. "<tr bgcolor='#d7e7f3'>\n"
. " <td>Forum</td>\n"
. " <td align='center'>Jumlah Topik</td>\n"
. " <td align='center'>Jumlah Reply</td>\n</td>\n</tr>\n";

// tampilkan semua forum
$hasil = mysql_query("SELECT * FROM tb_forum");
$jml_forum = mysql_num_rows($hasil);

// lakukan looping untuk menampilkan semua forum
while ($data = mysql_fetch_array($hasil))
{
 // buat variabel
 $id_forum = $data[0];
 $jdl_forum = $data[1];

 // dapatkan jumlah topik untuk forum ini
 $query = mysql_query("SELECT * FROM tb_topik WHERE id_forum='$id_forum'");
 $jml_topik = mysql_num_rows($query);

 // dapatkan jumlah reply untuk forum ini
 $query = mysql_query("SELECT * FROM tb_reply WHERE id_forum='$id_forum'");
 $jml_reply = mysql_num_rows($query);

 // buat link untuk menuju topik dari forum yang diklik
 $link = "$jdl_forum ";

 echo "<tr>\n <td>$link</td>\n"
 . "<td align='center'>$jml_topik</td>\n"
 . "<td align='center'>$jml_reply</td>\n</tr>\n";
}

// tutup tabel
echo "<tr bgcolor='#d7e7f3' height='25'><td colspan='3'></td></tr>\n"
. "</table>\n\n"
. "
Total terdapat $jml_forum forum."
. "<p align='center'>© 2005 ForumBARENG.localhost</p>";

mysql_close(); // tutup koneksi ke MySQL server
?>

</body>
</html>

```

Simpan pada direktori **forum** dengan nama **index.php**. Jalankan pada browser anda, output dari file ini ditunjukkan oleh gambar 4.18

## PENJELASAN SCRIPT

Untuk mengecek user tersebut member atau tidak kita menggunakan fungsi `cek_session('member')`. Dimana jika member sudah login maka user ID-nya adalah username member itu sendiri. Namun jika belum login maka user ID-nya adalah guest. Untuk itu kita tampilkan link daftar agar guest menjadi member forumBARENG.

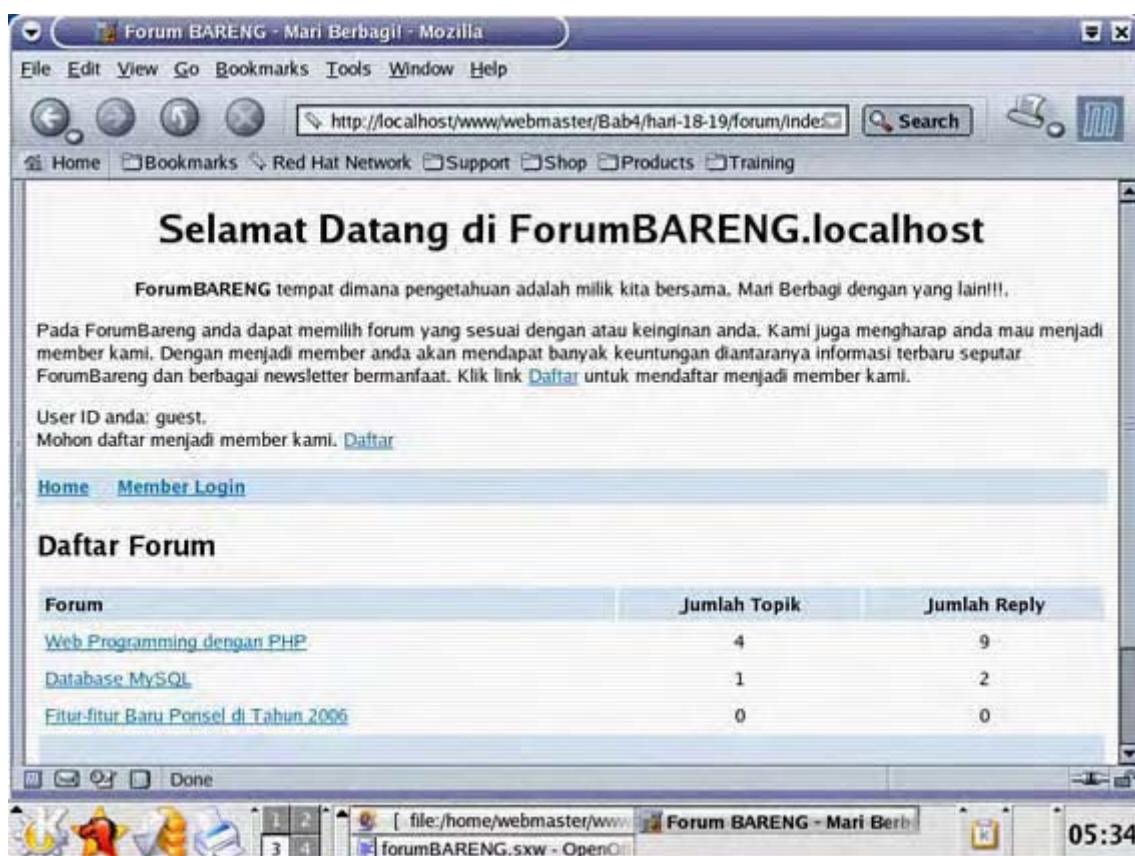
Karena tugas dari file ini hanya satu yaitu menampilkan daftar forum. Maka kita tidak perlu melakukan handel pada proses dengan switch dan case. Kita Mendapatkan daftar forum dengan melakukan query SQL seperti berikut

```
$hasil = mysql_query("SELECT * FROM tb_forum");
```

Untuk menampilkan seluruh forum kita gunakan looping while. Untuk mendapatkan jumlah topik dan reply pada masing-masing forum kita melakukan query berikut,

```
// dapatkan jumlah topik untuk forum ini
$query = mysql_query("SELECT * FROM tb_topik WHERE id_forum='$id_forum'");

// dapatkan jumlah reply untuk forum ini
$query = mysql_query("SELECT * FROM tb_reply WHERE id_forum='$id_forum'");
```



Gambar 4.18: Halaman utama website forumBARENG.localhost

File berikutnya adalah file yang menampilkan daftar topik dan menyediakan form untuk membuat topik baru. Pada bagian daftar topik kita juga menampilkan siapa yang mem-post topik tersebut, kapan di-post, dan berapa reply dari topik tersebut. Buat file PHP baru pada PHP DESIGNER 2005 lalu ketik kode berikut,

Script dari **topik.php**

---

```
<?php
/*
** Nama File.....: topik.php
** Penulis.....: Rio Astamal

```

```

** Tanggal.....: 12-10-2005 ***
** Penjelasan....: Untuk menampilkan daftar topik dan ***
** memproses topik ***
*****/
```

```

session_start();

// panggil file-file yang diperlukan
include('inc/header.php');
include('inc/fungsi.php');
include('inc/class_forum.php');
include('inc/konfig.php');

// koneksi ke MySQL Server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

// cegah XSS
$proses = filter_str($proses);

// gunakan fungsi cek_session() untuk mengecek id user
if (!cek_session('member'))
 $user_id = 'guest'; // user tersebut belum login atau bukan member
else
 $user_id = $_SESSION['member'];

// handel setiap proses dengan switch dan case
switch ($proses)
{
 *****/
 case 'view':
 /***/

 // dapatkan id forum yang akan kita tampilkan
 $forum = $_GET['forum'];

 // dapatkan nomor halaman
 $page = $_GET['page'];
 if ($page == '')
 $page = 0;

 // cegah XSS
 $forum = filter_str($forum);
 $page = filter_str($page);

 // lakukan query untuk mendapatkan keterangan forum
 $hasil = mysql_query("SELECT * FROM tb_forum WHERE id_forum='$forum'");
 // pecah menjadi array
 $data = mysql_fetch_array($hasil);

 // tampilkan keterangan forum
 echo "<h2>Daftar Topik untuk forum ".$data[1]."</h2>\n"
 . "<p>".$data[2]."</p>\n";

 echo "<p>User ID anda: $user_id</p>";

 // panggil class halaman
}

```

```

$hal = new halaman; // buat objek halaman
$hal->set_tabel('tb_topik'); // tentukan tabel yang digunakan
$hal->set_page($page); // data halaman yang di handel
$pph = $hal->set_pph($u_tph); // $u_tph ada di konfig.php

$hal->query_SQL(2, 'id_forum', $forum); // query SQL
// sama dengan SELECT * FROM tb_topik WHERE id_forum='$forum'
$jml_topik = $hal->get_jml_data(); // hitung jumlah data/baris
$hal->get_jml_hal(); // hitung jumlah halaman

// cek jumlah topik
if ($jml_topik == 0)
{
 echo "<p>Tidak ada topik pada forum ini.\n"
 . "

Home &nbsp "
 . "Buat Topik
Baru</p>\n";
}
else
{

// buat header tabel
echo "<table border='0' cellpadding='4' width='100%>\n"
 . "<tr bgcolor='#d7e7f3'>\n"
 . " <td>Posted By</td>\n"
 . " <td>Topik</td>\n"
 . " <td align='center'>Jumlah Reply</td>\n"
 . "<tr>\n";

$record = $hal->get_record(); // dapatkan jumlah record yang harus ditampilkan
$hasil = $hal->query_SQL(3, 'id_forum', $forum, 'id_topik');
// query untuk mengurutkan data dan jumlah recod yang ditampilkan

// buat variabel warna
$color = '#ffe5c3';

// tampilan seluruh dengan looping
while ($data = mysql_fetch_array($hasil))
{
 // buat variabel
 // kita gunakan stripslashes() karena waktu memasukkan pada database
 // kita menggunakan addslashes()
 $user = $data['username'];
 $topik = stripslashes($data['jdl_topik']);
 $tgl = $data['tgl_post'];
 $id = $data['id_topik'];

 // buat link topik
 $link = "" .
 . "$topik";
 $reply = "" .
 . "Reply Topik Ini";

 // hitung jumlah reply
 $query = mysql_query("SELECT * FROM tb_reply WHERE id_topik='$id'");
 $jml_reply = mysql_num_rows($query);

 // jika warna <tr> sekarang '#ffe5cb' ganti dengan '#eddeded'
 // selain itu ganti warnanya dengan '#ffe5c3'
 if ($color == '#ffe5c3')
 $color = '#eddeded';
 else
 $color = '#ffe5c3';
}

```

```

// tampilkan hasil
echo "<tr bgcolor='$color'>\n"
. " <td width='20%'>$user
[$tgl]</td>\n"
. " <td width='65%'>$link</td>\n"
. " <td width='15%' align='center'>$jml_reply
$reply</td>\n </tr>\n";
}

// tutup tabel
echo "</table>\n\n";

$hal->set_hal(); // berikan nilai untuk First, Before, Next dan Last
// tampilkan link nomor halaman
echo $hal->show_page("topik.php?proses=view&forum=$forum")
. "<p>Home &nbsp "
. ""
. "Buat Topik Baru</p>\n"
. "Total terdapat $jml_topik topik pada forum ini.\n"
. "<p align='center'>© 2005 ForumBARENG.localhost</p>\n";
} // akhir dari else

break;
***** */

***** */
case 'tambah':
***** */

// ambil data dari URL
$forum = $_GET['forum'];

// cegah XSS
$forum = filter_str($forum);

// lakukan quer untuk mendapatkan judul forum
$hasil = mysql_query("SELECT * FROM tb_forum WHERE id_forum='$forum'");
// pecah menjadi array
$data = mysql_fetch_array($hasil);

// buat form
echo "<h2>Topik Baru untuk forum ".$data[1]."</h2>\n"
. "<form action='topik.php?proses=proses_tambah' method='post'>\n"
. "Topik/Permasalahan: max. 100 karakter
\n"
. "<input type='text' name='topik' size='80' maxlength='100'>

\n"
. "<input type='hidden' name='forum' value='".$forum.">\n"
. "<input type='submit' value='K I R I M'>\n"
. "</form>\n\n"
. "<p>Home &nbsp "
. "Topik</p>\n";

break;
***** */

***** */
case 'proses_tambah':
***** */

// ambil data yang di-post
// gunakan fungsi addslashes() untuk menghindari eksplorasi
$topik = addslashes($_POST['topik']);
$forum = $_POST['forum'];

// cek field apa masih kosong atau tidak
if (empty($topik))

```

```

exit("Error: Anda belum mengisi topik.

\n
 Kembali\n");

// dapatkan ID user
$username = $user_id;

// filter kata-kata kotor dari topik
// $daftar_kata ada pada file konfig.php
$topik = filter_kata($topik, $daftar_kata);

// tanggal dan waktu sekarang
$tgl = date('d-m-Y, H:i');

// masukkan ke database
$hasil = mysql_query("INSERT INTO tb_topik (id_forum, username, jdl_topik,
tgl_post) VALUES ('$forum', '$username', '$topik', '$tgl')");

// cek status
if (!$hasil)
 echo "<p>Error: Gagal memasukkan data ke database.

\n"
 . "Kembali</p>\n";
else
{
 echo "<p>Topik berhasil dikirim.

\n"
 . "Lihat</p>\n";

 // update nilai posting jika yang mem-post bukan guest
 if ($username != 'guest')
 mysql_query("UPDATE member SET posting=posting+1 WHERE
username='$username'");
}

break;
***** /
}

mysql_close(); // tutup koneksi ke MySQL server

?>

</body>
</html>

```

Simpan pada direktori **forum** dengan nama ***topik.php***. Jalankan pada browser anda, output dari file ini akan terlihat seperti gambar 4.19.



Gambar 4.19: Tampilan halaman daftar topik untuk user

### PENJELASAN SCRIPT

Pada halaman *topik.php* ini terdapat beberapa proses atau case. Diantaranya case *view* menampilkan daftar topik yang ada. Case *tambah* untuk membuat topik baru. Dan yang terakhir case *proses\_tambah* untuk memproses data yang dipost dari form tambah topik.

Pada case *view* kita menampilkan judul tentang forum yang ada menggunakan query berikut

```
$hasil = mysql_query("SELECT * FROM tb_forum WHERE id_forum='".$forum"'");
```

Selanjutnya kita pecah menjadi array untuk mendapatkan judul forum dan keterangan forum. Sebelum melakukan query untuk menampilkan daftar forum kita terlebih dulu mengecek jumlah topik yang ada menggunakan metode *get\_jml\_data()*. Nilai dari metode ini tergantung dari nilai metode *query\_SQL()* sebelumnya yaitu seperti berikut

```
$hal->query_SQL(2, 'id_forum', $forum); // query SQL
```

Jika nilai *\$jml\_topik* sama dengan nol keluar dari script dan tampilkan pesan bahwa belum ada topik. Kita dapat menginstruksikan user untuk membuat topik baru. Jika tidak sama dengan nol maka tampilkan daftar topik.

Untuk menampilkan topik kita menghilangkan tanda backslash menggunakan *stripslashes()*. Ini kita lakukan karena nanti pada saat memasukkan topik ke database kita menggunakan *addslashes()*.

Agar tampilan daftar topik tidak monoton alias membosankan. Maka warnanya kita buat berbeda. Untuk itu kita gunakan statemen *if* untuk menguji kondisi.

```
if ($color == '#ffe5c3')
 $color = '#eddede';
else
 $color = '#ffe5c3';
```

Variabel \$color kita beri nilai sebelum looping while yaitu #ffe5c3. Karena pada blok looping maka isi dari \$color akan berubah antara #ffe5c3 dan #eddeded pada setiap eksekusi. Jadi jika warna sebelumnya #ffe5c3 maka ganti warnanya dengan #eddeded begitu juga sebaliknya.

Tidak ada yang istimewa pada case *tambah*. Fungsi case ini hanya menyediakan form topik baru untuk user. Agar topik yang kita masukkan pada database sesuai dengan forum yang diinginkan maka kita dapatkan forum dari URL. Dan pada saat mengirim data kita memasukkan id forum ini pada input dengan type *hidden*.

Case yang terakhir yaitu case *proses\_tambah*. Pada case ini kita memasukkan topik dengan menggunakan fungsi addslashes(). Sehingga jika user mengetikkan karakter " atau ' akan diubah menjadi \" dan \'. Itulah kenapa pada saat menampilkan topik kita menggunakan fungsi stripslashes().

File berikutnya adalah file yang berfungsi untuk menampilkan daftar reply dari user. Prinsip kerja file ini sama dengan *topik.php*. Pada setiap reply kita menampilkan siapa yang mengirim reply tersebut, alamat email dari pengirim dan kapan dikirim. Pada PHP DESIGNER 2005 klik menu **File > New > PHP** lalu ketik kode berikut

Script dari file *reply.php*

---

```
<?php

/*
** Nama File.....: reply.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk menampilkan daftar reply dan memproses reply
*/
session_start();

// panggil file-file yang diperlukan
include('inc/header.php');
include('inc/fungsi.php');
include('inc/class_forum.php');
include('inc/konfig.php');

// koneksi ke MySQL Server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

$forum = $_GET['forum'];

// cegah XSS
$proses = filter_str($proses);
$forum = filter_str($forum);

// gunakan fungsi cek_session() untuk mengecek id user
if (!cek_session('member'))
 $user_id = 'guest'; // user tersebut belum login atau bukan member
else
 $user_id = $_SESSION['member'];
```

```

// handel setiap proses dengan switch dan case
switch ($proses)
{
 //*****
 case 'view':
 //*****

 // ambil data dari URL
 $topik = $_GET['topik'];
 if ($topik == '')
 exit("Mohon pilih topik.
 Topik\n");

 $page = $_GET['page'];
 if ($page == '')
 $page = 0;

 // cegah XSS
 $topik = filter_str($topik);
 $page = filter_str($page);

 // lakukan query untuk mendapatkan keterangan topik
 $query_topik = mysql_query("SELECT * FROM tb_topik WHERE id_topik='$topik'");
 // pecah menjadi array
 $data = mysql_fetch_array($query_topik);

 echo "<h2>Daftar Reply</h2>\n"
 . "<p>Berikut ini adalah daftar reply untuk,
\n"
 . "Topik: <i>".$data['jdl_topik']
 . "</i>
\n Posted By: ".$data['username']."</p>\n"
 . "<p>User ID anda: $user_id</p><hr>\n";

 // panggil class halaman
 $hal = new halaman; // buat objek halaman
 $hal->set_tabel('tb_reply'); // pilih tabel tb_reply
 $hal->set_page($page); // data halaman yang dihandel
 $pph = $hal->set_pph($u_rph); // pesan per halaman (di konfig.php)

 $hal->query_SQL(2, 'id_topik', $topik); // query SQL
 // sama dengan SELECT * FROM tb_reply WHERE id_topik='$topik'
 $jml_reply = $hal->get_jml_data(); // dapatkan jumlah reply
 $hal->get_jml_hal(); // dapatkan jumlah halaman

 // cek jumlah reply
 if ($jml_reply == 0)
 {
 echo "<p>Belum ada reply untuk topik ini.\n"
 . "

\n"
 . ""
 . "Reply Topik ini &nbsp Home</p>\n";
 }
 else
 {

 // buat header tabel dan form
 echo "<table border='0' cellpadding='4' width='100%>\n"
 . "<tr bgcolor='#d7e7f3'>\n"
 . " <td>Posted By</td>\n"
 . " <td>Reply</td>\n"
 . " <td>Posted On</td>\n"
 . "</tr>\n";

 $record = $hal->get_record(); // dapatkan jumlah record yang ditampilkan
 }
}

```

```

$hasil = $hal->query_SQL("SELECT * FROM tb_reply WHERE id_topik='$topik' LIMIT
 $record, $pph");

$color = '#ffe5c3'; // tentukan warna

// lakukan looping untuk menampilkan semua daftar reply
while ($data = mysql_fetch_array($hasil))
{
 // buat variabel
 // kita gunakan stripslashes() karena waktu memasukkan pada database
 // kita menggunakan addslashes()
 $id = $data['id_reply'];
 $username = $data['username'];
 $topik = $data['id_topik'];
 $email = $data['email'];
 // gunakan htmlspecialchars() agar tag-tag HTML tidak diproses
 $isi_rep = htmlspecialchars(stripslashes($data['isi_reply']));
 $tgl = "".$data['tgl_post']."";

 $link = "<p align='right'><a href='reply.php?proses=reply&
 . "forum=$forum&topik=$topik&id=$id&show=yes'>Reply Topik Ini
 </p>\n";

 // ganti karakter garis baru \n dengan tag
 menggunakan
 // fungsi nl2br()
 $isi_rep = nl2br($isi_rep);

 // cek isi $username, jika bukan guest lakukan query
 if ($username != 'guest')
 {
 $query = mysql_query("SELECT * FROM member WHERE username='$username' ");
 $data2 = mysql_fetch_array($query);
 $jml_post = $data2['posting']; // dapatkan isi dari posting
 }
 else
 $jml_post = '-';

 // ganti warna untuk setiap baris
 if ($color == '#ffe5c3')
 $color = '#dcdcdc';
 else
 $color = '#ffe5c3';

 // tampilkan hasil
 echo "<tr bgcolor='".$color.">\n".
 . " <td valign='top'>$username
[$email]

".
 . " <i>Total Post: $jml_post</i></td>\n".
 . " <td valign='top'>$isi_rep $link</td>\n".
 . " <td valign='top'>$tgl</td>\n".
 . "</tr>\n".
 . "<tr bgcolor='#d7e7f3' height='15'><td colspan='3'></td></tr>\n";
}

// tutup tabel
echo "</table>\n\n";

$hal->set_hal(); // set nilai Before, Next dan Last

// tampilkan link nomor halaman
echo $hal->show_page("reply.php?proses=view&forum=$forum&topik=$topik")
. "<p>Home "
. "Topik</p>\n"
. "
Total terdapat $jml_reply reply pada topik ini. \n"
. "<p align='center'>© 2005 ForumBARENG.localhost</p>\n";

```

```

} // akhir dari else

break;
***** */

// ambil data dari URL
$id = filter_str($_GET['id']);
$topik = filter_str($_GET['topik']);
$show = filter_str($_GET['show']);
$forum = filter_str($_GET['forum']);

// lakukan query untuk mendapatkan keterangan reply
$query_reply = mysql_query("SELECT * FROM tb_reply WHERE id_reply='$id'");
// pecah menjadi array
$data1 = mysql_fetch_array($query_reply);

// gunakan fungsi stripslashes() dan nl2br() pada isi reply
// htmlspecialchars() agar tag HTML tidak diproses
$isi_rep = htmlspecialchars(stripslashes($data1['isi_reply']));
$isi_rep = nl2br($isi_rep);

// lakukan query untuk mendapatkan keterangan topik
$query_topik = mysql_query("SELECT * FROM tb_topik WHERE id_topik='$topik'");
// pecah menjadi array
$data2 = mysql_fetch_array($query_topik);

if ($show == 'yes')
{
 $asal = "<p><table border='0' cellpadding='4' width='80%'>\n"
 . " <tr bgcolor='#ffe5c3'>\n <td>$isi_rep \n"
 . " </td>\n</tr>\n"
 . "</table></p>.<hr>\n";
}
else
 $asal = '';
// buat form dan tabel
echo "<h2>Reply Topik</h2>\n"
 . "<p>Topik: ". $data2['jdl_topik'] . "\n"
 . "
Posted By: ". $data2['username'] . "</p>\n"
 . "<p>User ID anda: $user_id</p><hr>\n"
 . $asal
 . "<h2>Isi Reply anda dibawah ini</h2>\n"
 . "<form action='reply.php?proses=proses_reply' method='post'>\n"
 . "<table border='0' cellpadding='4' bgcolor='#d7e7f3'>\n"
 . " <tr>\n"
 . " <td>Email: </td>\n"
 . " <td><input type='text' name='email'></td>\n</tr>\n"
 . " <tr>\n"
 . " <td>Reply: </td>\n"
 . " <td><textarea name='reply' cols='75' rows='10'></textarea>\n"
 . " </td>\n</tr>\n"
 . " <tr>\n"
 . " <td></td><td><input type='submit' value='R E P L Y'></td>\n"
</tr>\n"
 . "</table>\n"
 . "<input type='hidden' name='forum' value='".$forum.">\n"
 . "<input type='hidden' name='topik' value='".$topik.">\n"
 . "</form>\n"

```

```

. "<p>Home &nbsp "
. ""
. "Back To Reply</p>";

break;
/************/

/* **** */
case 'proses_reply':
/* **** */

// ambil data yang di-post
$email = $_POST['email'];
$topik = $_POST['topik'];
$forum = $_POST['forum'];
// gunakan fungsi addslashes() untuk memberi tanda \ didepan petik
// untuk mencegah SQL INJECTION
$reply = addslashes($_POST['reply']);

// filter kata-kata kotor
// $daftar_kata ada pada konfig.php
$reply = filter_kata($reply, $daftar_kata);

// cek apakah semua field sudah terisi
if (!cek_field($_POST))
 exit("Error: Masih ada field yang kosong.

\n
 Kembali\n");

// cek kevalidan email
if (!cek_email($email))
 exit("Error: Email anda tidak valid.

\n
 Kembali\n");

// buat tanggal sekarang
$tgl = date('d-m-Y, H:i'); // format DD-MM-YYYY, HH:MM

// tentukan user yang dimasukkan
$username = $user_id; // jika belum login maka guest

// jika tidak ada kesalahan masukkan ke database
$hasil = mysql_query("INSERT INTO tb_reply (id_topik, id_forum, username, email,
isi_reply, tgl_post) VALUES ('$topik', '$forum', '$username', '$email',
'$reply', '$tgl')");

// cek status
if (!$hasil)
 echo "Error: Gagal memasukkan data ke database.

\n
 .Kembali\n";
else
{
 echo "Reply berhasil dipost.

\n"
 . "Lihat
 \n";

 // update nilai posting jika user bukan guest
 if ($username != 'guest')
 mysql_query("UPDATE member SET posting=posting+1 WHERE
username='$username'");
}

break;
/************/
} // akhir dari switch

```

```

mysql_close(); // tutup koneksi ke MySQL server

?>

</body>
</html>

```

Simpan pada direktori **forum** dengan nama **reply.php**. Coba jalankan pada browser anda, output dari file tersebut ditunjukkan oleh gambar 4.20

### PENJELASAN SCRIPT

Saya rasa case yang perlu kita perhatikan adalah pada case *reply*. Pada case ini kita mengambil seluruh variabel data dari URL yaitu forum, topik dan show. Untuk show kita peroleh jika user mengklik link Reply Komenter Ini. \$show berguna untuk menguji kondisi apakah komentar yang ingin kita reply ditampilkan atau tidak. Nilai dari show yang dikirim dari case *view* adalah yes.

Secara otomatis jika kita mempost reply dari halaman reply bukan dari halaman topik maka komentar dari user yang anda pilih akan ditampilkan pada form reply topik. Komentar kita tampilkan menggunakan variabel \$isi\_rep sebagai berikut

```
$isi_rep = stripslashes(nl2br($data1['isi_reply']));
```

fungsi nl2br() merupakan fungsi built-in PHP untuk mengubah karakter garis baru \n menjadi tag XHTML <br />. Pada form ini kita memberitahu user, kalau user dapat menggunakan tag HTML. Namun terbatas pada tag <b>, <i>, dan <u> saja.

Case yang terakhir yaitu *proses\_reply* berguna memproses data yang dikirim dari form topik baru. Khusus untuk \$\_POST['reply'] yang isinya adalah reply dari user. Kita gunakan fungsi addslashes() untuk menghindari SQL Injection.



Gambar 4.20: Tampilan halaman daftar reply

Seperti yang sudah dijelaskan sebelumnya kalau tag-tag yang diperbolehkan digunakan pada reply hanya tag <b>, <i> dan <u> jadi isi dari \$reply kita filter kembali menggunakan strip\_tags().

```
$reply = strip_tags($reply, '<i><u>');
```

Parameter kedua pada fungsi diatas adalah tag-tag yang diperbolehkan. Selain ketiga tag tersebut maka akan dihilangkan oleh fungsi strip\_tags(). Setelah kita memfilter tag HTML selanjutnya kiat akan memfilter kata-kata kotor yang mungkin ditulis user.

```
$reply = filter_kata($reply, $daftar_kata);
```

fungsi filter\_kata() merupakan fungsi buatan kita sendiri yang sudah pernah kita bahas pada file fungsi.php. \$daftar\_kata merupakan file yang menyimpan daftar kata-kata kotor. Isi dari variabel ini kita tentukan pada file *konfig.php*.

Kemudian kita mengecek setiap filed apakah masih ada yang salah. Jika tidak ada maka masukkan reply ke database. Jika user bukan guest berarti ia member forumBARENG jadi update nilai dari posting user tersebut dengan cara menambah satu.

Selanjutnya kita akan membuat file untuk login member. Untuk urusan ini kita hanya menggunakan satu file saja. Mengingat sebenarnya fungsinya hanya untuk menset user ID dari member tersebut dan mengupdate profil. Untuk membuatnya klik menu **File > New > PHP** pada PHP DESIGNER 2005 lalu ketik kode berikutnya

#### Script dari file *member.php*

---

```
<?php

/*
** Nama File.....: member.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk memproses login dan update
** profil member
***** */

session_start();

// panggil file-file yang diperlukan
include('inc/header.php');
include('inc/fungsi.php');
include('inc/class_forum.php');
include('inc/konfig.php');

// koneksikan ke MySQL Server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'login';

// filter isi dari proses
$proses = filter_str($proses);

// handel setiap proses dengan case dan switch
switch ($proses)
{
```

```

/*
case 'login':
*/

// gunakan fungsi cek session untuk mengecek apakah user sudah
// login atau belum
if (!cek_session('member'))
{
 echo "<h1>ForumBARENG.localhost Login</h1>\n"
 . "<p>Untuk masuk member area isikan username dan password anda</p>\n"
 . "<form action='member.php?proses=proses_login' method='post'>\n"
 . "<table border='0' cellpadding='4'>\n"
 . " <tr bgcolor='#d7e7f3' height='20'>\n"
 . " <td colspan='2'></td>\n</tr>\n"
 . " <tr>\n<td>Username: </td>\n"
 . " <td><input type='text' name='username'></td>\n</tr>\n"
 . " <tr>\n<td>Password: </td>\n"
 . " <td><input type='password' name='password'></td>\n</tr>\n"
 . " <tr>\n<td></td>\n<td><input type='submit' value='LOGIN'>
</td>\n</tr>\n"
 . "<tr bgcolor='#d7e7f3' height='20'><td colspan='2'></td>\n</tr>\n"
 . "</table>\n</form>\n";

 echo "<p>Belum menjadi member? Daftar\n"
 . " Sekarang juga.</p>\n"
 . "<p>Lupa password? Klik disini

\n"
 . "Home\n"
 . "<p align='center'>© 2005 ForumBARENG.localhost</p>\n";
}
else
{
 $user_id = $_SESSION['member'];
 // lakukan query untuk mendapatkan beberapa informasi
 $hasil = mysql_query("SELECT * FROM member WHERE username='$user_id'");
 $data = mysql_fetch_array($hasil); // pecah data menjadi array

 $nama = $data['nama'];
 $last = $data['last_login'];

 // jika user sudah login tampilkan berikut
 echo "<h1>ForumBARENG.localhost MEMBER AREA</h1>\n"
 . "<p>$nama Selamat Datang di Member Area</p>\n"
 . "<p>Login Terakhir: $last.</p>\n"
 . "<p>Update Profil</p>\n"
 . "<p>Halaman Utama</p>\n"
 . "<p>Logout</p>\n"
 . "<p align='center'>© 2005 ForumBARENG.localhost</p>\n";
}

break;
*/
/*
case 'proses_login':
*/

// ambil data sekaligus filter
$username = filter_str($_POST['username']);
$password = filter_str($_POST['password']);

// enkripsi password
$password = balik_md5($password);

```

```

// gunakan fungsi login untuk mencocokkan username dan password
if (!login('member', $username, $password))
 echo "Username atau Password salah.

\n"
 . "Kembali\n";
else
{
 // buatkan session karena berhasil login
 $_SESSION['member'] = $username;
 echo "Login berhasil. Klik di sini untuk "
 . "masuk ke member area.";
}

break;
***** */

***** */
case 'update':
***** */

$user_id = $_SESSION['member'];

// lakukan query untuk mendapatkan profil dari user yang bersangkutan
$hasil = mysql_query("SELECT * FROM member WHERE username='$user_id'");

// pecah setiap data
$data = mysql_fetch_array($hasil);

// buat form dan tabel
echo "<h1>UPDATE PROFIL</h1>\n"
 . "<p>Berikut ini adalah data pribadi anda: </p>\n"
 . "<form action='member.php?proses=proses_update' method='post'>\n"
 . "<table border='0' cellpadding='4'>\n"
 . " <tr bgcolor='#d7e7f3' height='20'><td colspan='2'></td>\n </tr>\n"
 . " <tr>\n <td>Username: </td>\n"
 . " <td><input type='text' name='username' value='".$data[0]."'"
 . " readonly='yes'></td>\n</tr>\n"
 . " <tr>\n <td>Password: </td>\n"
 . " <td><input type='password' name='pass1'></td>\n</tr>\n"
 . " <tr>\n <td>Ulangi: </td>\n"
 . " <td><input type='password' name='pass2'></td>\n</tr>\n"
 . " <tr>\n <td>>Nama: </td>\n"
 . " <td><input type='text' name='nama' value='".$data[2]."'> </td>\n</tr>\n"
 . " <tr>\n <td>Email: </td>\n"
 . " <td><input type='text' name='email' value='".$data[3]."'></td>\n"
 . "</tr>\n"
 . "<tr>\n <td>Total Post: </td><td>".$data[4]."</td>\n</tr>\n"
 . "<tr>\n <td></td><td><input type='submit' value='UPDATE'></td>\n </tr>\n"
 . "<tr bgcolor='#d7e7f3' height='20'>\n<td colspan='2'></td>\n </tr>\n"
 . "</table>\n"
 . "</form>\n\n"
 . "<p>Home &nbsp Member</p>\n";

break;
***** */

***** */
case 'proses_update':
***** */

// ambil data yang di-post
$username = $_POST['username'];
$pass1 = $_POST['pass1'];
$pass2 = $_POST['pass2'];

```

```

$nama = $_POST['nama'];
$email = $_POST['email'];

$back = "Kembali\n";

// cek apakah semua field sudah terisi
if (!cek_field($_POST))
 exit("Error: Masih ada field yang kosong.

\n$back");

// cek apakah password 1 sama dengan password 2
if ($pass1 != $pass2)
 exit("Error: Password tidak sama.

\n$back");

// cek kevalidan email
if (!cek_email($email))
 exit("Error: Email tidak valid.

\n$back");

//enkripsi password
$pass1 = balik_md5($pass1);

// jika tidak ada kesalahan update database
$hasil = mysql_query("UPDATE member SET password='$pass1', nama='$nama',
email='$email' WHERE username='$username'");

// cek status
if (!$hasil)
 echo "Error: Gagal mengupdate database.

\n$back";
else
 echo "Profil berhasil diupdate.

\n$back";

break;
***** */

***** */
case 'logout':
***** */

// update isi dari last_login pada database
$user_id = $_SESSION['member'];
$tgl = date('d-m-Y, H:i'); // format DD-MM-YYYY, HH:MM
$hasil = mysql_query("UPDATE member SET last_login='$tgl' WHERE
username='$user_id'");

// panggil fungsi logout
if (!logout('member'))
 echo "Anda belum Login. Login dulu.\n";
else
 echo "Anda telah logout dari sistem. Login
kembali.\n";

break;
***** */
}

mysql_close(); // tutup koneksi ke MySQL server

?>

</body>
</html>

```

Simpan pada direktori **forum** dengan nama **member.php**. Jalankan pada browser anda, jika tidak ada kesalahan output dari file ini akan seperti gambar 4.21

## PENJELASAN SCRIPT

Sama seperti prinsip kerja dari file *index.php* pada halaman administrator. Pertama kita mengecek isi dari session member. Jika belum terisi maka user belum login jadi tampilkan form login. Sebaliknya jika sudah terisi maka user sudah login jadi tampilkan menu untuk member area.

Pada halaman member area kita menyediakan tiga menu yaitu menu untuk mengupdate profil, menu untuk kembali ke halaman utama forum, dan menu untuk logout. Jika anda kembali ke halaman menu utama forum, sekarang ID anda bukan guest lagi melainkan username anda.



Gambar 4.21: Halaman utama member area

Pada halaman ini terdapat lima proses yaitu *login*, *proses\_login*, *update*, *proses\_update* dan *logout*. Case *login* dan *update* hanya bertugas untuk menampilkan form masing-masing untuk form *login* dan form *update*.

Pada case *login* khususnya pada blok else atau halaman utama member. Kita menampilkan nama dari member dan kapan ia terakhir kali login ke member area. Data ini kita peroleh dengan melakukan query berikut

```
$user_id = $_SESSION['member'];
// lakukan query untuk mendapatkan beberapa informasi
$hasil = mysql_query("SELECT * FROM member WHERE username='$user_id'");
```

\$user\_id menunjukkan username dari member yang sedang login. Jadi data yang dihasilkan sudah tentu data dari member tersebut.

Pada case *proses\_login* kita mengambil data yang dipost dari form login yaitu username dan password. Untuk mencocokkan data kita menggunakan fungsi *login()* berikut

```
if (!login('member', $username, $password))
```

Arti dari *member* pada parameter pertama menunjukkan tabel yang kita gunakan. Jadi hasil dari fungsi tersebut *false*. Maka username dan password tidak cocok. Oleh karena itu jangan buatkan session untuk user ini. Sebaliknya jika cocok maka buatkan session bernama *member* yang isinya adalah username dari member tersebut.

Pada case *logout* kita mengupdate isi dari *last\_login* pada tabel member yang usernamenya merupakan member yang sedang login. Ini agar saat masuk ke member area kita dapat menampilkan waktu terakhir kali user tersebut login.

File yang berikut ini bertugas untuk menyediakan form pendaftaran/registrasi sekaligus memprosesnya. Untuk membuatnya klik menu **File > New > PHP** pada PHP DESIGNER 2005, kemudian salin script kode berikut

#### Script dari file ***daftar.php***

---

```
<?php

/*
** Nama File.....: daftar.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk menampilkan dan memproses form registrasi
*/
session_start();

// panggil file-file yang diperlukan
include('inc/header.php');
include('inc/fungsi.php');
include('inc/class_forum.php');
include('inc/konfig.php');

// koneksi ke MySQL Server
if (!koneksi())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form';

// filter proses
$proses = filter_str($proses);

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*
 */
 case 'form':
 /*
 */

 // buat form dan tabel
 echo "<h2>Form Registrasi</h2>\n"
 . "<p>Silahkan isi semua field di bawah ini lalu tekan DAFTAR</p>\n"
 . "<form action='daftar.php?proses=proses_form' method='post'>\n"
 . "<table border='0' cellpadding='4'>\n"
 . " <tr bgcolor='#d7e7f3' height='20'>\n"
 . " <td colspan='2'></td>\n</tr>\n"
 . " <tr><td>Username: </td>\n"
 . " <td><input type='text' name='username' maxlength='16'></td>\n</tr>\n"
 . " <tr><td>Password: </td>\n"
 . " <td><input type='password' name='pass1' maxlength='16'></td>\n</tr>\n"
 . " <tr><td>Ulangi: </td>\n"
 . " <td><input type='password' name='pass2' maxlength='16'></td>\n</tr>\n"
 . " <tr><td>Nama Lengkap: </td>\n"
 . " <td><input type='text' name='nama'></td>\n</tr>\n"
 . " <tr><td>Email: </td>\n"
 . " <td><input type='text' name='email'></td>\n</tr>\n"

```

```

. " <tr><td></td><td><input type='submit' value='DAFTAR'></td> </tr>\n"
. " <tr bgcolor='#d7e7f3' height='20'><td colspan='2'></td></tr>\n"
. "</table>\n"
. "</form>\n\n"
. "<p>Home &nbsp </p>"
. "<p align='center'>#169 2005 ForumBARENG.localhost</p>\n";

break;
***** */

***** */
case 'proses_form':
***** */

// ambil data yang dipost
$username = $_POST['username'];
$pass1 = $_POST['pass1'];
$pass2 = $_POST['pass2'];
$nama = $_POST['nama'];
$email = $_POST['email'];

$back = "Kembali\n";

// cek apakah masih ada field yang kosong
if (!cek_field($_POST))
 exit("Error: Masih ada field yang kosong.

\n$back");

// username dan password hanya boleh berupa alfabet, numerik dan _
if (ereg('^[a-zA-Z0-9_]', $username))
 exit("Error: Username hanya boleh berupa alfanumerik dan _.

\n$back");

if (ereg('^[a-zA-Z0-9_]', $pass1))
 exit("Error: Password hanya boleh berupa alfanumerik dan _.

\n$back");

// cek kesamaan password 1 dan password 2
if ($pass1 != $pass2)
 exit("Error: Password tidak sama.

\n$back");

// cek kevalidan email
if (!cek_email($email))
 exit("Error: Email tidak valid.

\n$back");

// enkripsi password
$pass1 = balik_md5($pass1);

// jika tidak ada kesalahan masukkan ke database
// untuk posting kita set 0 dan last_login -
$hasil = mysql_query("INSERT INTO member VALUES ('$username', '$pass1', '$nama',
'$email', 0, '-')");

// cek status
if (!$hasil)
 echo "Error: Gagal memasukkan data ke database.

\n$back";
else
 echo "Proses registrasi berhasil. Login.\n";

break;
***** */
}

mysql_close(); // tutup koneksi ke MySQL

?>

```

---

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

---

Simpan pada direktori **forum** dengan nama ***daftar.php***.

### PENJELASAN SCRIPT

File ini hanya mememiliki dua proses yaitu proses untuk menyediakan form registrasi dan proses untuk memproses data dari form registrasi. Tidak ada yang menarik dari script ini. Karena kita hanya membuat form registrasi pada case *form* dan memproses form *registrasi* pada case *proses\_form*.

File berikutnya merupakan yang terakhir untuk file yang berada pada direktori *forum*. Juga merupakan yang terakhir untuk website ForumBARENG.localhost yang kita buat. File ini berfungsi untuk mereset password user dan mengirimkan password baru ke alamat email user. Pada PHP DESIGNER 2005 klik menu **File > New > PHP** kemudian ketik kode berikut

#### Script dari file ***lupa\_pass.php***

---

```
<?php

/*
** Nama File.....: lupa_pass.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk mereset password user
***** */

session_start();

// panggil file-file yang diperlukan
include('inc/header.php');
include('inc/fungsi.php');
include('inc/class_forum.php');
include('inc/konfig.php');

// koneksi ke MySQL Server
if (!koneksi_db())
 exit('Error: Gagal melakukan koneksi ke MySQL server.

 Cek kembali settingan untuk host, username dan password');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form';

// filter proses
$proses = filter_str($proses);

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*
 case 'form':
 */

 // buat tabel dan form
 echo "<h2>Form Lupa Password</h2>\n";
 . "<p>Pada form ini anda dapat mereset password anda. Setelah itu gunakan"
 . " password baru tersebut untuk login. Dan mengupdate kembali password "
 . " anda.</p>\n";
 . "<form action='lupa_pass.php?proses=kirim' method='post'>\n";
 . "<table border='0' cellpadding='4'>\n";
 . " <tr bgcolor='#d7e7f3' height='20'><td colspan='2'></td></tr>\n"
```

---

```

. " <tr>\n <td>Username: </td>\n"
. " <td><input type='text' name='username'></td>\n</tr>\n"
. " <tr>\n <td>Email: </td>\n"
. " <td><input type='text' name='email'></td>\n</tr>\n"
. " <tr><td></td><td><input type='submit' value='KIRIM'></td> </tr>\n"
. " <tr bgcolor='#d7e7f3' height='20'><td colspan='2'></td></tr>\n"
. "</table>\n</form>\n\n"
. "<p>Home</p>\n";

break;
/************/
/* **** */
case 'kirim':
/* **** */

// ambil data yang di-post
$username = filter_str($_POST['username']);
$email = $_POST['email'];

// cek kevalidan email
if (!cek_email($email))
 exit("Error: Email tidak valid.

\n
 Kembali\n");

// lakukan query untuk mencocokkan username dan email
$hasil = mysql_query("SELECT * FROM member WHERE username='$username' AND
$email='$email'");

// cek jumlah baris yang dikembalikan
if (mysql_num_rows($hasil) == 0)
 echo "Username atau email anda tidak ada didatabase kami.";
else
{
 // jika cocok maka buat password baru, update database lalu kirim email
 // panggil fungsi pass_acak() untuk mendapatkan password secara acak
 $new_pass = pass_acak();

 // enkripsi password
 $pass_enkrip = balik_md5($new_pass);

 // update password yang ada di database
 $q_update = mysql_query("UPDATE member SET password='$pass_enkrip' WHERE
$username='$username'");

 // cek status
 if (!$q_update)
 exit("Error: Gagal Mengupdate password di database. Email tidak dikirim. "
 . "

\nKontak Admin\n");

 // jika tidak ada kesalahan kirim email
 $to = $email; // kirim ke alamat email user
 $subject = "[Password baru anda di ForumBARENG.localhost]";
 $tgl = date('d-m-Y, H:i');
 $isi = "Dari administrator forumBARENG.localhost\n\n"
 . "Pada tanggal $tgl anda dengan username $username \n"
 . "melakukan request untuk mereset password. Dan dibawah ini \n"
 . "adalah password baru anda untuk masuk ke member area.\n\n"
 . "=====*\n"
 . "Username: $username\n"
 . "Password: $new_pass\n"
 . "=====*\n"
 . "Gunakan password diatas untuk masuk ke member area dan \n";
}

```

```

. "mengupdate password anda agar mudah diingat.\n\n"
. "root@localhost (administrator)\n"
. "=====";
$from = "From: root@localhost"; // ganti dengan email anda

// jika anda tidak memiliki program mail server atau belum di server sebenarnya
// beri komentar pada fungsi mail berikut
mail($to, $subject, $isi, $from);

echo "<p>Password berhasil direset. Cek email anda.

\n"
. "Home</p>\n";

} // akhir dari else

} // akhir dari switch

?>

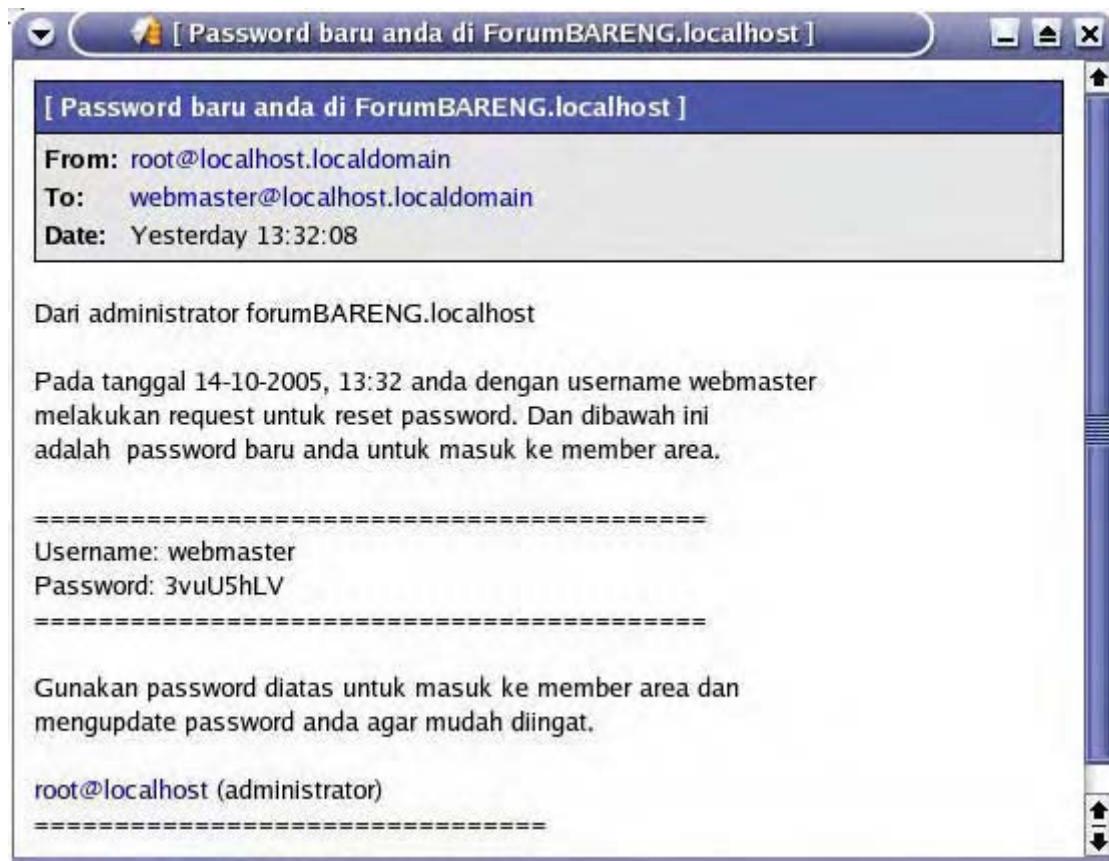
</body>
</html>

```

Simpan pada direktori **forum** dengan nama ***lupa\_pass.php***. Gambar 4.22 menunjukkan email yang diterima salah satu member yang meminta reset password.

### PENJELASAN SCRIPT

Blok kode pada case *form* biasa-biasa saja karena hanya menampilkan form lupa password. Yang menarik adalah pada case *proses\_form* dimana kita melakukan reset password dan mengirim email.



Gambar 4.22: Email yang diterima user yang melakukan reset password

Pada case *proses\_form* hal pertama yang kita lakukan tentu adalah mengambil data yang dipost dari form lupa password. Kita hanya memfilter username karena untuk email sudah

diambil alih oleh fungsi cek\_email(). Kemudian kita mengecek kecocokan antara username dan email dengan data yang ada di database dengan query berikut

```
$hasil = mysql_query("SELECT * FROM member WHERE username='$username' AND email='$email'");
```

Jika username dan email tidak cocok makam jangan mereset password dan jangan mengirim email. Akan tetapi jika cocok maka reset password yang ada didatabase dan kirim email ke user tersebut. Untuk mendapatkan password baru kita memanggil fungsi pass\_acak() sebagai berikut

```
$new_pass = pass_acak();
```

Sekarang isi dari \$new\_pass adalah string acak dengan panjang karakter 8. Isi dari \$new\_pass inilah yang akan kita kirim ke alamat email user. Untuk mereset password yang ada didatabase kita harus mengubah isi dari \$new\_pass menjadi string chiper menggunakan fungsi balik\_md5().

```
$pass_enkrip = balik_md5($new_pass);
```

Isi dari \$pass\_enkrip ini yang kita gunakan untuk mengupdate password user. Jika tidak demikian maka user tidak dapat login karena waktu login kita menggunakan fungsi balik\_md5() untuk enkripsi.

Selanjutnya kita membuat beberapa variabel yang kita gunakan untuk mengirim email menggunakan fungsi mail().

\*\*\*\*\*

## Kalender

27	28	29	30	31	1	2	
3	4	5	6	7	8	9	
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

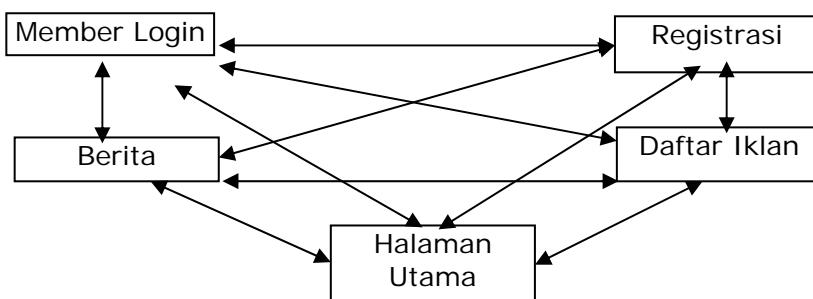
## Hari Ke-21, 22, 23 & 24

Seperti biasa karena sudah menginjak hari baru maka kita ikuti langkah berikut:

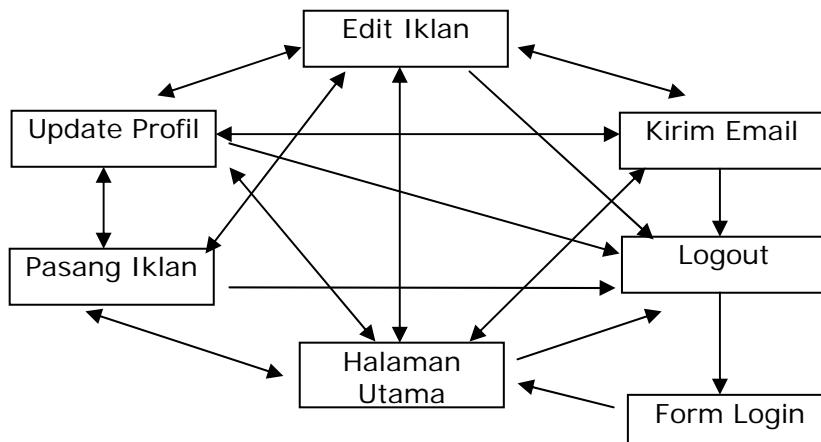
- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab4
- Buat folder baru, namai folder tersebut **hari-21-22-23-24**

# SuperIklan.localhost

Aplikasi ke tiga yang akan kita buat adalah website iklan. Berbeda dengan proyek sebelumnya. Pada proyek ini kita kembali menggunakan skin untuk tampilan website kita. Meskipun sederhana namun kita akan buat website ini lebih rapi dari sebelumnya. Gambar sketsa proyek dapat anda lihat pada gambar dibawah ini.



Gambar 4.23: Sketsa hubungan antar halaman non-member



Gambar 4.24: Sketsa hubungan antar halaman member

### Sketsa Proyek

Kita akan membuat sebuah website iklan baris sederhana. Sketsa website kali ini kita bagi dalam 3 bagian yaitu bagian administrator, bagian user biasa, dan bagian member. Untuk yang pertama mari kita bahas bagian administrator terlebih dahulu.

#### Bagian Administrator

- Untuk masuk ke halaman administrator/control panel admin harus memasukkan username dan password.
- Admin dapat mengedit, menambah, dan menghapus berita.
- Admin dapat menghapus keanggotaan member.
- Admin dapat mengirim email ke semua member.
- Admin dapat menghapus iklan tertentu
- Iklan yang sudah 7 hari otomatis akan dihapus jika admin login.

#### Bagian Member

- Untuk masuk ke halaman member area user harus memasukkan username dan password.
- Member dapat mengupdate profil data dirinya
- Member dapat memasang iklan (maksimal 10 iklan)
- Isi iklan maksimal 500 karakter
- Member dapat mengedit iklan yang telah ia pasang
- Member dapat mengirim iklan berupa email ke seluruh member lainnya (7 hari sekali)

### **Bagian Non-Member**

- User dapat melihat daftar iklan berdasarkan kategori masing-masing
- User dapat melihat daftar iklan kemarin dan dua hari sebelumnya atau lebih
- User dapat mencari iklan dengan memanfaatkan form pencarian iklan
- User dapat melihat daftar berita yang disediakan pengelola
- User dapat mendaftar untuk menjadi member lewat form daftar
- Bagi member dapat login ke member area lewat form login

Sebelum mulai mengerjakan proyek ini lakukan konfigurasi direktori seperti berikut:

Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab4\hari-21-22-23-24

- Buat folder baru dan namai folder tersebut **iklan**
- Masuk pada direktori **iklan** yang baru anda buat
- Buat beberapa folder baru berikut **admin**, **inc**, **member**, dan **template**.

Daftar file yang akan kita buat untuk membangun website superIklan.localhost ditunjukkan oleh tabel 4.3

**Tabel 4.3** Daftar file untuk website superIklan.localhost

Direktori **iklan**

Nama File	Fungsi
index.php	Menampilkan halaman utama superIklan.localhost
login.php	Menampilkan form login untuk masuk ke member area
daftar.php	Menampilkan dan memproses form registrasi user baru
news.php	Menampilkan daftar berita
iklan.php	Menampilkan daftar iklan berdasarkan kategori yang dipilih
cari.php	Menampilkan iklan yang sesuai kriteria user

Direktori **iklan\admin**

Nama File	Fungsi
index.php	Menampilkan form login dan halaman utama admin area
email.php	Menampilkan form kirim email dan memprosesnya
iklan.php	Menampilkan daftar iklan yang dipost member
member.php	Menampilkan daftar member yang telah bergabung
berita.php	Mengedit, menambah dan menghapus berita
db_iklan.sql	File SQL untuk membangun database website superIklan.localhost

Direktori **iklan\inc**

Nama File	Fungsi
index.php	Melakukan redirect ke halaman utama
fungsi.php	Berisi kumpulan fungsi untuk website superIklan.localhost
konfig.php	Bersisi konfigurasi untuk website superIklan.localhost
class_halaman.php	Berisi class halaman untuk menampilkan nomor halaman
class_skin.php	Berisi class skin untuk mengganti tampilan halaman
class_waktu.php	Berisi class waktu untuk manipulasi tanggal

Direktori <b>iklan\member</b>	
Nama File	Fungsi
index.php	Menampilkan halaman utama member area
pasang.php	Menampilkan form pasang iklan dan memprosesnya
edit.php	Mengedit iklan yang telah dipost
email.php	Menampilkan form kirim email dan memprosesnya
profil.php	Menampilkan forum update profil dan memprosesnya

Direktori <b>iklan\template</b>	
Nama File	Fungsi
index.php	Melakukan redirect ke halaman utama
admin_var.php	Kumpulan variabel untuk halaman admin area
member_var.php	Kumpulan variabel untuk halaman member area
var_utama.php	Kumpulan variabel untuk halaman pengunjung/non member
skin_utama.php	File template untuk semua halaman superiklan.localhost

Seperti proyek-proyek sebelumnya kita mulai mengerjakan proyek ini dengan membuat databasenya terlebih dahulu. Oleh karena itu jalankan PHP DESIGNER 2005 lalu klik menu **File > New > SQL** untuk membuat file SQL baru, kemaudian ketik script berikut.

#### Script dari file **db\_iklan.sql**

---

```
-- Buat database iklan
-- Jika sudah ada hapus

DROP DATABASE IF EXISTS iklan;
CREATE DATABASE iklan;

-- aktifkan database iklan
USE iklan;

-- buat tabel untuk admin
CREATE TABLE admin
(
 username varchar(16) PRIMARY KEY,
 password varchar(32)
) TYPE='MyISAM';

-- buat tabel untuk member
CREATE TABLE member
(
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 nama varchar(50),
 email varchar(50),
 alamat varchar(75),
 kota varchar(20),
 telpon varchar(25)
) TYPE='MyISAM';

-- buat tabel untuk iklan
CREATE TABLE tb_iklan
(
 id_iklan int(4) AUTO_INCREMENT PRIMARY KEY,
 kategori varchar(15),
 username varchar(16),
 jdl_iklan varchar(100),
 isi_iklan text,
 tgl_post varchar(20),
 timestamp int(20)
```

```

) TYPE='MyISAM';

-- buat tabel untuk iklan email
CREATE TABLE tb_email
(
 username varchar(16) PRIMARY KEY,
 email varchar(50),
 next_post int(20)
) TYPE='MyISAM';

-- buat tabel untuk berita
CREATE TABLE tb_berita
(
 id_berita int(4) PRIMARY KEY AUTO_INCREMENT,
 jdl_berita varchar(100),
 isi_berita text,
 tgl_berita varchar(20)
) TYPE='MyISAM';

-- buat privilege untuk database ini
GRANT SELECT, UPDATE, DELETE, INSERT ON iklan.*
TO useriklan IDENTIFIED BY 'iklan123';

-- Akhir dari file

```

---

Simpan pada direktori **iklan** dengan nama **db\_iklan.sql**. Untuk mengeksekusi file ini terdapat dua cara yaitu melalui MySQL monitor atau phpMyAdmin. Jika pada MySQL Monitor maka ketikkan perintah berikut,

```
mysql> source C:/Apache2/htdocs/Bab4/hari-21-22-23/iklan/db_iklan.sql
```

Jika pada phpMyAdmin klik tab SQL lalu klik **Browse** arahkah pada alamat seperti diatas.

Untuk selanjutnya kita akan membuat file-file yang berada pada direktori *inc*. File-file pada direktori ini kita buat terlebih dahulu karena merupakan file pendukung untuk file-file lainnya. File pertama yang akan kita buat adalah *index.php*, jadi jalankan PHP DESIGNER 2005 klik **File > New > PHP** untuk membuat file PHP baru kemudian ketik kode berikut.

Script dari file **index.php**

---

```

<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 20-10-2005
** Penjelasan....: Untuk meredirect ke halaman utama
*/
// jika user mencoba masuk ke halaman inc
// kembalikan ia ke halaman utama
header ("Location: ../index.php");

?>

```

---

Simpan pada direktori **iklan\inc** dengan nama **index.php**. Tugas file ini hanya melakukan redirect ke halaman utama. Jika ada user yang mencoba masuk ke direktori *inc*.

File berikutnya yang akan kita buat adalah file yang menyimpan konfigurasi. Seperti banyaknya iklan atau berita yang akan kita tampilkan. Berapa lama iklan harus dihapus dari database dan sebagainya. Buat file PHP baru pada PHP DESIGNER 2005, lalu ketik kode berikut.

---

### Script dari file ***konfig.php***

---

```
<?php

/*
** Nama File.....: konfig.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: menyimpan konfigurasi baik untuk halaman member dan admin
*/
// cegah pengaksesan langsung dari browser
if (eregi('konfig.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php'); // kembalikan ke halaman utama
 exit;
}

/* KONFIGURASI UNTUK ADMIN */
/*
// ganti jumlahnya sesuai keinginan anda
$a_bph = 2; // berita per halaman
$a_mph = 2; // member per halaman
$a_iph = 2; // iklan per halaman

$max_post = 10; // jumlah iklan maximal yang dipost user
$lama_iklan = 7; // lama hari iklan user akan dihapus
$lama_email = 7; // lama hari user boleh mengirim email kembali

/* KONFIGURASI UNTUK USER/MEMBER */
/*
$u_bph = 2; // berita per halaman
$u_iph = 2; // iklan per halaman
$u_jbph = 2; // judul berita per halaman

// AKHIR KONFIGURASI //

?>
```

---

Simpan pada direktori **iklan\inc** dengan nama ***konfig.php***. Konfigurasi untuk administrator kita beri identitas variabelnya dengan tanda a seperti \$a\_bph, \$a\_mph dan sebagainya. Sedangkan untuk user kita beri identitas u seperti \$u\_bph dan sebagainya.

Selanjutnya kita akan membuat file yang akan menyimpan seluruh fungsi yang akan kita panggil hampir pada setiap halaman. Jalankan PHP DESIGNER 2005 klik **File > New > PHP** kemudian ketik kode berikut.

---

### Script dari file ***fungsi.php***

---

```
<?php

/*
** Nama File.....: fungsi.php
** Penulis.....: Rio Astamal
** Tanggal.....: 15-10-2005
** Penjelasan....: menyimpan kumpulan fungsi
*/
// cegah pengaksesan langsung dari browser
if (eregi('fungsi.php', $_SERVER['PHP_SELF']))
```

---

```

exit('Error: Akses ditolak.');

// fungsi untuk mengkoneksikan ke MySQL server
function koneksi_db()
{
 // untuk username = secure dan password = secure123
 @ $koneksi = mysql_connect('localhost', 'root', 'b3naffl3ck');
 // jika gagal melakukan koneksi tampilkan pesan kesalahannya
 if (!$koneksi)
 {
 echo "Error: ".mysql_errno()."
\n";
 echo "Keterangan: ".mysql_error()."
\n";
 exit;
 }
 else
 {
 // pilih database yang digunakan
 mysql_select_db('iklan');
 return true;
 }
}

// cek setiap field apa ada yang kosong
function cek_field($var)
{
 foreach ($var as $field)
 {
 if ($field == '' || !isset($field))
 return false;
 }
 return true;
}

// cek kevalidan email
function cek_email($email)
{
 // fungsi untuk mengecek kevalidan email
 if (ereg('^[a-zA-Z0-9_-]+@[a-zA-Z0-9\-.]+\.[a-zA-Z0-9\-\.\.]+\$', $email))
 return true;
 else
 return false;
}

// fungsi untuk menyaring string selain alfabet, numerik dan _
function filter_str($string, $lainnya='')
{
 if ($lainnya == '')
 $filter = ereg_replace('[^a-zA-Z0-9_]', '', $string);
 else
 $filter = ereg_replace("[^a-zA-Z0-9_$lainnya]", '', $string);

 return $filter;
}

// fungsi untuk mengenkripsi string dengan metode MD5
// dan membalik urutannya
function balik_md5($string)
{
 // untuk membalik urutan string digunakan fungsi strrev()
 $chiper_text = strrev(md5($string));
 return $chiper_text;
}

// fungsi untuk mengecek session

```

```

function cek_session($nama_ses)
{
 // jika session kosong
 if (!isset($_SESSION[$nama_ses]))
 return false; // kembalikan nilai false
 else
 // jika tidak kosong
 return true; // kembalikan nilai true
}

// fungsi untuk login
function login($tabel, $username, $password)
{
 // lakukan query untuk mencocokkan data
 $hasil = mysql_query("SELECT * FROM $tabel WHERE username='$username'
 AND password='$password'");

 // cek baris yang dikembalikan
 if (mysql_num_rows($hasil) == 0)
 return false; // data tidak cocok
 else
 return true; // kembalikan nilai true
}

// fungsi untuk membuat password secara acak
// digunakan untuk mengirim password pada form lupa password
function pass_acak($panjang=8)
{
 $kar = "ABCDEFGHIJKLMNPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz";
 // acak karakter
 srand((double)microtime() * 1000000);
 // lakukan looping sebanyak $panjang
 for ($i=0; $i<$panjang; $i++) // default diulang sebanyak 8x
 {
 $nom_acak = rand() % 53; // untuk mendapatkan nomor acak, pada substr()
 $pass .= substr($kar, $nom_acak, 1); // ambil satu karakter
 }

 return $pass; // kembalikan hasil
}

// fungsi untuk menampilkan tanggal sekarang
function show_tgl()
{
 // buat array nama hari
 $nama_hari = array('Minggu', 'Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat',
'Sabtu');
 // buat array nama bulan
 $nama_bulan = array('Januari', 'Februari', 'Maret', 'April', 'Mei', 'Juni',
'Juli', 'Agustus', 'September', 'Oktober', 'Nopember', 'Desember');

 $tanggal = date('j'); // tanggal sekarang 01-31
 $hari = date('w'); // kode hari 0=minggu, 1=senin dst.
 $bulan = date('n') - 1; // dikurangi satu agar index awal 0
 $tahun = date('Y'); // tahun format 4 digit

 $hari_ini = $nama_hari[$hari]; // string hari ini
 $bulan_ini = $nama_bulan[$bulan]; // string bulan ini

 // gabungkan hasil
 $today = $hari_ini.', '.$tanggal.' '.$bulan_ini.' '.$tahun;

 return $today; // kembalikan hasil
}

```

```
// fungsi untuk logout (menghapus session)
function logout($nama_ses)
{
 // jika session kosong
 if (!isset($_SESSION[$nama_ses]))
 return false; // kembalikan false
 else
 {
 // jika tidak kosong hancurkan session tersebut
 unset($_SESSION[$nama_ses]);
 session_destroy();
 return true; // kembalikan nilai true
 }
}

?>
```

---

Simpan pada direktori **iklan\inc** dengan nama ***fungsi.php***.

### PENJELASAN SCRIPT

Pada proyek kali ini kita melakukan sedikit perubahan pada file fungsi.php. Diantaranya kita menghilangkan fungsi untuk mengecek kata-kata kotor yaitu fungsi filter\_kata(). Dan kita menambahkan fungsi baru yaitu show\_tgl(). Fungsi show\_tgl() ini sebenarnya sudah pernah kita buat pada bab sebelumnya.

Perubahannya adalah pada fungsi konek\_db() dimana pada fungsi ini kita tidak mengembalikan nilai false melainkan langsung mencetak nomor error dan pesan errornya kemudian keluar dari rutin program dengan fungsi exit.

Fungsi filter\_str() juga kita modifikasi sehingga memiliki dua parameter. Satu bersifat wajib dan satu opsional. Isi dari parameter kedua merupakan karakter lain yang mungkin ingin anda tetap pertahankan tanpa terfilter oleh fungsi ini. Ini berguna pada form pencarian iklan dimana user biasanya memasukkan spasi untuk pencarian.

Selanjutnya kita akan membuat file yang menyimpan class untuk tampilan setiap halaman dari website superiklan.localhost yaitu class skin. Class ini sudah pernah kita gunakan berulang-ulang pada proyek bab-bab sebelumnya. Jadi tidak perlu saya jelaskan ulang. Untuk membuatnya seperti biasa, klik **File > New > PHP** pada PHP DESIGNER 2005 lalu ketik kode berikut.

Script dari file ***class\_skin.php***

---

```
<?php

class skin
{
 // deklarasikan properti
 var $tag = array();
 var $file_skin;
 var $halaman;

 // metode untuk menentukan nama tag yang diganti
 // dan string penggantinya
 function ganti_tag($namatag, $str_pengganti)
 {
 $this->tag[$namatag] = $str_pengganti;
 }

 // metode untuk menentukan file template yang digunakan
 function ganti_skin($namafile)
```

```

 {
 $this->file_skin = $namafile;
 }

 // metode untuk mengganti tampilan
 function ganti_tampilan()
 {
 // buka file template menggunakan file()
 $this->halaman = file($this->file_skin);

 // gabungkan setiap baris dengan implode()
 $this->halaman = implode("", $this->halaman);

 // gunakan looping foreach() untuk mengganti setiap tag
 foreach($this->tag as $str_dicari => $str_baru)
 {
 $this->halaman = ereg_replace($str_dicari, $str_baru, $this-> halaman);
 }

 echo $this->halaman; // tampilkan ke layar
 }

}

?>

```

Simpan pada direktori **iklan\inc** dengan nama ***class\_skin.php***.

File berikutnya juga sudah pernah kita buat yaitu class untuk menampilkan nomor halaman. Langsung saja buat file PHP baru pada PHP DESIGNER 2005, kemudian ketik kode berikut.

Script dari file ***class\_halaman.php***

```

<?php

/*
** Nama File.....: class_halaman.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Class untuk menampilkan link nomor
** halaman
***** */

// cegah pengaksesan langsung dari browser
if (eregi('class_forum.php', $_SERVER['PHP_SELF']))
{
 // kembalikan ke halaman utama
 header('Location: ../index.php');
 exit; // keluar dari rutin script
}

class halaman
{

 // deklarasikan properti untuk class
 var $tabel, $page, $file, $per_halaman, $jml_data, $jml_hal, $hasil_query;
 var $record, $last_page, $sebelumnya, $berikutnya, $tampilkan_hal;

 // metode untuk memilih tabel yang digunakan
 function set_tabel($nama_tabel)
 {
 $this->tabel = $nama_tabel;
 }
}

```

```

// metode untuk menentukan data yang di-post
function set_page($halaman)
{
 $this->page = $halaman;
}

// metode untuk menentukan banyaknya pesan per halaman
function set_pph($angka)
{
 $this->per_halaman = $angka;

 return $this->per_halaman;
}

// metode untuk mendapatkan jumlah record yang harus ditampilkan
// pada query SQL. jumlah record = halaman x pesan per halaman
function get_record()
{
 $hasil = $this->page * $this->per_halaman;
 $this->record = $hasil;

 return $this->record;
}

// metode untuk melakukan query sql. digunakan untuk mendapatkan jumlah
// data, jumlah record dan sebagainya
function query_SQL($q = 1, $id='', $isi='', $lainnya='')
{
 // jika parameter pertama berisi 1 lakukan query berikut
 if ($q == 1)
 $query = mysql_query("SELECT * FROM $this->tabel");
 else if ($q == 2) // jika 2 lakukan query berikut
 $query = mysql_query("SELECT * FROM $this->tabel WHERE $id='$isi'");
 else if ($q == 3) // jika 3 lakukan query berikut
 $query = mysql_query("SELECT * FROM $this->tabel WHERE $id='$isi' ORDER BY
$lainnya DESC LIMIT $this->record, $this->per_halaman");
 else
 $query = mysql_query($q); // jika bukan 1,2 atau 3 lakukan query berikut

 $this->hasil_query = $query;

 return $this->hasil_query; // kembalikan hasil dari query
}

// metode untuk mendapatkan jumlah data pada database
function get_jml_data()
{
 $jumlah = mysql_num_rows($this->hasil_query);
 $this->jml_data = $jumlah;

 return $this->jml_data;
}

// metode untuk mendapatkan jumlah halaman
function get_jml_hal()
{
 // untuk menghitung jumlah halaman digunakan fungsi ceil
 // dimana >> Jumlah data : pesan per halaman
 // jika hasilnya koma, maka dibulatkan ke atas
 $jumlah = ceil($this->jml_data / $this->per_halaman);

 $this->jml_hal = $jumlah;
}

```

```

 return $this->jml_hal;
}

// metode untuk mendapatkan halaman
// paling awal, sebelumnya, berikutnya dan paling akhir
function set_hal()
{
 // halaman terakhir kita kurangi satu karena pada nomor
 // karena nilai jml_hal lebih besar 1 dari halaman terakhir
 $this->last_page = $this->jml_hal - 1;

 // link halaman sebelumnya didapat dengan mengurangi nilai
 // halaman sekarang(yang aktif) dengan satu
 $this->sebelumnya = $this->page - 1;

 // link halaman berikutnya didapat dengan menambahkan nilai
 // halaman sekarang dengan satu
 $this->berikutnya = $this->page + 1;
}

// metode untuk menampilkan halaman
function show_page($URL)
{
 // jika jumlah halaman lebih dari satu tampilkan selain itu jangan
 if ($this->jml_hal > 1)
 {
 // jika halaman sekarang 0 (paling awal) jangan tampilkan link
 // first dan before ganti dengan tulisan biasa(warna abu-abu)
 if ($this->page == 0)
 {
 $first = "<< First";
 $back = "< Before";
 }
 else
 {
 $first = "<< First";
 $back = "sebelumnya.'>< Before";
 }

 // jika halaman sekarang sama dengan nilai halaman terakhir
 // jangan tampilkan link last dan next
 if ($this->page == $this->last_page)
 {
 $last = "Last >>";
 $next = "Next >>";
 }
 else
 {
 $last = "last_page.'>Last >>";
 $next = "berikutnya.'>Next >>";
 }

 // tampilkan
 $halaman = "$first $back \n";

 // gunakan looping untuk menampilkan setiap nomor halaman
 for ($i=0; $i<$this->jml_hal; $i++)
 {
 // jika nomor halaman sama dengan halaman yang sedang
 // dibuka tebalkan angka tersebut dan hapus link
 // nomor halaman kita tambah 1 agar nomor awal tidak 0
 if ($i == $this->page)
 $halaman .= " ".intval($i + 1)."";
 else

```

```

 $halaman .= " ".intval($i + 1)." \n";
 }

 $halaman .= " $next $last\n";

 // output dari halaman jika jumlah halamannya lebih dari satu
 // kurang lebih seperti berikut, dimana x adalah nomor halaman
 // << First < Before x x x Next > Last >>
}
else
 $halaman = ''; // artinya halaman yang ada hanya satu

$this->tampilkan_hal = $halaman;

// kembalikan nilai karena akan dicetak ke layar
return $this->tampilkan_hal;
}

} // akhir dari class halaman

?>

```

Simpan pada direktori **iklan\inc** dengan nama ***class\_halaman.php***.

Untuk file berikut ini belum pernah kita buat sebelumnya. Jadi waspadalah, lho kok!. File ini secara garis besar hampir sama dengan fungsi `show_tgl()` pada fungsi.php. Hanya saja karena ini merupakan suatu class maka lebih fleksibel karena kita dapat menentukan tanggal berapa setelah kita tambah atau kurangi sejumlah hari tertentu.

Penasaran? Arahkah mouse anda pada menu **File > New > PHP**, lalu ketik kode berikut.

Script dari file ***class\_waktu.php***

```

<?php

/*
** Nama File.....: class_waktu.php
** Penulis.....: Rio Astamal
** Tanggal.....: 15-10-2005
** Penjelasan....: class untuk memanipulasi tanggal
** baik dikurangi,ditambah atau tetap
***** */

class waktu
{
 // deklarasikan properti
 var $date, $bulan, $year, $jml_hari, $is_kabisat;

 // metode untuk memberi nilai tanggal, bulan, tahun, jumlah
 // hari dalam bulan sekarang
 function set_date()
 {
 $this->date = date('d'); // tanggal 01-28/29/30/31
 $this->bulan = date('m'); // bulan 01-12
 $this->year = date('Y'); // tahun 4 digit
 $this->jml_hari = date('t'); // jumlah hari
 $this->is_kabisat = date('Y') % 4; // jika 0 = kabisat selain itu tidak
 }

 // metode untuk menentukan tanggal dikurangi, ditambah atau tetap
 // 0 = mengurangi, 1 = menambah, 2 = tetap
 function set_mode($pilihan)

```

```

{
 $this->mode = $pilihan;
}

// metode untuk menentukan output tampilan dan angka manipulasi
// format 0 = tanggal biasa, 1 = detik(timestamp)
function set_tgl($format = 0, $angka = 1)
{
 // jika lebih dari 31 keluar dari rutin program
 if ($angka > 31)
 return "Error: Angka terlalu tinggi(max. 31), keluar dari fungsi.";

 $bulan = $this->bulan;
 $thn = $this->year;

 if ($this->mode == 0) // jika mode pengurangan
 {
 $tgl = $this->date - $angka; // tanggal dikurangi

 if ($tgl <= 0) // jika kurang/sama dengan nol, maka bulan sebelumnya
 {
 $bulan = $this->bulan - 1; // kurangi bulan dengan 1

 if ($this->jml_hari == 30) // bulan yang tanggalnya sampai 30
 $tgl = (31 + $this->date) - $angka; // tanggal pada bulan yang tanggalnya
31
 else if ($this->bulan == 1) // januari
 {
 $tgl = (31 + $this->date) - $angka; // tanggal pada bulan DESEMBER
 $bulan = 12; // desember
 $thn = $this->year - 1; // tahun dikurangi 1
 }
 else if ($this->bulan == 3) // maret
 {
 if ($this->is_kabisat == 0) // jika kabisat
 $tgl = (29 + $this->date) - $angka; // tanggal pada februari
 else
 $tgl = (28 + $this->date) - $angka; // tanggal pada februari non kabisat
 }
 else if ($this->bulan == 8) // Agustus
 $tgl = (31 + $this->date) - $angka; // tanggal pada bulan juli
 else if ($this->bulan == 2) // februari
 $tgl = (31 + $this->date) - $angka; // tanggal pada februari
 else
 $tgl = (30 + $this->date) - $angka; // tanggal pada bulan yang tanggalnya
30
 }
 }
 else if ($this->mode == 1)
 {
 $tgl = $this->date + $angka; // angka ditambah

 if ($this->jml_hari == 31 && $bulan != 12) // bulan bertanggal 31 selain
desember
 {
 if ($tgl > 31) // jika melebihi tanggal 31
 {
 $tgl = ($this->date - 31) + $angka; // kurangi 31 dan tambah sesuai
parameter
 $bulan = $bulan + 1; // bulan ditambah 1
 }
 }
 else if ($this->jml_hari == 30) // bulan bertanggal 30
 }
}

```

```

{
 if ($tgl > 30)
 {
 $tgl = ($this->date - 31) + $angka;
 $bulan = $bulan + 1;
 }
}
else if ($bulan == 12) // bulan desember
{
 if ($tgl > 31) // jika melebihi tanggal 31
 {
 $tgl = ($this->date - 31) + $angka;
 $bulan = 1; // bulan kita set januari
 $thn = $thn + 1; // tahun kita tambah satu
 }
}
else if ($bulan == 2) // jika bulan februari
{
 if ($this->is_kabisat == 0) // jika kabisat
 {
 if ($tgl > 29) // jika lebih dari tanggal 29
 {
 $tgl = ($this->date - 29) + $angka; // tanggal = penambahan hari
 $bulan = $bulan + 1; // bulan ditambah 1
 }
 }
 else
 if ($tgl > 28) // jika non kabisat
 {
 $tgl = ($this->date - 28) + $angka;
 $bulan = $bulan + 1;
 }
 }
}
else
 $tgl = $this->date; // tetap

if (strlen($tgl) < 2)
 $tgl = "0$tgl"; // agar berformat 2 digit

if (strlen($bulan) < 2)
 $bulan = "0$bulan"; // agar berformat 2 digit

// cek format yang diinginkan
if ($format == 0 || $format != 1)
 $this->hasil = "$tgl - $bulan - $thn"; // tanggal biasa
else
 $this->hasil = mktime(0, 0, 0, $bulan, $tgl, $thn); // detik

return $this->hasil; // kembalikan hasil
} // akhir dari metode set_tgl()
} // akhir dari class
?>

```

Simpan pada direktori **iklan\inc** dengan nama **class\_waktu.php**. Fungsi dari class ini sebenarnya adalah untuk memainkan tanggal untuk manajemen iklan. Sehingga user dapat memilih iklan pada tanggal berapa yang ingin ia tampilkan.

## PENJELASAN SCRIPT

Class ini terdiri dari tiga metode yaitu `set_date()`, `set_mode()`, dan `set_tgl()`. Selanjutnya mari kita bahas satu per satu. Pertama kita mulai dari metode `set_date()`.

Pada metode `set_date()` kita hanya memberi nilai untuk tanggal, bulan, tahun dan jumlah hari sekarang. Kita juga menentukan apakah tahun ini kabisat atau tidak dapat menggunakan cara manual dengan menggunakan modulus(sisa pembagian) seperti berikut,

```
$this->is_kabisat = date('Y') % 4;
```

isi dari properti `is_kabisat` merupakan hasil dari pembagian tahun sekarang dibagi 4. Jika sekarang tahun 2005 modulusnya adalah 1. Jadi bukan tahun kabisat namun jika modulusnya 0 maka sekarang adalah tahun kabisat.

Metode yang kedua adalah `set_mode()`. Fungsi dari metode ini berpengaruh pada metode terakhir. Karena pada metode ini kita menentukan tanggal yang kita inginkan akan ditambah atau dikurangi. Angka 0 untuk mengurangi tanggal, 1 untuk menambah, dan 2 tetap.

Metode yang terakhir adalah `set_tgl()`. Fungsi metode ini adalah untuk menampilkan tanggal yang kita inginkan baik dalam bentuk standar ataupun dalam detik (timestamp). Cara kerja dari metode ini bergantung dari nilai `$this->mode` yang diperoleh dari metode `set_mode()`.

Metode `set_tgl()` memiliki dua parameter dan keduanya bersifat opsional. Parameter pertama merupakan format tampilan yang ingin digunakan apakah 0 = standar(dd-mm-yyyy) atau 1 = detik (timestamp). Parameter kedua adalah angka yang kita inginkan berapa hari tanggal harus ditambah atau dikurangi.

Pada metode ini pertama-tama kita mengecek isi dari parameter dua apabila lebih dari 31 maka keluar dari fungsi. Selanjutnya kita menentukan operasi yang dijalankan apakah itu pengurangan, penambahan atau tetap. Dengan mengecek isi dari `$this->mode`.

Untuk kondisi `$this->mode` sama dengan 0 maka kita lakukan operasi pengurangan tanggal. Pertama kita kurangi tanggal sekarang dengan isi dari parameter dua yaitu variabel `$angka`,

```
$tgl = $this->date - $angka; // tanggal dikurangi
```

Operasi pengurangan diatas sudah cukup untuk menghasilkan tanggal sebelum tanggal sekarang. Namun bagaimana jika hasil dari operasi diatas kurang dari atau sama dengan nol. Bisa-bisa pengunjung tertawa melihat tanggal pada website kita misalnya -5 Februari. Jadi kita harus menghandel kondisi dimana kemungkinan tanggal 0 atau minus.

```
if ($tgl <= 0)
```

Secara otomatis jika kondisi diatas terpenuhi maka bulan yang akan ditampilkan adalah bulan sebelumnya. Oleh karena itu kita mengurangi bulan dengan satu seperti berikut

```
$bulan = $this->bulan - 1; // kurangi bulan dengan 1
```

Tentunya jumlah hari antara bulan yang satu dengan yang lainnya ada yang tidak sama. Jadi kita juga harus memperhitungkan kondisi tersebut. Untuk bulan yang jumlah tanggalnya 30 kita gunakan statemen seperti berikut

```
if ($this->jml_hari == 30) // bulan yang tanggalnya sampai 30
 $tgl = (31 + $this->date) - $angka; // tanggal pada bulan yang tanggalnya 31
```

Misalnya sekarang bulan September tanggal 3 dan kita ingin menampilkan tanggal 5 hari sebelumnya hasilnya adalah 29 Agustus sehingga untuk mendapatkan tanggal tersebut logikanya adalah seperti berikut

Tanggal = ( 31 + 3 ) - 5 >>> HASIL = 29

Hal yang sama berlaku untuk bulan-bulan yang lain, kecuali bulan Januari karena bulan sebelumnya 12 (Desember) dan tahunnya berkurang satu. Juga pada bulan Maret karena bulan sebelumnya adalah Februari maka jumlah tanggalnya 28 untuk non kabisat dan 29 untuk kabisat.

Langkah yang terakhir dari metode ini adalah menampilkan output tanggal sesuai dengan isi dari parameter pertama. Jika 0 maka formatnya adalah tanggal biasa yaitu dd – mm – yyyy sedangkan jika 1 maka formatnya adalah dalam detik(timestamp).

```
$this->hasil = mktime(0, 0, 0, $bulan, $tgl, $thn); // detik
```

Fungsi yang kita gunakan untuk merubah tanggal menjadi detik adalah mktime(). Tiga parameter pertama adalah untuk jam, menit dan detik yang kita isikan 0. Selanjutnya adalah bulan, tanggal dan tahun.

Akhir dari file yang ada pada direktori **Iklan\inc**.

---

Selanjutnya file-file yang akan kita buat adalah file pada direktori **iklan\template**. File-file pada direktori ini umumnya berisi konfigurasi tampilan seperti skin dan variabel-variabel yang digunakan pada halaman lainnya.

File pertama yang akan kita buat adalah *index.php*. File ini hanya bertugas untuk melakukan redirect halaman jika user mencoba masuk ke halaman template. Pada PHP DESIGNER 2005 klik menu **File > New > PHP** untuk membuat file PHP baru, kemudian ketik kode berikut.

Script dari file ***index.php***

```
<?php
/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk meredirect ke halaman utama
*/
// jika user mencoba masuk ke direktori template
// kembalikan ke halaman utama
header ("Location: ../index.php");
?>
```

Simpan pada direktori **iklan\template** dengan nama ***index.php***.

Kali ini kita akan membuat file yang menjadi dasar tampilan dari semua halaman dari website superIklan.localhost. File ini merupakan file template yang berisi kode-kode HTML sebagai output. Klik **File > New > PHP** pada PHP DESIGNER 2005, kemudian ketik kode berikut.

Script dari file ***skin\_utama.php***

```
<html>
<head>
<title>Super Iklan</title>
<style>
 body{font-family: verdana, arial, sans serif; font-size: 13px}
 td{font-family: verdana, arial, sans serif; font-size: 13px}
 table{border-collapse: collapse; border-color: #cccccc}
 a{color: #7cb500}
 a:visited{color: #7cb500}
 a:hover{font-weight: bold; text-decoration: none; font-size: 14px}
 a:hover.iklan{font-size: 12px}
```

```

.batas{margin-left: 0; margin-right: 0; margin-top: 0; margin-bottom: 0}
.putih{color: white; font-weight: bold; font-size: 14px}
.jdl_iklan{color: white; font-weight: bold; font-size: 13px}
.kecil{font-size: 12px}
</style>
</head>
<body class='batas'>

<!-- Tabel Untuk HeadLine website -->
<table border='0' cellpadding='0' width='100%'>
<tr bgcolor='#7cb500' height='75'>
<td align='center' class='putih'><h1>- SuperIklan.localhost -</h1></td>
</tr>
</table>

<!-- tabel utama -->
<table border='1' cellpadding='4' width='100%'>
<tr height='25'>
<td width='30%'>{SEKARANG}</td>
 |
```

Simpan pada direktori **iklan\template** dengan nama ***skin\_utama.php***. Jika anda buka pada browser output dari file ini ditunjukkan oleh gambar 4.25



Gambar 4.25: Output dari file *skin\_utama.php*

Berikutnya kita akan membuat file yang akan menyimpan variabel-variabel yang dipanggil dari semua halaman non member. Langsung saja buat file PHP baru pada PHP DESIGNER 2005, kemudian salin kode berikut

Script dari file ***var\_utama.php***

```
<?php
```

```

/*
** Nama File.....: var_utama.php
** Penulis.....: Rio Astamal
** Tanggal.....: 15-10-2005
** Penjelasan....: menyimpan variabel yang dipanggil
** semua halaman.
***** */

// cegah pengaksesan langsung dari browser
if (eregi('var_utama.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php'); // kembalikan ke halaman utama
 exit;
}

// panggil file class_halaman.php
include ('inc/class_halaman.php');
// panggil file konfig.php
include ('inc/konfig.php');

// buat variabel untuk tanggal sekarang
$tgl = "Hari ini: ".show_tgl(); // fungsi untuk menampilkan tanggal sekarang

// buat variabel untuk menampilkan menu
$menu = "Home :: \n"
 . "Member Login ::
 \n"
 . "Daftar ::
\n"
 . "News :: \n"
 . "Iklan\n" ;

// dapatkan data page dari URL
$page = $_GET['page'];
if ($page == '')
 $page = 0;

$page = filter_str($page);

/* variabel untuk link daftar berita pada kiri halaman utama */
***** */

konek_db(); // koneksi ke MySQL server
// panggil class halaman
$hal = new halaman; // buat objek halaman
$hal->set_tabel('tb_berita'); // tentukan tabel
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph($u_jbph); // judul berita per halaman (konfig.php)

// lakukan query
$hal->query_SQL(1); // lakukan query
$jml_berita = $hal->get_jml_data(); // dapatkan jumlah berita
$hal->get_jml_hal(); // dapatkan jumlah halaman

$record = $hal->get_record(); // dapatkan jumlah record
$hasil = $hal->query_SQL("SELECT * FROM tb_berita ORDER BY id_berita DESC LIMIT
$record, $pph");

// variabel untuk menampilkan cuplikan berita
$daftar_berita = "<table border='0' cellpadding='4'>\n"
 . "<tr bgcolor='#7cb500'>\n"
 . " <td class='putih'>Berita Terbaru</td>\n </tr> \n";

```

```

while ($data = mysql_fetch_array($hasil))
{
 $link = "$data[jdl_berita]";
 $daftar_berita .= "<tr><td class='kecil'>$link</td></tr>\n";
}

$hal->set_hal(); // tentukan nilai Last, Next, dan Before
$daftar_berita .= "</table>\n" . $hal->show_page("index.php?proses=view") . "\n";

mysql_close(); // tutup koneksi ke MySQL

/* variabel untuk link iklan pada sebelah kiri halaman utama */
*****iklan_sisi = "<table border='0' cellpadding='4' width='100%>\n"
. "<tr bgcolor='#7cb500'>\n"
. " <td class='putih'>Iklan</td>\n</tr>\n"
. "<tr>\n <td>\n"
. "<p>Klik link iklan yang anda inginkan</p>\n"
. "<p>Iklan hari ini</p>\n"
. "<p>Iklan kemarin</p>\n"
. "<p>Ingin pasang iklan?
"
. "Mohon register dulu.</p>\n"
. "</td>\n</tr>\n</table>\n";
*****// variabel untuk form pencarian iklan
$cari = "<form action='cari.php' method='get'>\n"
. "<table border='0' cellpadding='4' width='100%>\n"
. "<tr bgcolor='#7cb500'>\n"
. " <td class='putih'>Pencarian</td>\n"
. "<tr> \n<td>\n"
. "<p>Pilih kategori dan keyword lalu tekan CARI untuk melakukan pencarian.</p>\n"
. "Kategori: <select name='kat'>\n"
. "<option value='komputer'>Komputer</option>\n"
. "<option value='internet'>Internet</option>\n"
. "<option value='elektronik'>Elektronik</option>\n"
. "<option value='lowongan'>Lowongan</option>\n"
. "<option value='otomotif'>Otomotif</option>\n"
. "<option value='properti'>Properti</option>\n"
. "<option value='lainnya'>Lainnya</option>\n"
. "</select>\n"
. "

Keyword:
<input type='text' name='keyword'>\n"
. "<input type='submit' value='CARI'>\n"
. "</table>\n</form>\n";
*****// buat link kembali berguna jika ada error.
$kembali = "
<<< Kembali\n"

```

?&gt;

Simpan pada direktori **iklan\template** dengan nama **var\_utama.php**. Pada form pencarian kita menggunakan metode **get** bukan **post**, karena kita akan menampilkan menggunakan format per halaman. Jadi jika memakai metode post otomatis halaman berikut dan sebelumnya tidak dapat kita ketahui.

Selanjutnya kita akan membuat file yang menyimpan variabel untuk halaman member. Hampir sama dengan sebelumnya variabel-variabel yang kita buat diperlukan oleh semua halaman member. Untuk membuatnya klik **File > New > PHP**, lalu ketik kode berikut.

**Script dari file *member\_var.php***

---

```
<?php

/*
** Nama File.....: member_var.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan....: menyimpan variabel yang dipanggil
** semua halaman member
***** */

// cegah pengaksesan langsung dari browser
if (eregi('var_utama.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php'); // kembalikan ke halaman utama
 exit;
}

include ('../inc/fungsi.php'); // panggil fungsi.php

$tgl = "Hari ini: ".show_tgl(); // tanggal sekarang

$login = "<p>Anda login sebagai: $_SESSION[member]</p>\n"; // id member

$not_login = "<p>Anda belum login. Login
dulu.</p>\n";

$mem_menu = "Home &nbsp :: &nbsp \n"
. "Update Profil &nbsp :: &nbsp \n"
. "Pasang Iklan &nbsp :: &nbsp \n"
. "Email &nbsp :: &nbsp \n"
. "Log out\n";

// variabel yang berisi javascript untuk membatasi karakter iklan
// isi = nama textarea untuk iklan
// sisa = nama textbox untuk sisa karakter
$java = "<script language='javascript'>
 function batas_kar(form_ini)
 {
 var max = 500;
 panj = max - form_ini.isi.value.length;
 if (panj < 0)
 {
 form_ini.isi.value = form_ini.isi.value.substring(0, max);
 panj = 0;
 }
 form_ini.sisa.value = panj;
 }
</script>";

// variabel untuk menampilkan daftar iklan yang di-post member tersebut
koneksi_db(); // koneksi ke mySQL Server

$iklanku = "<table border='0' cellpadding='4'>\n"
. "<tr bgcolor='#7cb500'>\n"
. " <td class='putih'>IklanKu</td></tr>\n"
. "<tr><td><p>Klik masing-masing link iklan untuk
mengedit.</p></td></tr>\n";
// lakukan query
$hasil = mysql_query("SELECT * FROM tb_iklan WHERE
username='$_SESSION[member]'");
// cek baris
if (mysql_num_rows($hasil) == 0)
```

```

$iklanku .= "<tr><td><p>Anda belum memasang iklan. klik "
 . "pasang untuk memasang
iklan.</p></td></tr>\n";
else
{
// tampilkan hasil dengan looping
while ($data = mysql_fetch_array($hasil))
{
 $iklanku .= "<tr><td><p><a class='iklan' href='edit.php?kat=$data[kategori]&
 . "id=$data[id_iklan]'>$data[jdl_iklan]</p></td></tr>\n";
}
}

$iklanku .= "</table>\n";

// buat link kembali berguna jika ada error.
$kembali = "
<<< Kembali\n";
?>

```

Simpan pada direktori **iklan\template** dengan nama ***member\_var.php***.

### PENJELASAN SCRIPT

Yang perlu dijelaskan disini mungkin hanya isi dari variabel java saja. Isi dari variabel ini adalah sebuah javascript yang kita gunakan untuk membatasi jumlah karakter pada textarea. Pada script ini kita membuat variabel **max** yang merupakan jumlah karakter maximal yang dapat ditampung pada textarea.

Fungsi batas\_kar() pada script diatas memiliki satu parameter yaitu form tempat textarea dan textbox untuk sisa karakter. Nantinya untuk dapat menjalankan fungsi ini nama textarea dan textbox masing **isi** dan **sisa**.

File terakhir pada direktori ini adalah *admin\_var.php*. File ini bertugas menyimpan variabel-variabel yang digunakan pada halaman administrator. Buat file PHP baru pada PHP DESIGNER 2005 dengan mengklik menu **File > New > PHP**, lalu ketik kode berikut.

Script dari file ***admin\_var.php***

```

<?php

/*
** Nama File.....: admin_var.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk menyimpan variabel-variabel
** pada halaman control panel
***** */

// cegah pengaksesan langsung dari browser
if (eregi('admin_var.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php');
 exit;
}

include ('../inc/fungsi.php'); // panggil file fungsi.php

$tgl = "Hari ini: ".show_tgl(); // tanggal sekarang

// variabel untuk menu admin
$admin_menu =
<table border='0' cellpadding='4' width='100%'>

```

```

<tr bgcolor="#7cb500">
 <td class='putih'>Admin Menu</td></tr>
<tr><td>
 <p>Manaje Iklan</p>
 <p>Manaje Member</p>
 <p>Manaje Berita</p>
 <p>Email Member</p>
 <p>Logout</p>
</td></tr>
</table>\n";

// varibel untuk menampilkan pesan belum login
$not_login = "<p>Anda belum login. Login dulu</p>\n";

// variabel untuk menampilkan teks berjalan
$anim_teks = "
<marquee scrolldelay='50' scrollamount='2'><<< Control panel
superiklan.localhost
</marquee>\n";

// variabel untuk menampilkan link kembali
$kembali = "
<<< Kembali\n";

?>

```

Simpan pada direktori **iklan\template** dengan nama **admin\_var.php**. Saya rasa tidak ada yang baru pada script ini sehingga tidak perlu saya jelaskan.

Akhir dari file yang berada pada direktori **iklan\template**.

Selanjutnya kita akan mulai membangun control panelnya terlebih dahulu. File-file ini kita letakkan pada direktori **admin**. File pertama yang akan kita buat pada direktori ini adalah **index.php**. File ini berfungsi menyediakan form login dan halaman utama. Klik menu **File > New > PHP** kemudian ketik kode berikut.

Script dari file **index.php**

```

<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk menampilkan menu atau form
** login untuk admin
***** */

session_start();

// panggil file-file yang diperlukan
include ('../inc/class_skin.php');
include ('../inc/class_waktu.php');
include ('../template/admin_var.php');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'login';

$proses = filter_str($proses); // filter string

konek_db(); // koneksi ke database

```

```

// handel setiap proses dengan switch dan case
switch ($proses)
{
 //*****
 case 'login':
 //*****

 // cek apakah admin sudah login atau belum
 if (!cek_session('admin'))
 {
 $judul = "<h2>superiklan.localhost - login</h2>\n";
 $admin = "
<form action='index.php?proses=proses_login' method='post'>
<table border='0' cellpadding='4'>
<tr bgcolor='#7cb500' align='center'>
<td colspan='2' class='putih'>Admin Login</td></tr>
<tr><td>Username: </td><td><input type='text' name='username'></td></tr>
<tr><td>Password: </td><td><input type='password' name='password'></td></tr>
<tr><td></td><td><input type='submit' value='L O G I N'></td></tr>
<tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>
</table>
</form>\n\n";

 // kosongkan nilai variabel $admin_menu
 $admin_menu = '';
 }
 else
 {
 $admin = "
<h2>Selamat datang di control panel - superiklan.localhost</h2>
<p>Silahkan pilih menu disamping untuk memanage website superiklan.
localhost</p>\n";

 // panggil class waktu
 $waktu = new waktu;
 $waktu->set_date();
 $waktu->set_mode(0); // mode pengurangan
 $waktu_itu = $waktu->set_tgl(1, $lama_iklan); // $lama_iklan di konfig.php

 // hapus iklan yang dipost lebih dari $lama_iklan
 $hasil = mysql_query("DELETE FROM tb_iklan WHERE timestamp<$waktu_itu");
 // cek status
 if (!$hasil)
 $admin .= "<p>Error: gagal menghapus iklan pada
database.
\n$kembali</p>\n";
 }

 break;
 //*****
}

//*****
case 'proses_login':
//*****

$username = filter_str($_POST['username']);
$password = filter_str($_POST['password']);

$password = balik_md5($password); // enkripsi password

// kosongkan variabel $admin_menu
$admin_menu = '';

```

```

// cek kecocokan data dengan fungsi login
if (!login('admin', $username, $password))
 $admin = "<p>Username atau password salah.
\n$kembali</p>" ;
else
{
 $admin = "<p>Login berhasil. Klik disini"
 . " untuk masuk admin area</p>\n" ;

 // buatkan session karena berhasil login
 $_SESSION['admin'] = $username;
}

break;
***** */

***** */
case 'logout':
***** */

if (!logout('admin'))
{
 $admin_menu = ''; // kosongkan menu
 $admin = "<p>Tidak bisa logout. Login dulu.</p>\n";
}
else
{
 $admin_menu = ''; // kosongkan menu
 $admin = "<p>Anda telah logout dari sistem. Login"
 . " kembali.</p>\n";
}
break;
***** */

***** */
case '__add_admin_to_database__':
***** */

/*
case ini berfungsi untuk memasukkan account administrator ke database
ini dikarenakan fungsi yang kita gunakan login adalah balik_md5()
dan untuk menghasilkan string chiper ini hanya bisa dilakukan lewat
script PHP bukan pada MySQL

untuk memanggil fungsi ini harus diketikkan langsung pada address bar
index.php?proses=__add_admin_to_database__ lalu ENTER

untuk mencegah eksploitasi sistem, kita tidak menyediakan form untuk
menambahkan account admin ke database melainkan langsung
melakukan query
*/
// kosongkan nilai $admin_menu
$admin_menu = '';

// tentukan username dan password yang diinginkan
$username = 'admin';
$password = balik_md5('__pas123__');

// lakukan query INSERT untuk memasukkan account ke database
$hasil = mysql_query("INSERT INTO admin VALUES('$username', '$password')");
if (!$hasil)
 $admin = "Error: Gagal memasukkan ke database. Mungkin account "
 . "sudah dimasukkan.
\n$kembali";

```

```

else
 $admin = "Account untuk administrator berhasil dimasukkan ke database.

 .Login\n";

break;
***** */

} // akhir dari switch

mysql_close(); // tutup koneksi

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $admin);
$skin->ganti_tag('{MENU}', $anim_teks);
$skin->ganti_tag('{SISI1}', $admin_menu);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan\admin** dengan nama **index.php**. Untuk membukanya arahkan browser anda ke <http://localhost/www/webmaster/Bab4/hari-21-22-23/iklan/admin/>. Output dari script diatas dapat akan terlihat seperti gambar 4.26.



Gambar 4.26: Halaman utama admin area

#### PENJELASAN SCRIPT

Sebagian besar dari script ini sudah sering kali kita gunakan. Namun ada bagian yang baru yaitu penggunaan class waktu untuk menghapus iklan yang sudah 1 minggu. Jika admin berhasil login secara otomatis query penghapusan akan dilakukan. Lihat cuplikan script dibawah ini.

```

$waktu = new waktu;
$waktu->set_date();
$waktu->set_mode(0); // mode pengurangan
$waktu_itu = $waktu->set_tgl(1, $lama_iklan);

```

Pertama kita membuat objek waktu yang kita simpan pada variabel \$waktu. Kemudian kita memanggil metode set\_date() untuk mendapatkan tanggal, bulan dan sebagainya. Ketiga, kita menentukan modenya pengurangan karena kita akan menghapus iklan yang dipost pada hari-hari sebelumnya menggunakan set\_mode(0).

Kemudian kita membuat variabel \$waktu\_itu yang berisi detik \$lama\_hari yang lalu. Yang isinya ada pada file konfig.php defaultnya adalah 7 hari. Mengapa kita menggunakan detik? Mudah saja itu dikarenakan jika kita menggunakan tanggal biasa kita tidak bisa melakukan perbandingan ataupun kalkulasi ketika melakukan query SQL.

```
$hasil = mysql_query("DELETE FROM tb_iklan WHERE timestamp < $waktu_itu");
```

Query diatas akan menghapus iklan yang dipost lebih dari 7 hari yang lalu.

File berikutnya yang akan kita buat adalah file yang bertugas memanage iklan yang dipost member. Disini admin dapat memilih kategori iklan yang ingin ditampilkan. Setelah itu muncul daftar judul iklan, pengirim dan tanggal posting. Untuk membuat file ini buat file PHP baru pada PHP DESIGNER 2005, ketik listing kode dibawah ini.

#### Script dari file *iklan.php*

---

```
<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk memanage iklan
*/
session_start();

// panggil file-file yang diperlukan
include ('../inc/class_skin.php');
include ('../inc/class_halaman.php');
include ('../inc/konfig.php');
include ('../template/admin_var.php');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

$proses = filter_str($proses); // filter string

koneksi_db(); // koneksi ke database

// cek apakah admin sudah login atau belum
if (!cek_session('admin'))
 $iklan = $not_login; // $not_login ada di admin_var.php
else
{

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*
 case 'view':
 */
}
```

```

$judul = "<h2>Pilih Kategori Iklan</h2>\n";
$iklan = "";

Pilih kategori iklan yang ingin anda edit.</p>

<p>Komputer</p>
<p>Internet</p>
<p>Elektronik</p>
<p>Lowongan</p>
<p>Otomotif</p>
<p>Properti</p>
<p>Lainnya</p>


```

*****/
case 'edit':
*****/

// dapatkan data dari URL
$kat = filter_str($_GET['kat']);
$page = filter_str($_GET['page']);

if ($page == '')
    $page = 0;

$judul = "<h2>Daftar iklan untuk kategori: $kat</h2>\n";

// panggil class halaman
$hal = new halaman; // buat objek halaman
$hal->set_tabel('tb_iklan'); // pilih tabel tb_iklan
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph($a_iph); // iklan per halaman (konfig.php)

// lakukan query
$hal->query_SQL(2, 'kategori', $kat); // lakukan query
$jml_iklan = $hal->get_jml_data(); // dapatkan jumlah berita
$hal->get_jml_hal(); // dapatkan jumlah halaman

// cek jumlah iklan
if ($jml_iklan == 0)
    $iklan = "<p>Tidak ada iklan untuk kategori: $kat<br>$kembali</p>\n";
else
{
    $record = $hal->get_record(); // dapatkan jumlah record
    $hasil = $hal->query_SQL(3, 'kategori', $kat, 'id_iklan');

    // buat tabel dan form
    $iklan = "


Jumlah iklan untuk kategori komputer: $jml_iklan</p>
<form action='iklan.php?proses=hapus' method='post'>
    <table border='0' cellpadding='4' width='100%'>
        <tr bgcolor='#7cb500' class='putih'>
            <td>Judul Iklan</td><td>Posted By</td><td>Posted On</td>
            <td align='center'>Hapus?</td></tr>\n";
}

// lakukan looping untuk menampilkan iklan
while ($data = mysql_fetch_array($hasil))
{
    $checkbox = "<input type='checkbox' name='hapus[]' value='";
    $checkbox .= $data['id_iklan'];
    $checkbox .= "'>";
    $iklan .= "


```


```

```

<tr>
<td>$data[jdl_iklan]</td><td>$data[username]</td><td>$data[tgl_post]</td>
<td align='center'>$checkbox</td>
</tr>\n";
}

$hal->set_hal(); // tentukan nilai next, before dan last

// tutup tabel
$iklan .= "
<tr bgcolor='#7cb500' height='20'><td colspan='3'></td>
<td><input type='submit' value='HAPUS'></td></tr>
</table>\n</form>\n"

// tampilkan link nomor halaman
.$hal->show_page("iklan.php?proses=edit&kat=$kat");
}

break;
***** */

***** */
case 'hapus':
***** */

$hapus = $_POST['hapus']; // hapus bertipe array
if ($hapus == '')
 $iklan = "<p>Error: Anda belum memilih item.
\n$kembali</p>\n";
else
{
 // gunakan looping untuk menghapus setiap item
 foreach ($hapus as $sampah)
 {
 $hasil = mysql_query("DELETE FROM tb_iklan WHERE id_iklan=$sampah");
 // cek status
 if (!$hasil)
 $iklan = "Error: gagal menghapus iklan dengan id: $sampah.
$kembali\n";
 else
 $iklan = "Iklan dengan id: $sampah berhasil dihapus.
$kembali\n";
 }
}

break;
***** */

} // akhir dari switch
} // akhir dari else

mysql_close(); // tutup koneksi

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $iklan);
$skin->ganti_tag('{MENU}', $anim_teks);
$skin->ganti_tag('{SISI1}', $admin_menu);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan\admin** dengan nama **iklan.php**. Jalankan pada browser anda output dari file ini akan terlihat seperti gambar 4.27

Judul Iklan	Posted By	Posted On	Hapus?
Dijual Komputer PENTIUM 4 HT MURAH!!	webmaster	18-10-2005, 18:02	<input type="checkbox"/>
DIJUAL Laptop Toshiba Pentium 3!!	webmaster	18-10-2005, 17:32	<input type="checkbox"/>

Gambar 4.27: Output file iklan.php pada admin area

### PENJELASAN SCRIPT

File ini memiliki beberapa case yaitu *view*, *edit*, dan *hapus*. Case *view* hanya berfungsi menampilkan link semua kategori iklan. Dan untuk case *edit*, case ini berfungsi untuk menampilkan daftar iklan yang telah dipost pada kategori yang dipilih admin.

Anda dapat menentukan berapa banyak daftar iklan yang ditampilkan dengan per halaman dengan mengedit variabel `$_iph` pada file `konfig.php`. Untuk menampilkan daftar iklan serta link navigasinya kita gunakan class halaman.

Seperti yang sudah pernah kita kerjakan pada proyek sebelumnya. Pertama-tama kita memanggil class halaman dengan keyword *new*. Kemudian kita tentukan tabel yang kita gunakan karena akan mendapatkan daftar iklan maka pada metode `set_tabel()` kita isikan `tb_iklan` seperti berikut.

```
$hal->set_tabel('tb_iklan'); // pilih tabel tb_iklan
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph($a_iph); // iklan per halaman (konfig.php)
```

Kedua kita tentukan variabel data halaman yang kita handel yaitu `$page`. Ketiga kita tentukan berapa daftar iklan yang kita tampilkan per halaman. Pada kode diatas kita menggunakan isi dari variabel `$a_iph` yang defaultnya adalah 2. Setelah itu kita lakukan query menggunakan metode `query_SQL()` untuk menentukan jumlah halaman dan banyaknya data yang ditampilkan.

```
$hal->query_SQL(2, 'kategori', $kat); // lakukan query
$jml_iklan = $hal->get_jml_data(); // dapatkan jumlah berita
$hal->get_jml_hal(); // dapatkan jumlah halaman
```

metode `query_SQL(2, 'kategori', $kat)` sama dengan query berikutnya

```
SELECT * FROM tb_iklan WHERE kategori='$kat';
```

Isi dari variabel \$jml\_iklan merupakan jumlah data/baris yang dikembalikan mysql dari query pada metode query\_SQL() sebelumnya. Metode selanjutnya yaitu get\_jml\_hal() digunakan untuk mendapatkan jumlah halaman. Kemudian kita lakukan pengecekan kondisi dari jumlah data yang dikembalikan jika nol tidak usah melakukan query lagi, langsung saja tampilkan pesan.

Sebaliknya jika jumlah data tidak sama dengan nol maka lakukan query ulang. Namun sebelumnya panggil dulu metode get\_record() agar hasil query valid.

```
$record = $hal->get_record(); // dapatkan jumlah record
$hasil = $hal->query_SQL(3, 'kategori', $kat, 'id_iklan');
```

Kemudian kita buat header tabel dan form untuk menampilkan judul, pengirim, tanggal posting dan hapus. Untuk menampilkan daftar iklan seperti biasa kita gunakan looping while.

Setelah looping kita kan mencetak nomor link halaman. Untuk itu kita panggil metode set\_hal() untuk memberi nilai pada before, next dan last. Dan yang terakhir adalah mencetak link halaman dengan metode show\_page(view.php?proses=view). Isi parameter pada metode show\_page() harus sesuai dengan halaman yang sedang dibuka.

Case yang terakhir yaitu *hapus*. Tidak ada yang baru pada case ini yaitu hanya mengambil data yang berbentuk array. Kemudian kita gunakan looping foreach untuk menghapus setiap item.

Berikutnya file yang akan kita buat bertugas untuk menghapus member yang telah bergabung. Untuk membuatnya klik **File > New > PHP** pada PHP DESIGNER 2005 kemudian ketik kode berikut.

Script dari file **member.php**

---

```
<?php

/*
** Nama File.....: member.php
** Penulis....: Rio Astamal
** Tanggal....: 17-10-2005
** Penjelasan....: Untuk memanage member
***** */

session_start();

// panggil file-file yang diperlukan
include ('../inc/class_skin.php');
include ('../inc/class_halaman.php');
include ('../inc/konfig.php');
include ('../template/admin_var.php');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
$proses = 'view';

$proses = filter_str($proses); // filter string

konek_db(); // koneksi ke database

// cek apakah admin sudah login atau belum
if (!cek_session('admin'))
$member = $not_login;
```

```

else
{
 // handel setiap proses dengan switch dan case
 switch ($proses)
 {
 /*****
 case 'view':
 ****/

 // dapatkan data dari URL
 $page = filter_str($_GET['page']);
 if ($page == '')
 $page = 0;

 $judul = "<h2>Daftar Member superiklan.localhost</h2>\n";
 // panggil class halaman
 $hal = new halaman; // buat objek halaman
 $hal->set_tabel('member'); // pilih tabel tb_iklan
 $hal->set_page($page); // data halaman yang dihandel
 $pph = $hal->set_pph($a_mph); // member per halaman

 // lakukan query
 $hal->query_SQL(1); // lakukan query
 $jml_member = $hal->get_jml_data(); // dapatkan jumlah berita
 $hal->get_jml_hal(); // dapatkan jumlah halaman

 // cek jumlah member
 if ($jml_member == 0)
 $iklan = "<p>Tidak ada member yang register.
</p>$kembali";
 else
 {
 $record = $hal->get_record(); // dapatkan jumlah record
 $hasil = $hal->query_SQL("SELECT * FROM member LIMIT $record, $pph");

 // buat tabel dan form
 $member = "
<p>Total member yang bergabung: $jml_member orang.</p>
<form action='member.php?proses=hapus' method='post'>
<table border='0' cellpadding='4' width='130%'>
<tr bgcolor='#7cb500' class='putih'>
<td>Username</td><td>Nama</td><td>Email</td><td>Kota</td>
<td>Telpon</td><td align='center'>Hapus?</td></tr>\n";

 // gunakan looping untuk menampilkan member
 while ($data = mysql_fetch_array($hasil))
 {
 $checkbox = "<input type='checkbox' name='hapus[]' value='".$data[username]."'>";
 $member .= "
<tr>
<td>$data[0]</td><td>$data[2]</td><td>$data[3]</td><td>$data[5]</td>
<td>$data[6]</td><td align='center'>$checkbox</td>
</tr>\n";
 }

 $hal->set_hal(); // beri nilai untuk next, before dan last
 // tutup tabel
 $member .= "
<tr bgcolor='#7cb500' height='20'><td colspan='5'></td>
<td><input type='submit' value='H A P U S'></td></tr>

```

```

</table> \n</form>\n\n"

// tampilkan link nomor halaman
.$hal->show_page("member.php?proses=view");
}

break;
***** */

***** */
case 'hapus':
***** */

$hapus = $_POST['hapus']; // hapus bertipe array

if ($hapus == '')
 $member = "<p>Error: Anda belum memilih item.
\n$kembali</p>\n";
else
{
 // gunakan looping untuk menghapus setiap item
 foreach ($hapus as $sampah)
 {
 $hasil = mysql_query("DELETE FROM member WHERE username= '$sampah' ");
 // cek status
 if (!$hasil)
 $member = "Error: Username $sampah tidak dapat dihapus.
$kembali\n";
 else
 $member = "Username $sampah berhasil dihapus.
$kembali";
 }
}

break;
***** */

} // akhir dari switch
} // akhir dari else

mysql_close(); // tutup koneksi

$skin = new skin; // buat objek skin
$skin->ganti_skin('..../template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $member);
$skin->ganti_tag('{MENU}', $anim_teks);
$skin->ganti_tag('{SISI1}', $admin_menu);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan\admin** dengan nama **member.php**. Output dari file ini ditunjukkan oleh gambar 4.28. Tidak ada yang baru pada kode ini, dan strukturnya masih sama dengan kode sebelumnya.

Gambar 4.28: Output dari file member.php

Lanjut...!, Selanjutnya kita akan membuat file yang bertugas menambah, mengedit dan menghapus berita. Sama dengan sebelumnya struktur dari file ini juga tidak banyak berubah. Pada PHP DESIGNER 2005 klik **File > New > PHP**, lalu klik kode berikut.

#### Script dari file **berita.php**

```
<?php

/*
** Nama File.....: berita.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk memanage berita
***** */

session_start();

// panggil file-file yang diperlukan
include ('../inc/class_skin.php');
include ('../inc/class_halaman.php');
include ('../inc/konfig.php');
include ('../template/admin_var.php');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

$proses = filter_str($proses); // filter string

koneksi_db(); // koneksi ke database

// cek apakah admin sudah login atau belum
if (!cek_session('admin'))
```

```

$berita = $not_login;
else
{
 // handel setiap proses dengan switch dan case
 switch ($proses)
 {
 /*****
 case 'view':
 *****/
 // dapatkan data dari URL
 $page = filter_str($_GET['page']);
 if ($page == '')
 $page = 0;

 $judul = "<h2>Daftar Berita</h2>\n";

 // panggil class halaman
 $hal = new halaman; // buat objek halaman
 $hal->set_tabel('tb_berita'); // pilih tabel tb_iklan
 $hal->set_page($page); // data halaman yang dihandel
 $pph = $hal->set_pph($a_bph); // berita per halaman (konfig.php)

 // lakukan query
 $hal->query_SQL(1); // lakukan query
 $jml_berita = $hal->get_jml_data(); // dapatkan jumlah berita
 $hal->get_jml_hal(); // dapatkan jumlah halaman

 // cek jumlah member
 if ($jml_berita == 0)
 $berita = "<p>Tidak ada member yang register.
</p>$kembali\n";
 else
 {
 $record = $hal->get_record(); // dapatkan jumlah record
 $hasil = $hal->query_SQL("SELECT * FROM tb_berita ORDER BY id_berita DESC
LIMIT $record, $pph");

 // buat tabel dan form
 $berita =
 <p>Total terdapat: $jml_berita berita.</p>
 <p>Tambah Berita</p>
 <form action='berita.php?proses=hapus' method='post'>
 <table border='0' cellpadding='4' width='100%'>
 <tr bgcolor='#7cb500' class='putih'>
 <td>Judul Berita</td><td>Tanggal Post</td><td>Edit?</td>
 <td align='center'>Hapus?</td>
 </tr>\n";
 // tampilkan berita dengan looping
 while ($data = mysql_fetch_array($hasil))
 {
 // link untuk edit berita
 $link = "Edit";
 $checkbox = "<input type='checkbox' name='hapus[]' value='$data[0]'>";

 $berita .= "
 <tr>
 <td>$data[1]</td><td>$data[3]</td><td>$link</td>
 <td align='center'>$checkbox</td>
 </tr>\n";
 }
 }
 }
}

```

```

$hal->set_hal(); // beri nilai untuk Last, Before, Next

// tutup tabel
$berita .= "
<tr bgcolor='#7cb500' height='20'><td colspan='3'></td>
<td><input type='submit' value='H A P U S'></td></tr>
</table>\n</form>\n\n"
.$hal->show_page("berita.php?proses=view");
}

break;
***** */

***** */
case 'hapus':
***** */

$hapus = $_POST['hapus']; // hapus bertipe array

if ($hapus == '')
 $berita = "<p>Error: Anda belum memilih item.
\n$kembali</p>\n";
else
{
 // gunakan looping untuk menghapus setiap item yang dicek
 foreach ($hapus as $sampah)
 {
 $hasil = mysql_query("DELETE FROM tb_berita WHERE id_berita=$sampah");
 // cek status
 if (!$hasil)
 $berita = "Error: Berita dengan id: $sampah tidak dapat
dihapus.
$kembali\n";
 else
 $berita = "Berita dengan id: $sampah berhasil dihapus.
$kembali\n";
 }
}

break;
***** */

***** */
case 'edit':
***** */

$id = filter_str($_GET['id']);

$judul = "<h2>Edit Berita</h2>\n";

// lakukan query
$hasil = mysql_query("SELECT * FROM tb_berita WHERE id_berita=$id");
$data = mysql_fetch_array($hasil); // pecah menjadi array

$jdl = stripslashes($data['jdl_berita']); // judul iklan
// gunakan htmlspecialchars() agar tag-tag HTML tidak diproses
$isi = htmlspecialchars(stripslashes($data['isi_berita'])); // isi berita
$tgl = stripslashes($data['tgl_berita']); // tanggal

// buat form dan tabel
$berita =
<p>Silahkan edit berita lalu tekan EDIT untuk memasukkan perubahan.</p>
<form action='berita.php?proses=proses_edit' method='post'>
<table border='0' cellpadding='4'>
<tr bgcolor='#7cb500' align='center'>
<td class='putih' colspan='2'>Form Edit Berita</td></tr>
<tr>

```

```

<td>Judul Berita: </td>
<td><input type='text' name='jdl' size='50' value='$jdl'></td></tr>
<tr>
 <td>Isi Berita:
 <td><textarea name='isi' rows='10' cols='50'$isi</textarea></td></tr>
 <tr>
 <td>Tanggal: </td>
 <td><input type='text' name='tgl' value='$tgl'></td></tr>
 <tr>
 <td></td><td><input type='submit' value='E D I T'></td></tr>
 <tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>
</table>
<input type='hidden' name='id' value='$id'>
</form>\n\n";
break;
/************/

/************/
case 'proses_edit':
/************/

// ambil data yang di-post
$id = $_POST['id'];
$jdl = $_POST['jdl'];
$isi = $_POST['isi'];
$tgl = $_POST['tgl'];

// cek apakah masih ada field yang kosong
if (!cek_field($_POST))
 $berita = "<p>Error: Masih ada field yang kosong.
\n$kembali</p>\n";
else
{
 // lakukan query untuk mengupdate berita
 $hasil = mysql_query("UPDATE tb_berita SET jdl_berita='$jdl',
 isi_berita='$isi', tgl_berita='$tgl' WHERE id_berita=$id");
 // cek status
 if (!$hasil)
 $berita = "<p>Error: Gagal mengupdate berita.
\n$kembali</p>\n";
 else
 $berita = "<p>Berita berhasil diupdate.
\n$kembali</p>\n";
}
break;
/************/

/************/
case 'tambah':
/************/

$judul = "<h2>Tambah Berita</h2>\n";
$tgl = date('d-m-Y, H:i'); // tanggal sekarang

// buat form dan tabel
$berita =
<p>Isi semua field dibawah ini untuk menambah berita.</p>
<form action='berita.php?proses=proses_tambah' method='post'>
 <table border='0' cellpadding='4'>
 <tr bgcolor='#7cb500' align='center'>
 <td colspan='2' class='putih'>Form Tambah Berita</td></tr>
 <tr>
 <td>Judul berita: </td>
 <td><input type='text' name='jdl' size='50'></td></tr>
 <tr>

```

```

<td>Isi Berita: </td>
<td><textarea name='isi' rows='10' cols='50'></textarea></td></tr>
<tr>
<td>Tanggal: </td>
<td><input type='text' name='tgl' value='$tgl'></td></tr>
<tr>
<td></td><td><input type='submit' value='TAMBAH'></td></tr>
<tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>
</table>
</form>\n\n';

break;
***** */

***** */
case 'proses_tambah':
***** */

// ambil data yang dipost
$jdl = addslashes($_POST['jdl']); // tambahkan addslashes()
$isi = addslashes($_POST['isi']);
$tgl = addslashes($_POST['tgl']);

// cek apakah masih ada field yang kosong
if (!cek_field($_POST))
 $berita = "<p>Error: Masih ada field yang kosong.
\n$kembali</p>\n";
else
{
 // lakukan query SQL untuk memasukkan ke database
 $hasil = mysql_query("INSERT INTO tb_berita VALUES (0, '$jdl', '$isi',
'$tgl')");
 // cek status
 if (!$hasil)
 $berita = "<p>Error: Gagal memasukkan berita
kedatabase.
\n$kembali</p>\n";
 else
 $berita = "<p>Berita berhasil dimasukkan ke database.
\n$kembali</p>\n";
}

break;
***** */

} // akhir dari switch

} // akhir dari else

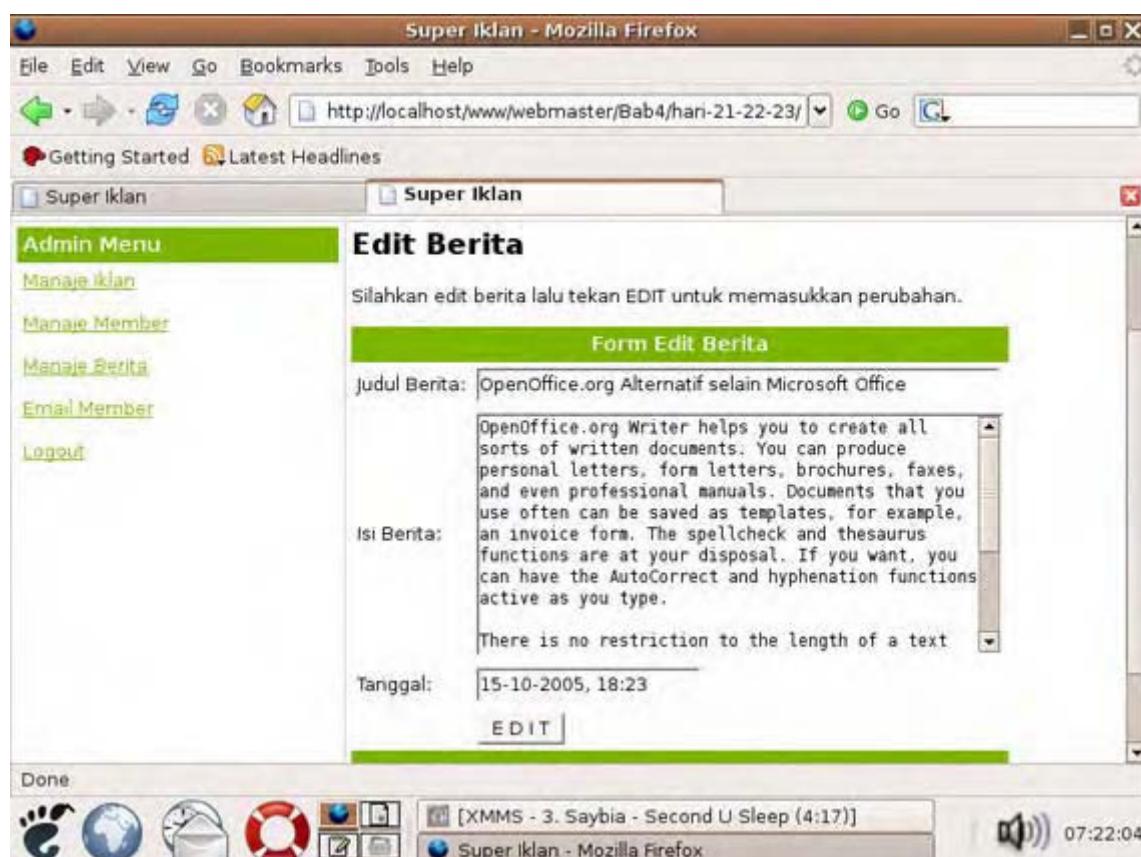
mysql_close(); // tutup koneksi

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $berita);
$skin->ganti_tag('{MENU}', $anim_teks);
$skin->ganti_tag('{SISI1}', $admin_menu);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan\admin** dengan nama **berita.php**. Gambar 4.29 menunjukkan output dari file ini saat mengedit berita.



Gambar 4.29: Tampilan file berita.php saat menambah daftar berita

File yang terakhir untuk direkori ini adalah *email.php*. File ini berfungsi untuk mengirim email ke semua member superiklan.localhost. Dengan ini admin dapat memberitahukan pemberitahuan atau promosi dan lain sebagainya ke seluruh member. Buat file baru pada PHP DESIGNER 2005 klik menu **File > New > PHP**, ketik listing kode berikut.

#### Script dari file *email.php*

```
<?php

/*
** Nama File.....: email.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk mengirim email ke semua member
** superiklan.localhost
*/

session_start();

// panggil file-file yang diperlukan
include ('../inc/class_skin.php');
include ('../template/admin_var.php');

// dapatkan data dari URL
$proses = $_GET['proses'];
if ($proses == '')
$proses = 'form';

$proses = filter_str($proses); // filter string

koneksi_db(); // koneksi ke database

// handel setiap data dengan switch dan case
```

```

switch ($proses)
{
 /*****
 case 'form':
 *****/
 $judul = "<h2>Form Send Mail</h2>";

 // buat tabel dan form
 $send_mail = "
<p>Silahkan isikan subject dan isi email. Tekan KIRIM untuk mulai mengirim
email</p>
<form action='email.php?proses=proses_form' method='post'>
<table border='0' cellpadding='4'>
 <tr bgcolor='#7cb500' align='center'>
 <td class='putih' colspan='2'>Form Send Mail</td></tr>
 <tr><td>Subject: </td>
 <td><input type='text' name='subject' size='60'></td></tr>
 <tr><td>Isi/body: </td>
 <td><textarea name='isi' rows='10' cols='60'></textarea></td></tr>
 <tr><td></td><td><input type='submit' value='K I R I M'></td></tr>
 <tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>
</table>
</form>\n\n";
}

break;
*****/
*****/
case 'proses_form':
*****/

// ambil data yang di-post
$subject = $_POST['subject'];
$isi = $_POST['isi'];
$from = "From: noreply@localhost"; // alamat email anda

// cek field
if (!cek_field($_POST))
 $send_mail = "<p>Error: Masih ada field yang kosong.
\n</p>\n";
else
{
 // lakukan query untuk mendaftar alamat email member
 $hasil = mysql_query("SELECT email FROM member");
 // lakukan looping untuk mengirim ke semua alamat
 while ($data = mysql_fetch_array($hasil))
 {
 $to = $data[0]; // alamat email masing-masing member

 // jika anda tidak memiliki program mail server atau anda belum di server
 // sebenarnya beri komentar pada fungsi mail berikut
 mail ($to, $subject, $isi, $from);
 }

 $send_mail = "<p>Email telah terkirim.</p>";
}

break;
*****/
}
 // akhir dari switch
}

```

```

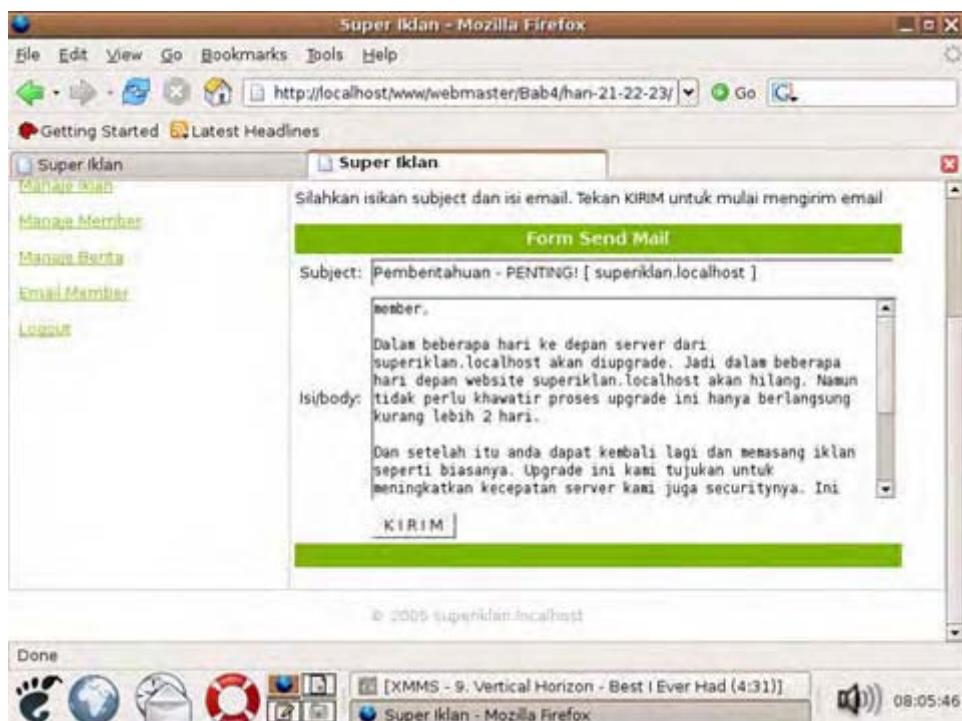
mysql_close(); // tutup koneksi

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $send_mail);
$skin->ganti_tag('{MENU}', $anim_teks);
$skin->ganti_tag('{SISI1}', $admin_menu);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan\admin** dengan nama ***email.php***. Output dari file ini ditunjukkan oleh gambar 4.30.



Gambar 4.30: Form untuk mengirim email ke seluruh member

Akhir dari file yang berada pada direktori **iklan\admin**.

Untuk selanjutnya kita akan bekerja pada direktori paling atas yaitu **iklan**. Pada direktori inilah kita tempatkan file yang menampilkan daftar berita, daftar iklan, form registrasi dan sebagainya. Seperti biasa kita akan membuat file utama untuk setiap direktori yaitu ***index.php***.

File ini berfungsi untuk menampilkan halaman utama ketika user mengunjungi. Dari halaman ini user juga dapat langsung memilih daftar iklan untuk hari ini atau kemarin. Juga user dapat melakukan pencarian iklan pada halaman ini karena pada sisi kiri halaman kita sediakan form pencarian iklan. Klik menu **File > New > PHP** pada PHP DESIGNER 2005.

Script dari file ***index.php***

```

<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 15-10-2005

```

```

** Penjelasan....: menampilkan halaman utama dari ***
** website superiklan.localhost ***

// panggil file-file yang diperlukan
include ('inc/class_skin.php');
include ('inc/fungsi.php');
include ('template/var_utama.php');

$judul = "<h2>Selamat Datang di superiklan.localhost</h2>\n";
$utama = "<p>Website superiklan.localhost merupakan website penyedia layanan
iklan \n"
 . "yang memberikan layanan terbaik bagi membernya. Kami sama sekali tidak
memungut \n"
 . "sepeserpun dari anda dalam pemasangan iklan. Artinya anda dapat
memasang iklan di website \n"
 . "ini secara gratis!. Kami juga menyediakan sarana iklan lewat email.
Jadi anda dapat \n"
 . "mengirim email ke seluruh member. Namun untuk email anda hanya dapat
mengirimnya satu \n"
 . "kali dalam seminggu. Ini untuk menghindari terjadinya SPAM.</p>\n"
 . "<p>Anda dapat memilih kategori yang menurut anda cocok dengan produk
\n"
 . "yang ingin anda pasarkan. Kategori-kategori tersebut antara lain \n"
 . "\nInternet\nKomputer\nElektronikLowongan
\n"
 . "OtomotifProperti\nLainnya\n\n</p>\n"
 . "<p>Untuk melihat iklan yang ada di website superiklan.localhost anda
tidak perlu menjadi \n"
 . "kami. Akan tetapi jika anda ingin memasang iklan, maka anda harus
registrasi dulu menjadi \n"
 . "menjadi member kami. Untuk menjadi member sama sekali gratis!.</p>"
 . "Klik link dibawah ini untuk registrasi

\n"
 . "Registrasi\n";

// panggil class skin
$skin = new skin; // buat objek skin
$skin->ganti_skin('template/skin_utama.php'); // tentukan file template
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{SISI1}', $iklan_sisi.$cari);
$skin->ganti_tag('{SISI2}', $daftar_berita);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan** dengan nama **index.php**. Tidak ada yang istimewa pada file ini. Variabel \$iklan\_sisi dan \$cari kita panggil dari file var\_utama.php. Output dari file ini ditunjukkan oleh gambar 4.31

Untuk sementara jika anda mengklik halaman-halaman lain maka muncul error karena file tersebut belum kita buat.



Gambar 4.31: Tampilan utama halaman suepriklan.localhost

File berikutnya yang akan kita buat adalah file yang berfungsi untuk menyediakan form login sekaligus memprosesnya. Pada PHP DESIGNER 2005 klik menu **File > New > PHP**, kemudian salin listing kode dibawah ini.

#### Script dari file *login.php*

<?php

```

** Nama File.....: login.php
** Penulis.....: Rio Astamal
** Tanggal.....: 15-10-2005
** Penjelasan....: menampilkan form login member dan
 memprosesnya

```

session\_start(); // karena berhubungan dengan session

// panggil file-file yang diperlukan

include ('inc/class\_skin.php');

include ('inc/fungsi.php');

include ('template/var\_utama.php');

// dapatkan data proses dari URL

\$proses = \$\_GET['proses'];

if (\$proses == '')

\$proses = 'form';

\$proses = filter\_str(\$proses);

// handel setiap proses dengan case dan switch

switch (\$proses)

{

```

/*
case 'form':
*/

$login = "<h2>Member Login</h2>\n"
 . "<p>Bagi member silahkan login dibawah ini.</p>\n"
 . "<form action='login.php?proses=proses_form' method='post'>\n"
 . "<table border='0' cellpadding='4'>\n"
 . " <tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>\n"
 . "<tr>\n <td>Username: </td>\n"
 . " <td><input type='text' name='username'></td>\n </tr>\n"
 . "<tr>\n <td>Password: </td>\n"
 . " <td><input type='password' name='password'></td>\n </tr>"
 . "<tr>\n <td></td>\n"
 . " <td><input type='submit' value='LOGIN'></td>\n </tr>\n"
 . "<tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>\n"
 . "</table>\n</form>\n"
 . "<p>Lupa Password? Klik disini.</p>\n"
 . "<p>Belum Menjadi Member? Daftar
sekarang.</p>\n";
break;
*/

// ambil data yang dipost sekaligus filter
$username = filter_str($_POST['username']);
$password = filter_str($_POST['password']);
// enkripsi password
$password = balik_md5($password);

koneksi_db(); // koneksi ke MySQL server

// gunakan fungsi login untuk mencocokkan data
if (!login('member', $username, $password))
 $login = "Username atau password salah.
\n $kembali";
else
{
 $_SESSION['member'] = $username; // buat session member
 $login = "Login berhasil. Klik disini "
 . "untuk masuk ke member area.";
}

mysql_close();
}

// panggil class skin
$skin = new skin; // buat objek skin
$skin->ganti_skin('template/skin_utama.php'); // tentukan file template
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $login);
$skin->ganti_tag('{SISI1}', $iklan_sisi);
$skin->ganti_tag('{SISI2}', $daftar_berita);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan** dengan nama **login.php**. Untuk sementara jangan login dulu karena percuma anda belum terdaftar didatabase juga halaman membernya belum kita buat.

Selanjutnya kita akan membuat file yang memproses pendaftaran/registrasi user. Langsung saja buat file PHP baru pada PHP DESIGNER 2005 kemudian ketik kode berikut.

#### Script dari file ***daftar.php***

---

```
<?php

/*
** Nama File.....: daftar.php
** Penulis.....: Rio Astamal
** Tanggal.....: 15-10-2005
** Penjelasan....: menampilkan form registrasi dan
** memprosesnya
***** */

// panggil file-file yang diperlukan
include ('inc/class_skin.php');
include ('inc/fungsi.php');
include ('template/var_utama.php');

// dapatkan data proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form';

$proses = filter_str($proses);

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*
 case 'form':
 */

$judul = "<h2>Form Registrasi - superiklan.localhost</h2>\n";
// buat form dan tabel untuk registrasi
$reg = "<p>Mohon isi semua field dibawah ini untuk mendaftar menjadi \n".
 ".member superiklan. Klik DAFTAR untuk melanjukan proses
registrasi.</p>\n".
 "<form action='daftar.php?proses=proses_form' method='post'>\n".
 "<table border='0' cellpadding='4' width='100%'\>\n".
 "<tr bgcolor='#7cb500'\>\n".
 ". <td class='putih' colspan='2' align='center'>Form
Registrasi</td></tr>\n".
 "<tr>\n <td>Username: </td>\n".
 ". <td><input type='text' name='username' maxlength='16'>".
 ". max. 16 karakter. </td>\n</tr>\n".
 "<tr>\n <td>Password: </td>\n".
 ". <td><input type='password' name='password' maxlength='16'>".
 ". max. 16 karakter. \n</td></tr>\n".
 "<tr>\n <td>Nama Lengkap: </td>\n".
 ". <td><input type='text' name='nama'></td>\n </tr>\n".
 "<tr>\n <td>Email: </td>\n".
 ". <td><input type='text' name='email'></td>\n </tr>\n".
 "<tr>\n <td>Alamat: </td>\n".
 ". <td><input type='text' name='alamat' size='50'></td>\n</tr>".
 "<tr>\n <td>Kota: </td>\n".
 ". <td><input type='text' name='kota'></td>\n </tr>\n".
 "<tr>\n <td>Telp./HP</td>\n".
 ". <td><input type='text' name='telpon'></td>\n </tr>\n".
 "<tr>\n <td></td>\n"
```

```

. " <td><input type='submit' value='DAFTAR'></td>\n </tr>"

. "<tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>\n"

. "</table>\n</form>";

break;

/************/

case 'proses_form':

/************/

// ambil data yang di-post dari form registrasi

$username = $_POST['username'];

$password = $_POST['password'];

$email = $_POST['email'];

// tambahkan addslashes()

$nama = addslashes($_POST['nama']);

$alamat = addslashes($_POST['alamat']);

$kota = addslashes($_POST['kota']);

$telp = addslashes($_POST['telp']);

$pesan_error = ''; // variabel untuk pesan error

// cek setiap field

if (!cek_field($_POST))

 $pesan_error = "Error: Masih ada field yang kosong.
\n";

if (!cek_email($email))

 $pesan_error .= "Error: Email tidak valid.
\n";

$format = '[^a-zA-Z0-9_]'; // username dan password hanya alpanumerik dan _

if (ereg($format, $username) || ereg($format, $password))

 $pesan_error .= "Error: Username atau password hanya boleh terdiri dari "

 ."alfabet, numerik dan _.
\n";

$password = balik_md5($password); // enkripsi password

// cek isi $pesan_error jika tidak kosong maka ada error

if ($pesan_error != '')

 $reg = $pesan_error.$kembali;

else

{
 konek_db(); // koneksi ke MySQL server

 // lakukan query

 $hasil = mysql_query("INSERT INTO member VALUES('$username', '$password',

'$nama', '$email', '$alamat', '$kota', '$telp')");

 // cek status

 if (!$hasil)

 $reg = "Error: Gagal memasukkan data ke database. Kontak

admin.
\n".$kembali;

 else

 $reg = "Proses registrasi sukses. Klik Login untuk "

 ."masuk ke member area.";

 // masukkan juga ke tb_email

 $hasil2 = mysql_query("INSERT INTO tb_email VALUES('$username', '$email',

0)");

 if (!$hasil2)

 $reg .= "<p>Error: Gagal memasukkan ke tb_email.</p>";

 mysql_close();
}

```

```

break;
***** */

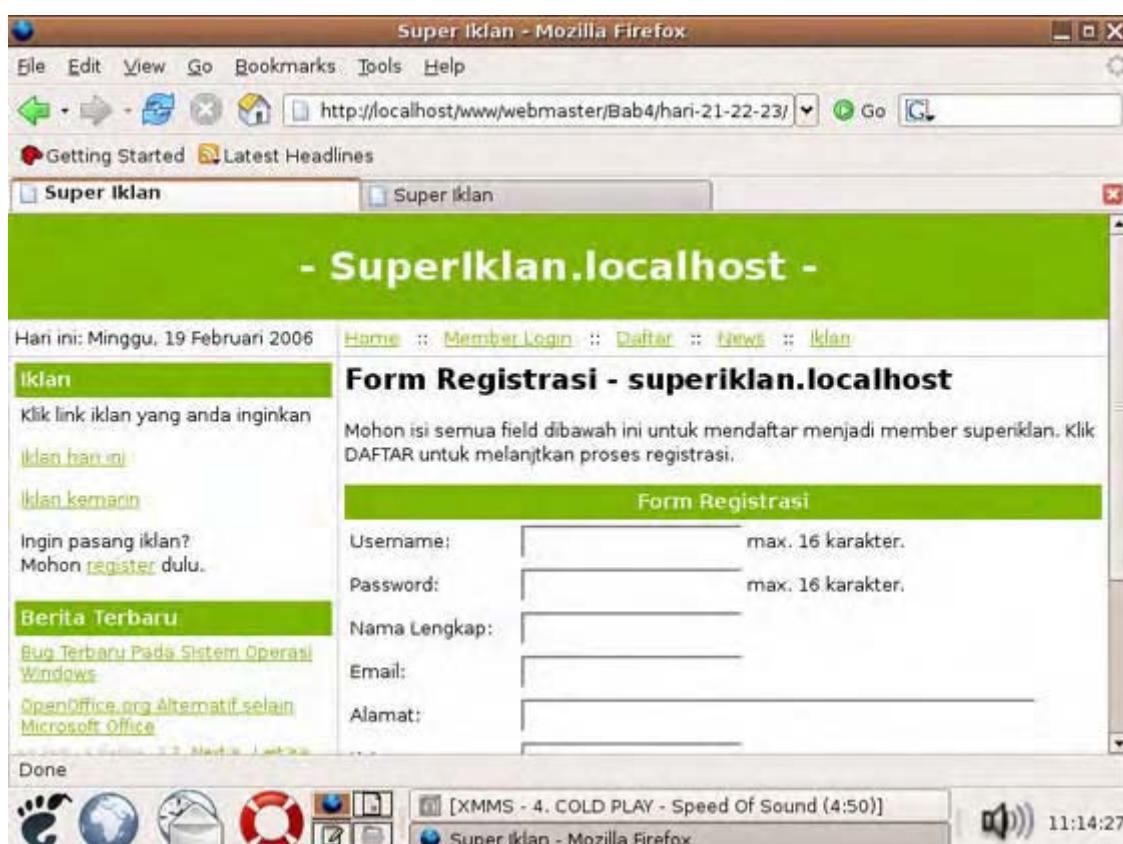
} // akhir dari switch

// panggil class skin
$skin = new skin; // buat objek skin
$skin->ganti_skin('template/skin_utama.php'); // tentukan file template
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $reg);
$skin->ganti_tag('{SISI1}', $iklan_sisi);
$skin->ganti_tag('{SISI2}', $daftar_berita);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan** dengan nama **daftar.php**. Output dari file ini ditunjukkan oleh gambar 4.32



Gambar 4.32: Tampilan form registrasi/pendaftaran

Selanjutnya kita akan membuat file yang akan menampilkan cuplikan berita dan keseluruhan berita. Untuk itu buat file PHP baru pada PHP DESIGNER 2005, lalu ketik kode berikut.

Script dari file **news.php**.

```

<?php
***** *****
** Nama File.....: news.php
** Penulis.....: Rio Astamal
** Tanggal.....: 15-10-2005
** Penjelasan....: menampilkan cuplikan berita dan
 berita lengkap
***** *****

```

```

// panggil file-file yang diperlukan
include ('inc/class_skin.php');
include ('inc/fungsi.php');
include ('template/var_utama.php');

// dapatkan data proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

$proses = filter_str($proses);

konek_db(); // koneksikan ke MySQL server

switch ($proses)
{
 *****/
 case 'view':
 /***/

 // dapatkan data page dari URL
 $page = $_GET['page'];
 if ($page == '')
 $page = 0;

 $page = filter_str($page);

 // panggil class halaman
 $hal = new halaman; // buat objek halaman
 $hal->set_tabel('tb_berita'); // tentukan tabel
 $hal->set_page($page); // data halaman yang dihandel
 $pph = $hal->set_pph($u_bph); // berita per halaman (konfig.php)

 // lakukan query
 $hal->query_SQL(1); // SELECT * FROM tb_berita
 $jml_berita = $hal->get_jml_data(); // dapatkan jumlah berita
 $hal->get_jml_hal(); // dapatkan jumlah halaman

 $record = $hal->get_record(); // dapatkan jumlah record
 $hasil = $hal->query_SQL("SELECT * FROM tb_berita ORDER BY id_berita DESC LIMIT
 $record, $pph");

 // variabel untuk menampilkan cuplikan berita
 $news = "<table border='0' cellpadding='4' width='100%'\>\n"
 . "<tr bgcolor='#7cb500'\>\n"
 . " <td class='putih'>Berita Terbaru</td>\n</tr>\n";

 // tampilkan hasil dengan looping
 while ($data = mysql_fetch_array($hasil))
 {
 $isi_berita = nl2br($data['isi_berita']);
 $tgl_berita = "[$data[tgl_berita]]";
 $jdl_berita = "$tgl_berita
$data[jdl_berita]

\n";

 $cuplikan = array(); // buat variabel array untuk cuplikan kata
 $pecahan_kata = explode(" ", $isi_berita); // pecah setiap kata

 // lakukan looping untuk mendapatkan 25 kata pertama
 for ($i=0; $i<25; $i++)
 $cuplikan[$i] = $pecahan_kata[$i];
 }
}
```

```

$cuplikan = implode(" ", $cuplikan); // gabung ke 25 kata tersebut
$link = "

baca
selengkapnya...<hr>";

// susun tampilan
$news .= "<tr><td>$jdl_berita $cuplikan... $link</td></tr>\n";
}

$hal->set_hal(); // tentukan nilai Last, Next, dan Before
$news .= "</table>\n";
.$hal->show_page("news.php?proses=view")."\n";

break;
***** */

***** */
case 'full_news':
***** */

// dapatkan id berita dari URL
$id = $_GET['id'];
if ($id == '')
 $id = 1;

$id = filter_str($id);

// lakukan query untuk menampilkan keseluruhan isi berita
$hasil = mysql_query("SELECT * FROM tb_berita WHERE id_berita='$id'");
$data = mysql_fetch_array($hasil); // pecah menjadi array
$isi = htmlspecialchars($data['isi_berita']); // agar tag HTML tidak diproses

// tampilkan isi berita
$news = "$data[tgl_berita]
\n$data[jdl_berita]\n"
 . "<p>".nl2br(stripslashes($data['isi_berita']))."</p>\n"
 . "Berita Lainnya\n";

break;
***** */
}

mysql_close();

// panggil class skin
$skin = new skin; // buat objek skin
$skin->ganti_skin('template/skin_utama.php'); // tentukan file template
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $news);
$skin->ganti_tag('{SISI1}', $iklan_sisi);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan** dengan nama **news.php**. Output dari file ini dapat anda lihat pada gambar 4.33



Gambar 4.33: Daftar cuplikan berita

### PENJELASAN SCRIPT

File ini memiliki dua case yaitu *view* untuk cuplikan berita dan *full\_news* untuk berita lengkap. Untuk case *view* kita gunakan class halaman untuk menampilkan seluruh cuplikan berita dan nomor link halaman.

Pada blok kode looping kita membuat variabel \$cuplikan yang bertipe array. File ini yang akan menampung 25 kata pertama dari setiap berita. Setelah itu kita pecah setiap kata pada isi berita menggunakan `explode()` dan hasilnya kita masukkan pada variabel \$pecahan\_kata.

```
for ($i=0; $i<25; $i++)
 $cuplikan[$i] = $pecahan_kata[$i];
```

looping for diatas berguna untuk mendapatkan 25 kata pertama dari isi berita yang hasilnya kita masukkan pada variabel \$cuplikan. Karena \$cuplikan masih dalam array maka kita gabungkan setiap array pada \$cuplikan hingga menjadi kalimat-kalimat utuh.

```
$cuplikan = implode(" ", $cuplikan); // gabung ke 25 kata tersebut
```

Jika ingin membaca berita secara lengkap user dapat mengklik link [baca selengkapnya](#). Sekarang case *full\_news* yang dijalankan. Pertama kita mengambil ID berita yang ada pada URL kemudian melakukan query SQL untuk mendapatkan isi berita secara lengkap.

File berikutnya adalah *iklan.php*. File ini merupakan file yang paling penting dari website superiklan.localhost karena file inilah yang menampilkan daftar iklan. Yang merupakan subject utama website ini. Pada PHP DESIGNER 2005 klik menu **File > New > PHP** kemudian ketik kode berikut.

#### Script dari file *iklan.php*

```
<?php
/*

```

```

** Nama File.....: iklan.php ***
** Penulis.....: Rio Astamal ***
** Tanggal.....: 15-10-2005 ***
** Penjelasan....: menampilkan iklan baik iklan hari ***
** atau kemarin ***
******/
```

```

// panggil file-file yang diperlukan
include ('inc/class_skin.php');
include ('inc/fungsi.php');
include ('template/var_utama.php');
include ('inc/class_waktu.php');

// dapatkan data proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';

$proses = filter_str($proses);

koneksi_db(); // koneksikan ke MySQL server

// handel setiap proses dengan case dan switch
switch ($proses)
{
 *****/
 case 'view':
 /***/

 $hari = $_GET['hari'];
 if ($hari == '')
 $hari = 'today';

 $hari = filter_str($hari);

 $waktu = new waktu; // buat objek waktu
 $waktu->set_date();
 $waktu->set_mode(0); // mode pengurangan tanggal
 $kemarin = $waktu->set_tgl(0, 1); // dikurangi 1
 // cek isi dari $hari untuk menentukan judul heading
 if ($hari == 'today')
 $heading = "<h2>Iklan Untuk Hari ini</h2>\n";
 else if ($hari == 'kemarin')
 {
 $heading = "<h2>Iklan kemarin, $kemarin</h2>\n";
 }
 else
 $heading = "<h2>Iklan sebelum tanggal $kemarin</h2>\n";

 $iklan = "<p>Silahkan pilih kategori iklan yang ingin anda tampilkan. \n"
 . "Klik masing-masing link menampilkan daftar iklan.

\n"
 . "Daftar Kategori iklan: </p>\n"
 . "\n"
 . "<p>Komputer</p>\n"
 . "<p>Internet</p>\n"
 . "<p>Elektornik</p>\n"
 . "<p>Lowongan</p>\n"

```

```

 . "<p>Otomotif</p>\n"
 . "<p>Properti</p>\n"
 . "<p>Lainnya</p>\n"
 . "\n"
 . "<p>Hari ini &nbsp \n"
 . "Kemarin &nbsp \n"
 . "Sebelum $kemarin</p>\n";

break;
***** */

***** /
case 'show':
***** /

// dapatkan data page dari URL
$page = filter_str($_GET['page']);
if ($page == '')
 $page = 0;

$hari = filter_str($_GET['hari']);

$kat = filter_str($_GET['kat']);
if ($kat == '')
 $kat = 'komputer';

$waktu = new waktu; // buat objek waktu
$waktu->set_date();

// cek kondisi hari untuk menentukan timestamp
if ($hari == 'today')
{
 $waktu->set_mode(2); // tanggal tetap
 $tstamp = $waktu->set_tgl(1); // sekarang dalam detik
}
else if ($hari == 'kemarin' || $hari == 'lama')
{
 $waktu->set_mode(0); // dikurangi
 $tstamp = $waktu->set_tgl(1, 1); // kemarin dalam detik
}
// variabel untuk melakukan query
$query = "SELECT * FROM tb_iklan WHERE kategori='$kat' AND timestamp=$tstamp";
if ($hari == 'lama')
 $query = "SELECT * FROM tb_iklan WHERE kategori='$kat' AND timestamp<$tstamp";

$page = filter_str($page);
$kat = filter_str($kat);

// panggil class halaman
$hal = new halaman; // uat objek halaman
$hal->set_tabel('tb_iklan'); // tentukan tabel
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph(2); // berita per halaman

// lakukan query
$hal->query_SQL($query); // lakukan query
$jml_iklan = $hal->get_jml_data(); // dapatkan jumlah iklan
$hal->get_jml_hal(); // dapatkan jumlah halaman

if ($jml_iklan == 0)

```

```

$iklan = "<p>Tidak ada iklan untuk kategori
$kat.</p>\n$kembali";
else
{

$record = $hal->get_record(); // dapatkan jumlah record
$hasil = $hal->query_SQL($query." ORDER BY id_iklan DESC LIMIT $record, $pph");

// lakukan looping untuk menampilkan semua iklan
$iklan = "<h2>Daftar Iklan untuk kategori: $kat</h2>\n"
. "<p>"
. "<< Kembali ke Daftar Kategori</p>\n"
. "<table border='0' cellpadding='4' width='100%>\n";

while ($data = mysql_fetch_array($hasil))
{
 $isi = nl2br(stripslashes($data['isi_iklan']));
 $jdl = stripslashes($data['jdl_iklan']);
 $iklan .= "<tr bgcolor='#7cb500'>\n"
. " <td class='jdl_iklan'>$jdl &nbsp &nbsp $data[tgl_post]</td>\n"
. "</td>\n</tr>\n"
. "<tr>\n <td class='kecil'>$isi</td>\n"
. "<td></td>\n</tr>\n";
}

$hal->set_hal();
$iklan .= "<tr height='20' bgcolor='#7cb500'><td colspan='2'></td></tr>\n"
. "</table>\n"
. $hal->show_page("iklan.php?proses=show&hari=$hari&kat=$kat");
}

break;
***** */

} // akhir dari switch

mysql_close();

// panggil class skin
$skin = new skin; // buat objek skin
$skin->ganti_skin('template/skin_utama.php'); // tentukan file template
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $heading);
$skin->ganti_tag('{UTAMA}', $iklan);
$skin->ganti_tag('{SISI1}', $iklan_sisi.$cari);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan** dengan nama **iklan.php**. Secara default jika user mengklik link iklan pada menu. Maka yang ditampilkan adalah iklan pada hari ini. Namun user tetap dapat memilih untuk melihat iklan kemarin atau dua hari yang lalu dan sebelumnya. User cukup mengklik link yang ada dibawah kategori iklan untuk memilih waktu iklan. Lihat gambar 4.34

## PENJALASAN SCRIPT

Pada case *view* kita diharapkan beberapa kondisi yaitu today, kemarin, dan lama. Secara default nilai dari \$hari adalah today. Untuk menampilkan tanggal kemarin jika nilai dari \$hari adalah kemarin kita gunakan class waktu.

```
$waktu = new waktu; // buat objek waktu
$waktu->set_date();
$waktu->set_mode(0); // mode pengurangan tanggal
$kemarin = $waktu->set_tgl(0, 1); // dikurangi 1
```

Isi dari \$kemarin merupakan tanggal kemarin dalam format dd-mm-yyyy.



Gambar 4.34: Daftar iklan

Pada case *show* kita mengambil isi dari hari pada URL untuk menentukan iklan pada tanggal berapa yang akan ditampilkan. Untuk keperluan itu kita panggil class waktu. Namun sebelumnya kita harus melihat beberapa kondisi dari \$hari. Jika kondisi hari adalah *today* yang berarti sekarang maka timestamp-nya kita set hari ini seperti berikut,

```
if ($hari == 'today')
{
 $waktu->set_mode(2); // tanggal tetap
 $tstamp = $waktu->set_tgl(1); // sekarang dalam detik
}
```

Jika isi dari \$hari tidak sama dengan *today* berarti isinya *kemarin* atau *lama*. Yang membedakan antara *kemarin* dan *lama* hanyalah saat melakukan query. Yaitu jika isi dari \$hari adalah *lama* maka operator pada query SQL-nya kita beri tanda < (kurang). Yang berarti iklan sebelum kemarin.

```
$query = "SELECT * FROM tb_iklan WHERE kategori='$kat' AND timestamp=$tstamp";
if ($hari == 'lama')
 $query = "SELECT * FROM tb_iklan WHERE kategori='$kat' AND timestamp<$tstamp";
```

Langkah selanjutnya seperti biasa kita gunakan class halaman untuk menampilkan semua data dan menampilkan link nomor halaman.

File berikutnya yang akan kita buat adalah file berfungsi untuk melakukan pencarian atas keriteria yang diinginkan oleh user. File ini akan dipanggil jika kita melakukan submit pada

form pencarian iklan. Untuk membuat file ini klik **File > New > PHP** pada PHP DESIGNER 2005, lalu ketik kode berikut.

### Script dari file *cari.php*

---

```
<?php

/*
** Nama File.....: cari.php
** Penulis.....: Rio Astamal
** Tanggal.....: 15-10-2005
** Penjelasan....: Menampilkan iklan hasil pencarian
***** */

// panggil file-file yang diperlukan
include ('inc/class_skin.php');
include ('inc/fungsi.php');
include ('template/var_utama.php');

koneksi_db(); // koneksi ke MySQL server

// dapatkan data dari URL
$page = $_GET['page'];
if ($page == '')
 $page = 0;

$kat = $_GET['kat'];
$keyword = $_GET['keyword'];

// filter data
$page = filter_str($page);
$kat = filter_str($kat);
$keyword = filter_str($keyword, " "); // spasi diperbolehkan

// variabel untuk query
$query = "SELECT * FROM tb_iklan WHERE isi_iklan LIKE '%$keyword%' AND
kategori='$kat'";

// panggil class halaman
$hal = new halaman; // buat objek halaman
$hal->set_tabel('tb_iklan'); // tentukan tabel
$hal->set_page($page); // data halaman yang dihandel
$pph = $hal->set_pph(2); // berita per halaman

// lakukan query
$hal->query_SQL($query); // lakukan query
$jml_iklan = $hal->get_jml_data(); // dapatkan jumlah iklan
$hal->get_jml_hal(); // dapatkan jumlah halaman

if ($jml_iklan == 0) // cek jumlah iklan
 $iklan = "<p>Iklan yang anda cari tidak
ditemukan.</p>\n$kembali";
else
{

$record = $hal->get_record(); // dapatkan jumlah record
$hasil = $hal->query_SQL($query." ORDER BY id_iklan DESC LIMIT $record, $pph");

// lakukan looping untuk menampilkan semua iklan
$iklan = "<h2>Hasil pencarian untuk kategori: $kat</h2>\n"
 . "<p>"
 . "<< Kembali ke Daftar Kategori</p>\n"
 . "<table border='0' cellpadding='4' width='100%>\n";
```

```

while ($data = mysql_fetch_array($hasil))
{
 $iklan .= "<tr bgcolor='#7cb500'>\n"
 . "<td class='jdl_iklan'>$data[jdl_iklan]
$data[tgl_post]</td>\n</tr>\n"
 . "<tr>\n <td class='kecil'>".nl2br($data[isi_iklan])."</td>\n"
 . "<td></td>\n</tr>\n";
}

$hal->set_hal();
$iklan .= "<tr height='20' bgcolor='#7cb500'><td colspan='2'></td></tr>\n"
 . "</table>\n"
 . $hal->show_page("cari.php?kat=$kat&keyword=$keyword");
}

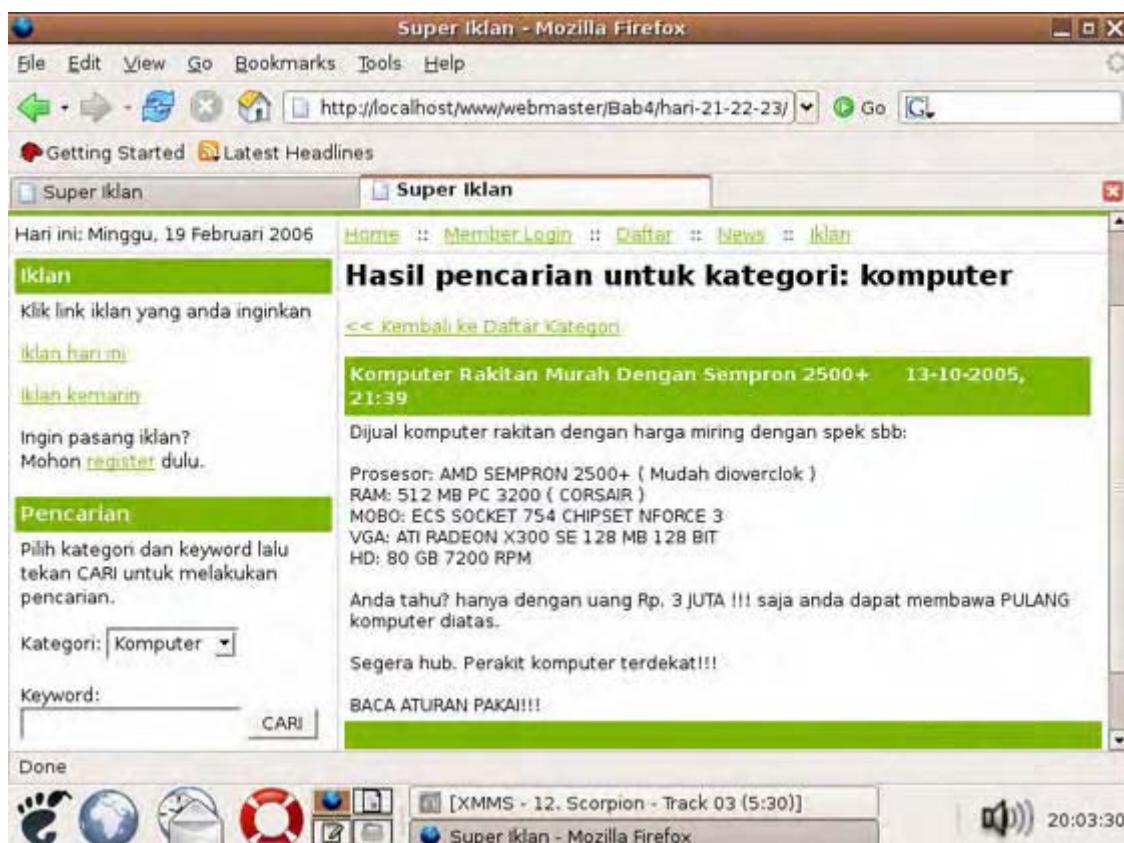
mysql_close();

// panggil class skin
$skin = new skin; // buat objek skin
$skin->ganti_skin('template/skin_utama.php'); // tentukan file template
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $iklan);
$skin->ganti_tag('{SISI1}', $iklan_sisi.$cari);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan** dengan nama **cari.php**. Gambar 4.35 menunjukkan hasil pencarian pada ketogori komputer.



Gambar 4.35: Daftar iklan hasil pencarian

## PENJALASAN SCRIPT

Form pencarian yang ada disamping halaman merupakan isi dari variabel \$cari yang ada pada file var\_utama.php. Pada form pencarian kita tidak menggunakan *method POST* melainkan *GET* karena jika kita menggunakan POST maka kita tidak bisa berpindah antar halaman hasil pencarian.

Hal itu disebabkan karena nilai dari \$keyword sudah hilang karena melalui method post. Sedangkan jika melalui GET kita tetap dapat menyimpannya karena GET menyimpan nilai keyword pada URL yang dapat kita ambil saat memanggil metode show\_page().

Hal menarik lainnya adalah kita mengambil isi dari \$keyword pada URL menggunakan metode filter\_str(\$keyword, " "). Parameter ke-2 adalah karakter yang diperbolehkan artinya pada fungsi tersebut kita membiarkan spasi untuk tidak difilter. Ini karena user biasanya melakukan pencarian tidak hanya memasukkan satu kata melainkan dua kata.

File terakhir untuk direktori *iklan* adalah *lupa\_pass.php*. File ini berfungsi untuk melakukan reset password dan mengirimkan password kepada user yang lupa akan passwordnya tersebut. Buat file baru pada PHP DESIGNER 2005 kemudian ketik kode berikut.

### Script dari file *lupa\_pass.php*

```
<?php

/*
** Nama File.....: lupa_pass.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk menampilkan form lupa password
** dan memprosesnya
*/
// panggil file-file yang diperlukan
include ('inc/class_skin.php');
include ('inc/fungsi.php');
include ('template/var_utama.php');

// dapatkan data proses dari URL
$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form';

$proses = filter_str($proses);

koneksi_db(); // koneksikan ke MySQL server

// handel setiap proses dengan case dan switch
switch ($proses)
{
 /*
 case 'form':
 */

$judul = "<h2>Lupa Password</h2>";
$lupa = "
<p>Isikan username dan email anda kemudian klik KIRIM untuk mereset password anda</p>
<form action='lupa_pass.php?proses=kirim' method='post'>
<table border='0' cellpadding='4'>
<tr bgcolor='#7cb500'>
```

```

<td class='putih' colspan='2'>Form Lupa Password</td></tr>
<tr> \n<td>Username: </td>
<td><input type='text' name='username'></td></tr>
<tr> \n<td>Email: </td>
<td><input type='text' name='email'></td></tr>
<tr><td></td><td><input type='submit' value='K I R I M'></td></tr>
<tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>
</table>
</form>\n\n';

break;
***** */

***** */
case 'kirim':
***** */

// ambil data yang dipost
$username = filter_str($_POST['username']);
$email = $_POST['email'];

// cek kevalidan email
if (!cek_email($email))
 $lupa = "<p>Error: Email tidak valid.
\n$kembali</p>\n";
else
{
 // lakukan query untuk mencocokkan data
 $hasil = mysql_query("SELECT * FROM member WHERE username='$username' AND email='$email'");

 // cek hasil
 if (mysql_num_rows($hasil) == 0)
 $lupa = "<p>Error: Username atau email tidak ada didatabase.
\n$kembali</p>";
 else
 {
 // jika cocok maka buat password baru, update database lalu kirim email
 // panggil fungsi pass_acak() untuk mendapatkan password secara acak
 $new_pass = pass_acak();

 // enkripsi password
 $pass_enkrip = balik_md5($new_pass);

 // update password yang ada di database
 $q_update = mysql_query("UPDATE member SET password='$pass_enkrip' WHERE username='$username'");

 // cek status
 if (!$q_update)
 $lupa = "<p>Error: Gagal mengupdate password didatabse.
\n$kembali</p>" .
 ". Kontak Admin";
 else
 {
 // kirim email
 $to = $email; // alamat email user
 $subject = "Password Baru Anda - superiklan.localhost";
 $tgl_reset = date('d-m-Y, H:i');
 $isi = "Dari admin@superiklan.localhost\n".
 "=====\\n\\n".
 ". Pada tanggal $tgl_reset anda dengan username $username, telah\\n".
 ". melakukan request password. Dan di bawah ini adalah password\\n".
 ". baru anda.\\n\\n".
 ". =====\\n"
 }
 }
}

```

```

 . "Username: $username\n"
 . "Password: $new_pass\n"
 . "=====\\n\\n"
 . "Gunakan password diatas untuk masuk ke member area. Kemudian\\n"
 . "update kembali password anda agar mudah anda ingat.\\n\\n"
 . "=====\\n"
 . "Admin superiklan.localhost\\n"
 . "=====";
$from = "From: admin@superiklan.localhost"; // ganti dengan email anda

// jika anda tidak memiliki program mail server atau anda belum di server
// sebenarnya, beri komentar pada fungsi mail berikut
mail($to, $subject, $isi, $from);

$lupa = "<p>Password berhasil direset. Silahkan cek email anda</p>";
}

}

break;
***** */

} // akhir dari switch

mysql_close(); // tutup koneksi ke MySQL server

// panggil class skin
$skin = new skin; // buat objek skin
$skin->ganti_skin('template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $lupa);
$skin->ganti_tag('{SISI1}', $iklan_sisi);
$skin->ganti_tag('{SISI2}', $daftar_berita);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan** dengan ***lupa\_pass.php***. Gambar 4.36 menunjukkan email yang diterima user setelah melakukan reset password. Jika anda menggunakan OS berbasis UNIX seperti Linux anda tidak perlu memberi komentar pada fungsi mail. Hanya saja anda harus sesuaikan penerimanya adalah user lokal.



Gambar 4.36: Email yang diterima user karena melakukan request reset password

Akhir dari file yang berada pada direktori **iklan**.

File-file berikutnya yang akan kita kerjakan adalah file yang berada pada direktori *member*. Dan ini merupakan direktori terakhir yang akan kita kerjakan pada proyek superiklan.localhost ini.

File pertama yang akan kita buat adalah *index.php*. File ini hanya bertugas memberikan pesan selamat datang dan fungsi untuk logout. Untuk membuatnya klik **File > New > PHP**, kemudian ketik kode berikutnya

#### Script dari file **index.php**

```
<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 16-10-2005
** Penjelasan....: menampilkan halaman utama untuk
** member area
***** */

session_start();

include ('../inc/class_skin.php');
include ('../template/member_var.php');

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'view';
```

```

$proses = filter_str($proses);

// cek user apakah sudah login atau belum
if (!cek_session('member'))
 $member = $not_login; // $not_login ada di member_var.php
else
{

switch ($proses)
{

/******
case 'view':
*****/

$judul = "<h2>Selamat Datang di Member Area</h2>\n";
$member = "<p>Terima kasih karena anda sudah bersedia menjadi member dari
website "
 ."superiklan.localhost. Karena hanya disini lah anda dapat memasang
iklan "
 ."secara efektif dan cepat. Anda dapat memasang iklan pada kategori
sesuai "
 ."dengan produk/jasa yang anda tawarkan.</p>\n"
 ."<p>Selain dengan menggunakan iklan baris, anda dapat menggunakan
fasilitas "
 ."email untuk mengirim email ke semua member superiklan.localhost.
Dengan "
 ."dengan demikian keefektifan iklan anda sangat tinggi. Namun untuk
fasilitas "
 ."email ini kami hanya memperbolehkan anda mengirim email hanya satu
kali "
 ."dalam 5 hari.</p>\n";

break;
*****/

/******
case 'logout':
*****/

if (!logout('member'))
 $member = "<p>Tidak bisa logout. Login Login
dulu</p>\n";
else
 $member = "<p>Anda telah logout dari sistem. Klik "
 ."disini untuk login kembali.</p>\n";

break;
*****/
}

}

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
// ganti tag tertentu dengan variabel yang diinginkan
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $member);
$skin->ganti_tag('{MENU}', $mem_menu);
$skin->ganti_tag('{SISI1}', $iklanku);
$skin->ganti_tag('{SISI2}', $login);
$skin->ganti_tampilan();

```

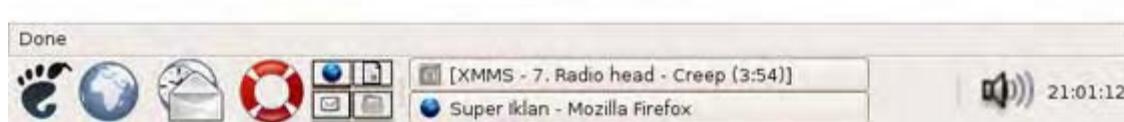
?>

---

Simpan pada direktori **iklan\member** dengan nama ***index.php***. Output dari file ini ditunjukkan oleh gambar 4.37



The screenshot shows a Mozilla Firefox browser window titled "Super Iklan - Mozilla Firefox". The address bar displays "http://localhost/www/webmaster/Bab4/hari-21-22-23/". The main content area has a green header bar with the text "- SuperIklan.localhost -". Below the header, there's a navigation menu with links: "Home :: Update Profil :: Pasang iklan :: Email :: Log out". A sidebar on the left is titled "IklanKu" and contains text about editing ads and two links: "Komputer Rakitan Murah Dengan Sampson 2500+" and "DILUAL Laptop Toshiba Pentium 3!!". The main content area features a large heading "Selamat Datang di Member Area" and a message about the benefits of being a member. At the bottom of the page, it says "Anda login sebagai: webmaster". The footer contains the copyright notice "© 2005 superiklan.localhost".



Gambar 4.37: Halaman utama member area

Lanjut? OK, file berikutnya adalah *profil.php*. File ini bertugas untuk mengupdate profil dari user. Langsung saja buat file baru pada PHP DESIGNER 2005 kemudian ketik kode berikut.

#### Script dari file *profil.php*

---

```
<?php

/*
** Nama File.....: profil.php
** Penulis.....: Rio Astamal
** Tanggal.....: 12-10-2005
** Penjelasan....: Untuk mengupdate profil dari member
*/
session_start();

include ('../inc/class_skin.php');
include ('../template/member_var.php');

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form';

$proses = filter_str($proses);

konek_db(); // koneksikan ke MySQL server
```

```

// cek user apakah sudah login atau belum
if (!cek_session('member'))
 $profil = $not_login;
else
{

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'form':
 *****/

 // lakukan query untuk mendapatkan profil dari user tersebut
 $user = $_SESSION['member'];
 $hasil = mysql_query("SELECT * FROM member WHERE username='$user'");
 $data = mysql_fetch_array($hasil); // pecah menjadi array

 $judul = "<h2>Update Profil</h2>\n";
 $profil = "<p>Untuk mengupdate Klik tombol UPDATE. Pastikan semua field terisi.</p>\n"
 . "<form action='profil.php?proses=proses_form' method='post'>\n"
 . "<table border='0' cellpadding='4'>\n"
 . "<tr bgcolor='#7cb500' align='center'>\n"
 . " <td colspan='2' class='putih'>Form Update Profil</td></tr>\n"
 . "<tr> \n<td>Username: </td><td>$data[username]</td></tr>\n"
 . "<tr> \n<td>Password: </td>\n"
 . " <td><input type='password' name='pass1' maxlength='16'></td></tr>\n"
 . "<tr> \n<td>Ulangi: </td>\n"
 . " <td><input type='password' name='pass2' maxlength='16'></td></tr>\n"
 . "<tr> \n<td>Nama Lengkap: </td>\n"
 . " <td><input type='text' name='nama' value='". stripslashes($data['nama']) ." '></td></tr>\n"
 . "<tr> \n<td>Email: </td>\n"
 . " <td><input type='text' name='email' value='". stripslashes($data['email']) ." '></td></tr>\n"
 . "<tr> \n<td>Alamat: </td>\n"
 . " <td><input type='text' name='alamat' value='". stripslashes($data['alamat']) ." ' size='50'></td></tr>\n"
 . "<tr> \n<td>Kota: </td>\n"
 . " <td><input type='text' name='kota' value='". stripslashes($data['kota']) ." '></td></tr>\n"
 . "<tr> \n<td>Telp./HP: </td>\n"
 . " <td><input type='text' name='telpon' value='". stripslashes($data['telpon']) ." '></td></tr>\n"
 . "<tr><td></td><td><input type='submit' value='UPDATE'></td></tr>\n"
 . "<tr bgcolor='#7cb500' height='20'>\n"
 . " <td colspan='2'></td></tr>\n"
 . "<input type='hidden' name='username' value='".$data[username]."'>\n"
 . "</table>\n</form>\n";

 break;
 *****/
}

// ambil data yang dipost
$username = $_POST['username'];
$pass1 = filter_str($_POST['pass1']);
$pass2 = filter_str($_POST['pass2']);

```

```

// tambahkan addslashes()
$nama = addslashes($_POST['nama']);
$email = addslashes($_POST['email']);
$alamat = addslashes($_POST['alamat']);
$kota = addslashes($_POST['kota']);
$telpo = addslashes($_POST['telpo']);

// buat variabel pesan_error
$pesan_error = '';

// cek semua field
if (!cek_field($_POST))
 $pesan_error = "Error: Masih ada field yang kosong
\n";

if (!cek_email($email))
 $pesan_error .= "Error: Email tidak valid.
\n";

if ($pass1 != $pass2)
 $pesan_error .= "Error: Password tidak sama.
\n";

$pass1 = balik_md5($pass1); // enkripsi password

// cek isi dari pesan error jika tidak kosong maka ada error
if ($pesan_error != '')
 $profil = $pesan_error.$kembali;
else
{
 // masukkan ke database
 $hasil = mysql_query("UPDATE member SET password='$pass1', email='$email',
nama='$nama', alamat='$alamat', kota='$kota', telpon='$telpo' WHERE
username='$username'");
 // cek status
 if (!$hasil)
 $profil = "Profil gagal diupdate.
\n $kembali";
 else
 $profil = "Profil berhasil diupdate.
\n $kembali";
}

break;
***** */

} // akhir dari switch

} // akhir dari else

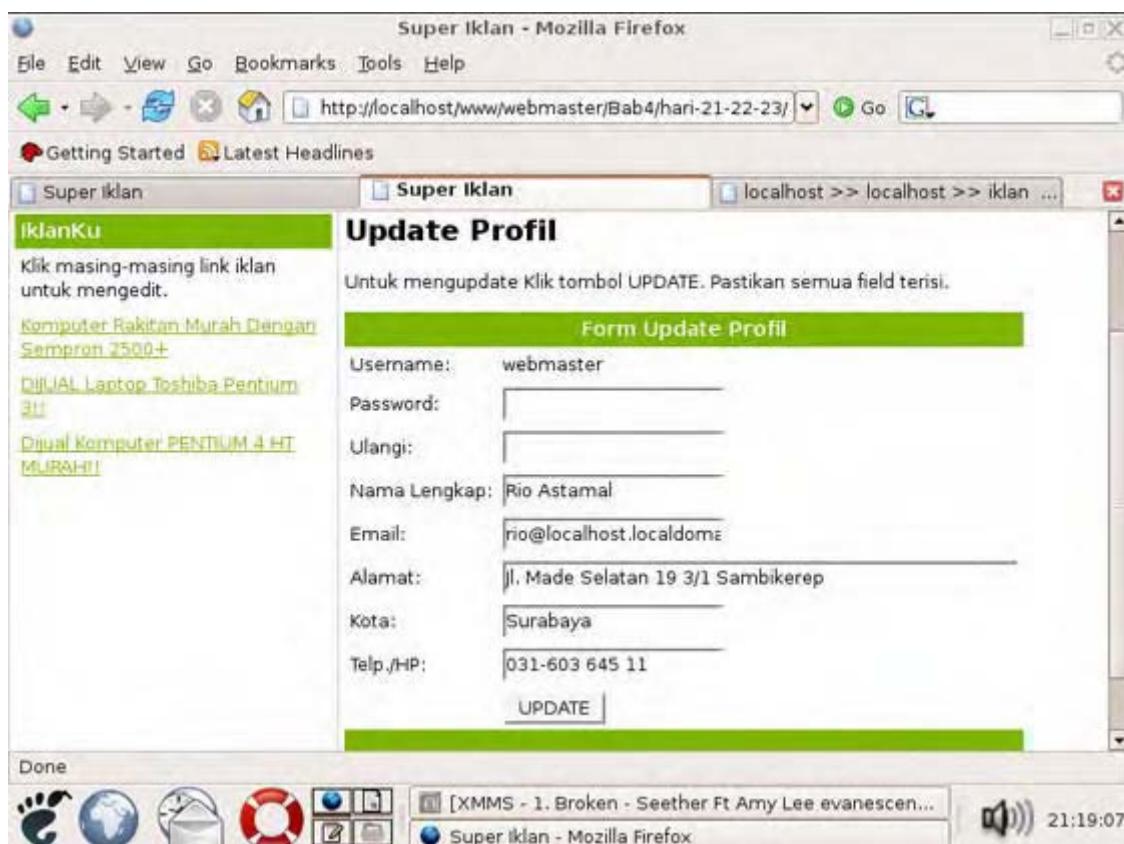
mysql_close(); // tutup koneksi MySQL Server

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $profil);
$skin->ganti_tag('{MENU}', $mem_menu);
$skin->ganti_tag('{SISI1}', $iklanku);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan\member** dengan nama ***profil.php***. Output dari file ini dapat anda lihat pada gambar 4.38



Gambar 4.38: Halaman profil untuk mengupdate profil

Selanjutnya kita akan membuat file yang bertugas menyedian form pemasangan iklan dan menyimpannya ke database. Pada PHP DESIGNER 2005 buat file PHP baru dengan mengklik menu **File > New > PHP**, kemudian ketik kode dibawah ini.

#### Script dari file *pasang.php*

```
<?php

/*
** Nama File.....: pasang.php
** Penulis....: Rio Astamal
** Tanggal....: 17-10-2005
** Penjelasan....: Untuk halaman pemasangan iklan
*/

session_start();

// panggil file-file yang diperlukan
include ('../inc/class_skin.php');
include ('../template/member_var.php');
include ('../inc/class_waktu.php');
include ('../inc/konfig.php');

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form';

$proses = filter_str($proses);

koneksi_db(); // koneksi ke MySQL server

// cek user apa sudah login atau belum
if (!cek_session('member'))
 $pasang = $not_login;
else
```

```

{
switch ($proses)
{
*****/*
case 'form':
*****/

$judul = "<h2>Pasang Iklan</h2>\n";

// cek jumlah iklan user ini jika sudah 10; tidak boleh post
$hasil = mysql_query("SELECT * FROM tb_iklan WHERE
username='$_SESSION[member]'");
$jml_iklan = mysql_num_rows($hasil); // jumlah iklan

// tentukan kondisi, $max_post ada di konfig.php
if ($jml_iklan > $max_post)
$pasang = "<p>Jumlah iklan anda sudah melebihi "
. "quota maksimum ($max_post iklan). Hapus salah satu iklan "
. "anda dan coba lagi.</p>\n";
else
{

$judul = "<h2>Pasang Iklan</h2>\n";
$pasang = "
<p>Pada halaman ini anda dapat memasang iklan sesuai dengan
jasa/produk yang anda tawarkan.</p>
$java
<form action='pasang.php?proses=proses_form' method='post'>
<table border='0' cellpadding='4'>
<tr bgcolor='#7cb500' align='center'>
<td colspan='2' class='putih'>Form Pasang Iklan</td></tr>
<tr> \n<td>Pilih kategori: </td>
<td><select name='kategori'>
<option value='komputer'>Komputer</option>
<option value='internet'>Internet</option>
<option value='elektronik'>Elektronik</option>
<option value='lowongan'>Lowongan</option>
<option value='otomotif'>Otomotif</option>
<option value='properti'>Properti</option>
<option value='lainnya'>Lainnya</option>
</select>\n</td></tr>
<tr> \n<td>Judul Iklan:</td>
<td><input type='text' name='jdl_iklan' size='50'
maxlength='100'></td></tr>
<tr> \n<td>Isi Iklan: </td>
<td><textarea name='isi' rows='10' cols='50'
onkeyup='batas_kar(this.form);'></textarea></td></tr>
<tr> \n<td>Sisa Karakter: </td>
<td><input type='text' name='sisa' value='500' size='3'
readonly='yes'></td></tr>
<tr> \n<td></td><td><input type='submit' value='P O S T'></td></tr>
<tr height='20' bgcolor='#7cb500'><td colspan='2'></td></tr>
</table>\n</form>\n\n";
}

break;
*****/
*****/
case 'proses_form':
*****/

```

```

// ambil data yang di-post
$kategori = $_POST['kategori'];
$jd1_iklan = addslashes($_POST['jd1_iklan']);
$isi = strip_tags(addslashes($_POST['isi']));

$pesan_error = ''; // nilai awal kosong

// cek setiap field
if (!cek_field($_POST))
 $pesan_error = "Error: Masih ada field yang kosong.
\n";

if (strlen($isi) < 10)
 $pesan_error .= "Error: Kelihatannya iklan anda terlalu pendek.
\n";

// jika $pesan_error tidak kosong maka ada error
if ($pesan_error != '')
 $pasang = $pesan_error.$kembali;
else
{
 // buat variabel untuk dimasukkan ke database
 $tgl = date('d-m-Y, H:i');
 $user = $_SESSION['member'];

 $waktu = new waktu;
 $waktu->set_date();
 $waktu->set_mode(2); // mode tetap
 $skr = $waktu->set_tgl(1); // sekarang dalam detik

 $hasil = mysql_query("INSERT INTO tb_iklan VALUES (0, '$kategori', '$user',
'$jd1_iklan', '$isi', '$tgl', '$skr')");
 // cek status
 if (!$hasil)
 $pasang = "Error: Tidak dapat memasukkan data ke database.
\n"
 . "Kontak Admin\n";
 else
 $pasang = "<p>Iklan berhasil di-post.</p>\n";
}

break;
***** */

}

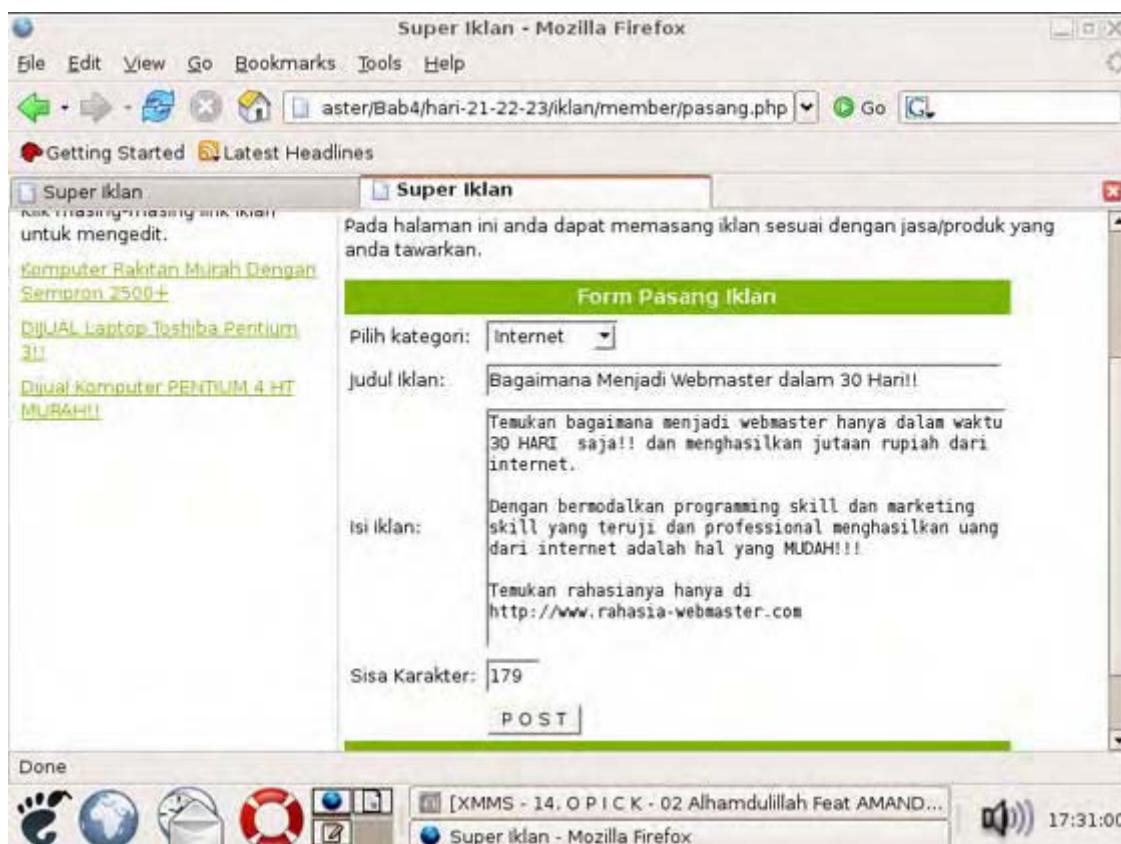
} // akhir dari else

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
// ganti tag tertentu dengan variabel yang diinginkan
$skin->ganti_tag('{SEKARANG}', $stgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $pasang);
$skin->ganti_tag('{MENU}', $mem_menu);
$skin->ganti_tag('{SISI1}', $iklanku);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **iklan\member** dengan nama ***pasang.php***. Output dari file ini ditunjukkan oleh gambar 4.39



Gambar 4.39: Tampilan halaman pasang iklan pada member area

### PENJALASAN SCRIPT

Sebenarnya script ini tidak terlalu istimewa. Kita hanya menyediakan form untuk pemasangan iklan kemudian kita memasukkannya pada database. Namun ada beberapa hal penting yaitu sebelum menyediakan form pemasangan iklan. Kita mengecek dulu jumlah iklan yang telah dipost oleh user.

```
if ($jml_iklan > $max_post)
```

Isi dari \$max\_post dapat anda tentukan pada file konfig.php. Untuk menghindari hal-hal yang tidak diinginkan seperti eksplorasi dan sebagainya. Kita menambahkan fungsi `strip_tags()` untuk menghilangkan tag-tag HTML. Juga `addslashes()` untuk memberi escaping karakter.

Kita juga membatasi jumlah karakter iklan yaitu sebanyak 500 karakter. Untuk itu kita gunakan javascript yang disimpan pada variabel \$java. Anda perhatikan bahwa nama dari textarea harus `isi` dan textbox harus `sisa`. Jika namanya berbeda maka script tersebut tidak berfungsi. Kecuali anda mengganti namanya pada javascriptnya juga.

Sesuai dengan sketsa proyek awal kita. Pada halaman member area ini kita menyediakan form untuk mengirim email ke semua member sebagai salah satu jenis iklan yang disediakan pengelola. Untuk membuatnya klik **File > New > PHP** pada PHP DESIGNER 2005, lalu ketik kode berikut.

#### Script dari file `email.php`

```
<?php
/*
** Nama File.....: email.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk mengirim email iklan ke
** semua member

```

```

session_start();

include ('../inc/class_skin.php');
include ('../template/member_var.php');
include ('../inc/class_waktu.php');
include ('../inc/konfig.php');

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form';

$proses = filter_str($proses);

konek_db(); // koneksikan ke MySQL server

// cek user apa sudah login atau belum
if (!cek_session('member'))
 $email_ad = $not_login;
else
{

switch ($proses)
{

/*****
case 'form':
*****/

$judul = "<h2>Email iklan ke seluruh member</h2>\n";
$email_ad = "<p>Silahkan isi suject dan isi iklan. Email ini akan dikirim ke semua member "
 . "superiklan.localhost</p>\n"
 . "$java.\n"
 . "<form action='email.php?proses=kirim' method='post'>\n"
 . "<table border='0' cellpadding='4'>\n"
 . "<tr bgcolor='#7cb500' align='center'>\n"
 . " <td colspan='2' class='putih'>Form Email</td></tr>\n"
 . "<tr> \n<td>Subject: </td>\n"
 . " <td><input type='text' name='subject' size='50'
maxlength='100'></td></tr>\n"
 . "<tr> \n<td>Isi Iklan: </td>\n"
 . " <td><textarea name='isi' rows='10' cols='50'
onkeyup='batas_kar(this.form);'></textarea></td></tr>\n"
 . "<tr> \n<td>Sisa karakter: </td>\n"
 . " <td><input type='text' name='sisa' size='3' value='500'
readonly='yes'></td></tr>\n"
 . "<tr><td></td><td><input type='submit' value='K I R I M'></td></tr>\n"
 . "<tr bgcolor='#7cb500' height='20'><td colspan='2'></td></tr>\n"
 . "</table>\n</form>\n\n";

break;
*****/

case 'kirim':
*****/

$hasil = mysql_query("SELECT * FROM tb_email WHERE
username='$_SESSION[member]' ");
$data = mysql_fetch_array($hasil);

$waktu = new waktu; // buat objek waktu
```

```

$waktu->set_date();
$waktu->set_mode(2); // mode tetap
$skr = $waktu->set_tgl(1); // sekarang dalam detik

$user = $_SESSION['member'];

// ambil data yang di-post
$subject = strip_tags($_POST['subject']." - superiklan.localhost";
$isi = strip_tags($_POST['isi']);

if (!cek_field($_POST))
 $email_ad = "Error: Masih ada field yang kosong.
\n$kembali";
else
{
if ($skr >= $data['next_post']) // sudah melewati lama hari yang ditentukan
{

$isi = "POSTED BY: $user\n"
. "*****\n"
. $isi."\n\n"
. str_repeat("*", 70)."\n"
. "Anda menerima email ini karena anda adalah member dari \n"
. "superiklan.localhost.\n\n"
. "(c) 2005 superiklan.localhost\n"
. str_repeat("*", 70);

$from = "From: $data[1]"; // email user

// lakukan looping untuk mengirim email ke seluruh member
$hasil = mysql_query("SELECT * FROM member WHERE username != '$user' ");
while ($data = mysql_fetch_array($hasil))
{
 $to = $data['email']; // alamat email masing2 member

 mail($to, $subject, $isi, $from); // kirim email
}

$waktu = new waktu; // buat objek waktu
$waktu->set_date();
$waktu->set_mode(1); // mode penambahan
$next = $waktu->set_tgl(1, $lama_email); // dalam detik, $lama_email hari ke depan

 // update nilai next_post
$query = mysql_query("UPDATE tb_email SET next_post=$next WHERE
username='$user'");
if (!$query)
 $email_ad = "<p>Gagal Mengupdate database.</p>\n";
else
 $email_ad = "<p>Email berhasil dikirim.</p>\n";
}
else
{
 // kalkulasi dari detik menjadi hari
$nanti = floor(($data['next_post'] - $skr) / 60 / 60 / 24);
$email_ad = "<p>Maaf, saat ini anda belum bisa mengirim
email. "
 . "Anda dapat mengirim email $nanti hari lagi.</p>\n";
}
}

break;

```

```

}

}

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $email_ad);
$skin->ganti_tag('{MENU}', $mem_menu);
$skin->ganti_tag('{SISI1}', $iklanku);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

---

Simpan pada direktori **iklan\member** dengan nama ***email.php***. Gambar 4.40 menunjukkan output dari file ini. Dan gambar 4.41 menunjukkan email yang diterima salah satu user.

### PENJALASAN SCRIPT

Untuk case form saya rasa tidak perlu dijelaskan. Yang perlu kita fokuskan adalah pada case *kirim*. Pada case ini kita pertama-tama kiat lakukan query pada tabel *tb\_email* untuk mendapatkan username, email dan next\_post.

Setelah itu kita memanggil class waktu untuk kita gunakan sebagai penentu kondisi apakah email dikirim atau tidak.

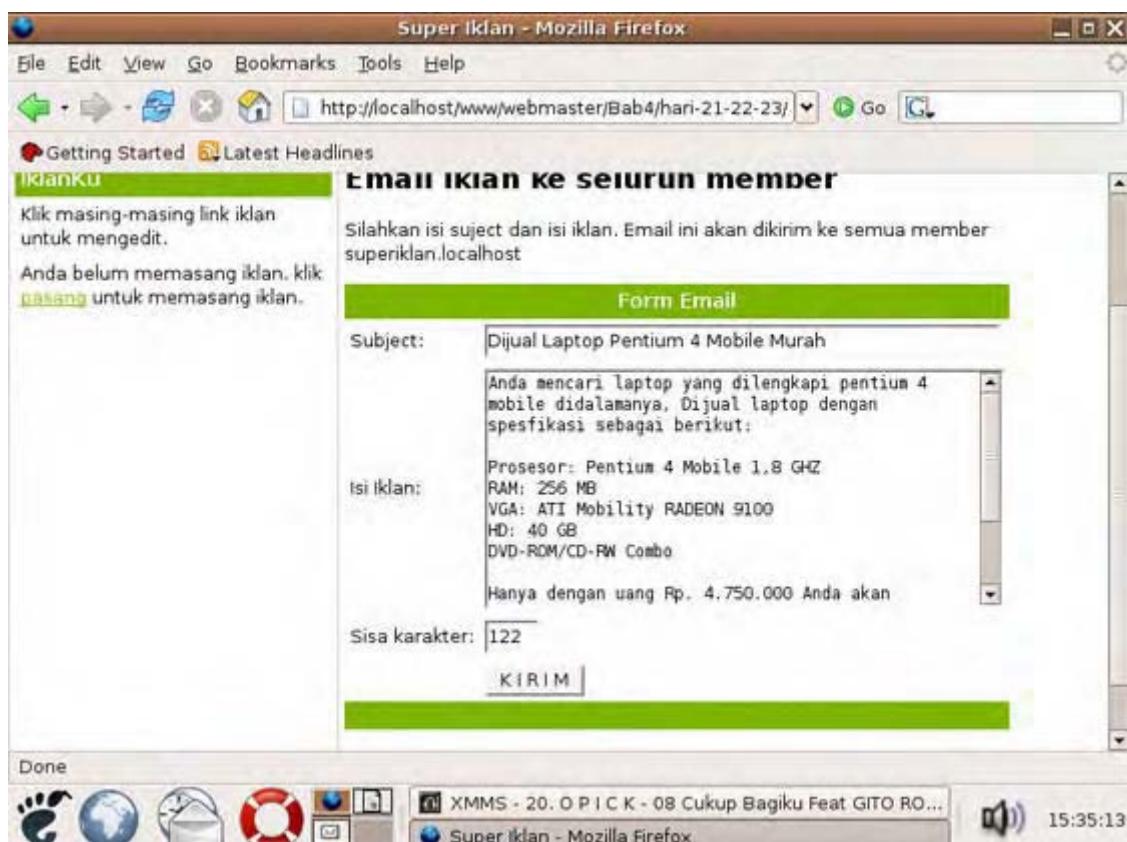
```

$waktu->set_mode(2); // mode tetap
$skr = $waktu->set_tgl(1); // sekarang dalam detik

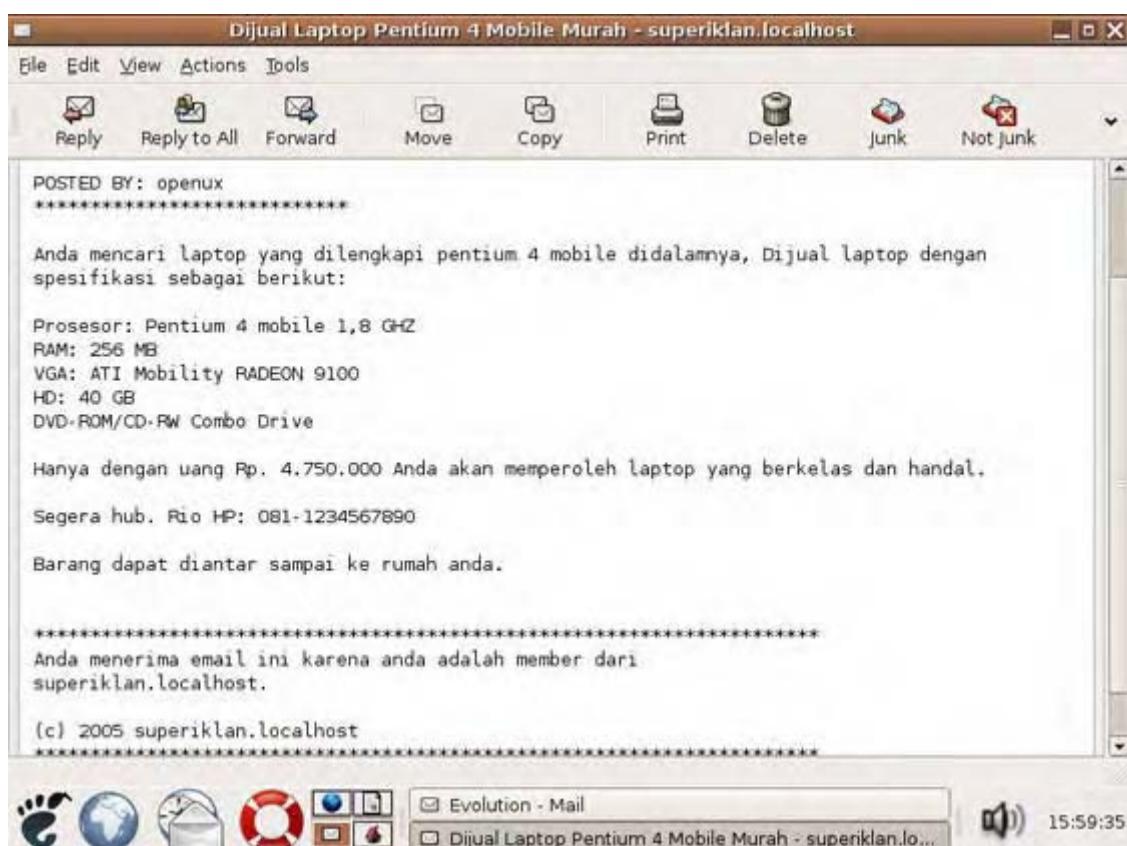
```

Isi dari \$skr merupakan tanggal sekarang dalam format detik. Setelah kita mendapatkan nilai dari sekarang dalam detik maka kita dapat membandingkan nilainya dengan nilai dari next\_post yang ada ditabel. Jika nilai \$skr sama atau lebih besar dari next\_post maka kirim email.

```
if ($skr >= $data['next_post']) // sudah melewati 7 hari
```



Gambar 4.40: Form kirim email iklan pada member area



Gambar 4.41: Email diterima oleh member

Untuk mengirim email keseluruhan member selain user yang mengirim email itu sendiri, sebelumnya kita lakukan query seperti berikut.

```
$query = mysql_query("UPDATE tb_email SET next_post=$next WHERE
username='\$user'");
```

Setelah mengirim email baru kita mengupdate isi dari next\_post 7 hari kedepan namun dalam detik. Untuk itu kita panggil buat kembali objek waktu.

```
$waktu->set_mode(1); // mode penambahan
$next = $waktu->set_tgl(1, $lama_email); // dalam detik, tujuh hari ke depan
```

Isi dari \$next merupakan tanggal 7 hari kedepan dalam detik. Setelah itu baru kita update isi dari next\_post menggunakan query UPDATE.

Namun jika nilai dari \$skr masih kurang dari \$data['next\_post'] maka tampilkan pesan bahwa ia belum bisa mengirim email karena belum melewati 7 hari.

```
$nanti = floor(($data['next_post'] - $skr) / 60 / 60 / 24);
```

Kita dapat menggunakan fungsi matematik floor() untuk melakukan perhitungan. Setelah Untuk mengubah hasilnya menjadi dalam hari. Kita perlu membagi hasilnya dengan 60(detik), 60 (menit), dan 24(jam).

File yang terakhir untuk direktori ini sekaligus yang terakhir untuk proyek superiklan.localhost adalah *edit.php*. File ini berfungsi mengedit iklan yang sudah dipost user. Buat file PHP baru pada PHP DESIGNER 2005 kemudian ketik kode berikut.

#### Script dari file *edit.php*

---

```
<?php

/*
** Nama File.....: edit.php
** Penulis.....: Rio Astamal
** Tanggal.....: 17-10-2005
** Penjelasan....: Untuk mengedit iklan yang terpasang
****/
```

```
session_start();

include ('../inc/class_skin.php');
include ('../template/member_var.php');

$proses = $_GET['proses'];
if ($proses == '')
 $proses = 'form';

$proses = filter_str($proses);

konek_db(); // koneksikan ke MYSQL server

// cek user apakah sudah login atau belum
if (!cek_session('member'))
 $edit = $not_login;
else
{

// handel setiap proses dengan case dan switch
switch ($proses)
{
 /*
 case 'form':
 */
```

```

// ambil data dari URL
$kat = filter_str($_GET['kat']);
$id = filter_str($_GET['id']);

// lakukan query untuk mendapatkan informasi iklan
$hasil = mysql_query("SELECT * FROM tb_iklan WHERE id_iklan='$id'");
$data = mysql_fetch_array($hasil);

$jdl = stripslashes($data['jdl_iklan']); // judul iklan
$isi = stripslashes($data['isi_iklan']); // isi iklan

$judul = "<h2>Edit Iklan</h2>\n"; // header judul

// variabel untuk menampilkan form edit
$edit = "

<p>Pada halaman ini anda dapat melakukan pengeditan pada iklan anda. Disini anda juga dapat menghapus iklan anda. Untuk memasukkan perubahan klik UPDATE, sedangkan untuk menghapus klik HAPUS.</p>

$java

<form action='edit.php?proses=proses_form' method='post'>
<input type='hidden' name='id' value='$id'>
<table border='0' cellpadding='4'>
<tr bgcolor='#7cb500' align='center'>
 <td class='putih' colspan='2'>Form Edit</td></tr>
<tr> \n<td>Kategori: </td><td>$kat</td></tr>
<tr> \n<td>Judul Iklan: </td>
 <td><input type='text' name='jdl_iklan' value='$jdl' size='50'></td></tr>
<tr> \n<td>Isi Iklan: </td>
 <td><textarea name='isi' rows='10' cols='50' onkeyup='batas_kar(this.form)'>$isi</textarea></td></tr>
<tr> \n<td>Sisa Karakter: </td>
 <td><input type='text' name='sisa' value='500' readonly='yes' size='3'></td></tr>
<tr><td><input type='submit' value='UPDATE'></form></td>
 <td><form action='edit.php?proses=hapus' method='post'>
 <input type='hidden' name='id' value='$id'>
 <input type='submit' value='HAPUS'></td></tr>
<tr height='20' bgcolor='#7cb500'><td colspan='2'></td></tr>
</table>

"; // akhir dari variabel $edit

break;
/************/
/* **** */
case 'proses_form':
/************/

$jdl = addslashes($_POST['jdl_iklan']);
$isi = addslashes($_POST['isi']);

$id = $_POST['id'];

// cek field
if (!cek_field($_POST))
 $edit = "Error: Masih ada field yang kosong.
\n$kembali";
else
{
 // update isi iklan

```

```

$hasil = mysql_query("UPDATE tb_iklan SET jdl_iklan='$jdl', isi_iklan='$isi'
WHERE id_iklan=$id");
// cek status
if (!$hasil)
 $edit = "Error: Gagal mengupdate iklan.
\n$kembali";
else
 $edit = "Iklan berhasil diupdate.
\n$kembali";
}

break;
***** */

***** */
case 'hapus':
***** */

$id = $_POST['id']; // dapatka id iklan yang akan dihapus

// lakukan query DELETE untuk menghapus
$hasil = mysql_query("DELETE FROM tb_iklan WHERE id_iklan=$id");
// cek status
if (!$hasil)
 $edit = "<p>Error: Gagal menghapus iklan yang ber-id: $id.
$kembali</p>\n";
else
 $edit = "<p>Iklan dengan id: $id berhasil dihapus
$kembali
</p>\n";
} // akhir dari switch
} // akhir dari else

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/skin_utama.php');
$skin->ganti_tag('{SEKARANG}', $tgl);
$skin->ganti_tag('{JUDUL}', $judul);
$skin->ganti_tag('{UTAMA}', $edit);
$skin->ganti_tag('{MENU}', $mem_menu);
$skin->ganti_tag('{SISI1}', $iklanku);
$skin->ganti_tag('{SISI2}', '');
$skin->ganti_tampilan();

?>

```

---

Simpan pada direktori **iklan\member** dengan nama ***edit.php***. Jika anda mengklik judul iklan yang ada pada sisi kiri halaman pada file edit.php yang akan dipanggil.

\*\*\*\*\*

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

## Berikutnya

Pada bab berikutnya kita akan mempelajari bagaimana menciptakan produk untuk dijual di internet. Dan bagaimana membuat website reseller yang "menjual" seperti rahasia-webmaster.com.

# Bab

5

- Memulai Bisnis di Internet
- Web Design
- Membuat Website Reseller

Sebelum melanjutkan materi, ikuti langkah-langkah berikut

- Masuk pada direktori C:\Apache2\htdocs\webmaster
- Buat folder, namai folder tersebut **Bab5**

\*\*\*\*\*

Karena kita sudah menginjak hari baru maka ikuti langkah-langkah berikut:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab5
- Buat folder baru dan namai folder tersebut **hari-25**

## Memulai Bisnis di Internet

Materi kita kali ini sama sekali tidak berhubungan dengan programming. Melainkan kita akan membahas bagaimana menghasilkan **uang** melalui internet. Materi ini tidak kalah menariknya dengan bermain kode-kode PHP. Pada bahasan ini anda akan menemukan bagaimana memulai bisnis di internet, seperti apa konsep membangun bisnis internet dan sebagainya.

Seperti bisnis-bisnis model lainnya. Bisnis diinternet tidak jauh berbeda, anda harus memiliki **produk** untuk dijual. Tanpa adanya produk sama halnya anda tidak memiliki apa-apa untuk dijual ke konsumen.

Model produk anda haruslah berbeda dengan produk yang dijual secara konvensional atau offline. Agar produk anda laku di pasaran, produk anda harus bisa menjawab atau memecahkan masalah konsumen.

Jenis produk yang cocok untuk dijual di internet adalah produk yang **kasat mata(digital)** artinya produk tersebut tidak tersedia secara offline. Contoh-contoh produk yang dimaksud antara lain:

- Website Membership (seperti rahasia-webmaster.com)
- Software komputer
- E-book (seperti yang anda baca)
- Mailing List
- Layanan iklan
- Servis web hosting
- dan sebagainya

Namun yang pasti harus anda miliki adalah **website membership**, kemudian baru produk lainnya. Karena dengan memiliki website membership kita dapat menjadikan member kita sebagai reseller yang menjadi tenaga penjual kita. Dengan memiliki reseller usaha promosi kita akan jauh lebih ringan dan mudah.

Selanjutnya kita akan membahas bagaimana menciptakan suatu produk untuk dijual.

## Menciptakan Produk untuk Dijual

Ketika kita bicara bagaimana membuat produk. Maka kita juga harus berpikir juga produk apa yang sebaiknya kita buat. Apakah produk ini dapat diterima konsumen? Apakah banyak orang yang mencarinya?

Itu adalah beberapa point yang harus anda perhatikan ketika akan menciptakan sebuah produk. Mengapa kita menjual produk digital bukan produk konvesional?. Pertanyaan yang bagus. Produk digital memiliki banyak kelebihan diantaranya:

- Produk digital dapat didistribusikan dengan cepat karena melalui internet.
- Anda dapat mengantongi 100% keuntungan untuk diri anda sendiri.
- Anda tidak perlu biaya tambahan untuk membuat salinan produk anda.
- Biaya untuk membuatnya sangat kecil dibanding produk konvensional.

- Dan masih banyak lagi...

Jika anda seorang programmer anda dapat membuat software untuk dijual. Jika anda seorang penulis anda dapat menulis sesuatu untuk dijual. Namun sebelumnya jawab dulu pertanyaan berikut.

## Informasi apa yang dibutuhkan orang saat ini?

Jika anda dapat menjawab pertanyaan diatas, SELAMAT. Anda telah memiliki produk yang akan dibeli oleh konsumen berapa pun harganya. Namun jika anda tidak bisa menjawabnya, berhentilah dulu. Jika anda menyuruh saya untuk menjawab pertanyaan diatas. Tentu jawaban saya adalah "Bagaimana menghasilkan uang...".

Uang adalah suatu kebutuhan universal yang dibutuhkan oleh setiap orang. Apapun produk anda nantinya. Jika produk anda tersebut terdapat informasi tentang ***uang***. Secara otomatis produk anda memiliki nilai ***plus*** dimata konsumen.

Informasi yang sering dicari oleh orang adalah informasi yang bersifat "bagaimana ..." atau "How to ...". Informasi semacam ini lebih banyak mengundang perhatian orang daripada informasi yang bersifat berita.

Contoh informasi yang tidak cocok untuk digunakan produk adalah

- Bagaimana menggunakan internet?
- Komunikasi lewat email

Namun bagaimana jika kita ubah menjadi seperti berikut:

- Bagaimana menghasilkan uang melalui internet?
- Bagaimana memanfaatkan email untuk menghasilkan jutaan rupiah?

Respon dari konsumen akan meningkat jika kita mengubah informasinya seperti yang kedua. Jadi kesimpulannya, apapun produk anda nantinya tambahkan konsep tentang ***bagaimana menghasilkan uang*** didalamnya.

Memang anda tidak harus menambahkan konsep uang pada produk anda. Namun nilai dari produk anda saya jamin meningkat jika anda menambahkan *ramuan rahasia* yaitu ***uang***.

Informasi apapun yang ada pada benak anda saat ini. Tuangkan saja pada segelas kopi..., lho kok!, maksud saya tuangkan ide anda tersebut pada kertas atau ketikkan pada komputer. Apapun informasi anda ketika sudah berhubungan dengan "bagaimana...", pasti informasi itu akan menarik.

## Menentukan Informasi yang Tepat

Jika anda benar-benar ingin sukses, anda perlu memperhatikan point-point berikut dalam menentukan produk yang akan anda jual.

- Apa yang mereka inginkan?
- Apa masalah yang sering mereka hadapi?
- Apa yang anda punya sehingga dapat membantu mereka?

Jika anda dapat menjawab pertanyaan-pertanyaan tersebut kepada mereka. Mereka akan dengan senang hati membeli produk anda.

## Mulai Menulis produk

OK, saya asumsikan anda telah memiliki ide dan akan mengembangkan ide tersebut untuk menjadi sebuah produk. Kemudian muncul pertanyaan, saya memang telah memiliki ide namun saya tidak terlalu ahli dalam bidang itu? Itu sih masalah gampang. Ada dua akternatif untuk mengatasi masalah anda:

1. Menggunakan jasa orang lain yang ahli dalam bidangnya
2. Memanfaatkan internet sebagai daftar pustaka

Untuk yang pertama yaitu menggunakan jasa orang lain. Anda hanya perlu mengajukan pertanyaan-pertanyaan yang sesuai dengan keahlian orang itu. Dan sesuai dengan produk anda tentunya.

Setelah semua data yang anda perlukan sudah terkumpul. Saatnya menyatukan semuanya ke dalam satu ke satuan.

Cara yang kedua adalah dengan memanfaatkan internet. Pemanfaatan internet disini adalah kita mengumpulkan segala informasi yang kita perlukan dengan bantuan search engine, google misalnya. Langkah-langkah yang harus anda lakukan adalah

- Buka situs search engine
- Ketikkan kata kunci yang anda inginkan (misalnya "internet business")
- Seluruh informasi akan muncul
- Kumpulkan data dengan melakukan browsing ke masing-masing link
- Setelah semua informasi anda dapatkan atur susunannya
- Buatlah bab-bab agar susunan tulisan anda terstruktur

Dengan cara diatas anda akan mendapatkan informasi dengan cepat dan tanpa perlu memikirkan acara tulis menulis.

## **Menentukan Judul E-book**

Judul merupakan salah satu hal yang sangat penting dalam menentukan kesuksesan produk anda.

Kegagalan produk anda juga dapat terletak pada judul e-book anda. Jika judul e-book anda tidak menarik bagaimana kita bisa menarik minat orang. Untuk itu anda perlu berpikir ekstra dalam menentukan judul e-book anda.

Sebagai contoh sewaktu saya menulis e-book ini saya memiliki beberapa pilihan judul diantaranya:

- Web Programming Dengan PHP
- Panduan Menjadi Webmaster
- Mastering PHP 5
- Menjadi Webmaster dalam 30 Hari

Sekarang saya bertanya kepada anda, manakah judul yang menurut anda paling menarik?. Silahkan putuskan sendiri.

Kalaupun isi e-book anda sangat menarik namun jika judulnya tidak menarik. Kemungkinan besar e-book anda tidak akan terbaca.

## **Membuat file E-book (PDF)**

Setelah produk anda selesai anda susun. Langkah terakhir adalah mengubahnya menjadi buku digital (e-book). File e-book biasanya berekstensi \*.pdf (portable document format).

Dan untuk membacanya kita memerlukan PDF viewer. Untuk lingkungan windows yang paling

banyak digunakan adalah Adobe(R) Acrobat(R) Reader. Sedangkan untuk lingkungan Linux kita dapat menggunakan Xpdf yang biasanya sudah dibundel disebagian besar distro linux.

Itu untuk membacanya, sedangkan untuk membuatnya bagaimana?. Untuk membuat file berformat PDF. Kita perlu Adobe(R) Acrobat(R), bukan reader. Harga software ini sangat mahal untuk ukuran kita, mencapai ratusan dollar. Wah... , kalau begitu nggak jadi bikin e-book nih!. Tunggu dulu, dalam dunia software terdapat dua kubu software yang berbayar (shareware) atau software yang gratis (freeware).

Jika anda ingin membuat file PDF tanpa ingin mengeluarkan uang sepeserpun. Anda dapat menggunakan software **PrimoPDF** (<http://www.primopdf.com/>). PrimoPDF sangat bagus dalam menghasilkan file PDF. Hasilnya pun tidak kalah dibanding Adobe(R) Acrobat(R).

Setelah anda mendownload PrimoPDF. Install software tersebut, kelebihan dari software ini adalah anda dapat membuat file PDF dari segala macam aplikasi. Ini dikarenakan PrimoPDF berfungsi sebagai **printer virtual**.

Untuk menghasilkan file PDF anda cukup klik **File > Print** pada aplikasi tertentu. Kemudian pada pilihan printer pilih **PrimoPDF**. Tentukan letak penyimpanan file. Setelah itu klik tombol **Save**. File PDF pun berhasil dibuat.

Anda juga dapat menentukan resolusi cetak dengan mengklik link properties pada saat akan mencetak. Selain itu anda dapat memproteksi document dengan password agar tidak dapat dibuka oleh sembarang orang.

Yang namanya program gratis pasti memiliki keterbatasan. Salah satu kelemahan PrimoPDF adalah kita tidak bisa memberi bookmark atau komentar pada file PDF, dan sebagainya yang bersifat editing. Sehingga jika e-book kita terdiri dari banyak halaman. User akan cukup sulit dalam berpindah antar bab karena pada file PDF kita tidak terdapat bookmark.

## Mengotomatiskan Bisnis Anda

Mengotomatiskan disini bukan berarti kita membeli sebuah mesin atau suatu alat yang berupa hardware dan sebagainya. Yang dimaksud mengotomatiskan disini adalah adalah sebuah **website**.

Ya sebuah website, dari bab 1 sampai bab 4 yang lalu. Anda telah memiliki kemampuan **programming**. Sekarang saatnya **memanfaatkan** kemampuan anda tersebut untuk menghasilkan uang.

Mungkin dalam pikiran anda saat ini anda sudah dapat membayangkan bagaimana cara membuatnya. Dalam hal teknis pembuatan saya kira nantinya tidak ada masalah. Namun kita akan mempraktekkannya pada bahasan selanjutnya.

Untuk kali ini kita akan membahas bagaimana membuat website yang secara otomatis bekerja untuk anda 24 jam X 7 hari tanpa henti. Website otomatis yang akan kita buat meliputi:

- Sistem Presentasi Otomatis
- Sistem Pelayanan Otomatis
- Sistem Follow Up Otomatis
- Sistem Reseller Otomatis

## Sistem Presentasi Otomatis

Bagian ini adalah bagian dari website kita yang mempresentasikan produk anda. Pada bagian ini yang paling harus anda perhatikan adalah headline/judul website dan sales letter.

Dalam beberapa bagian pada penjelasan ini, kita menggunakan [rahasia-webmaster.com](http://rahasia-webmaster.com)

sebagai studi kasus.

## Headline / Judul

Mungkin banyak dari pengunjung anda adalah orang yang sebenarnya datang tanpa maksud. Mereka mungkin kebetulan surfing dan secara tidak sengaja "terdampar" di website anda.

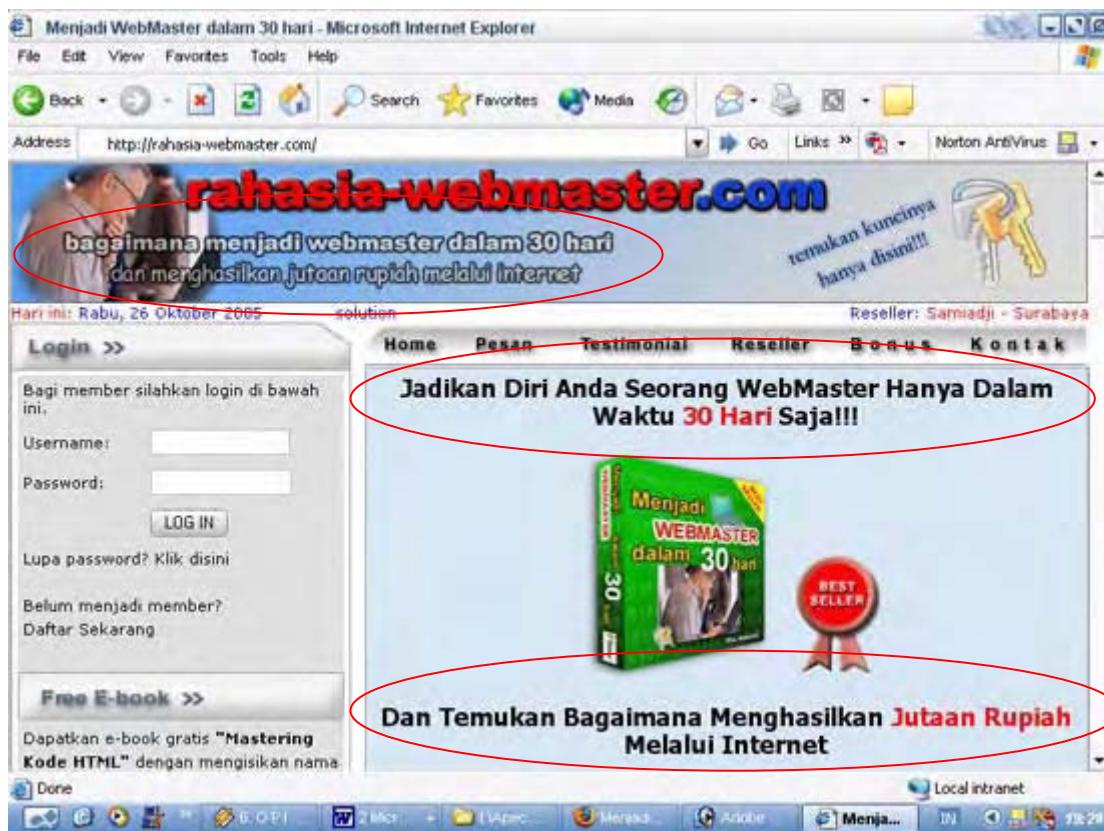
Tidak peduli apapun tujuan mereka ketika mereka sudah ada pada website kita. Maka mereka adalah calon pembeli produk kita. Oleh karena itu, kita harus mendapatkan perhatian lebih dari mereka.

Salah satu cara untuk menarik perhatian mereka adalah dengan membuat headline website yang menarik. Kita hanya memeliki beberapa detik untuk menangkap perhatian mereka. Jika mereka tidak tertarik maka boleh dibilang headline anda gagal.

Buatlah pengunjung website anda setidaknya berkata, "Hmm..., website ini kelihatannya menarik"

Ketika kita sudah mulai mendapatkan perhatian mereka. Maka kita mempunyai kesempatan untuk mempresentasikan produk kita lebih lanjut. Sehingga akhirnya mereka mau membeli produk kita.

Headline kita harus dapat mencerminkan manfaat produk kita. Dengan demikian pengunjung website anda akan lebih berlama-lama di website anda. Lihat contoh headline berikut.



Gambar 5.0: Headline rahasia-webmaster.com

Gambar diatas menunjukkan headline yang saya tulis untuk website rahasia-webmaster.com. Saya rasa headline tersebut yang membawa anda untuk membeli e-book ini, betul?.

Sebenarnya jika dikategorikan ada beberapa jenis headline diantaranya:

- Headline "Bagaimana" / "How To"
- Headline Pertanyaan

- Headline Berita

Jika anda perhatikan pada rahasia-webmaster.com saya menggunakan dua tipe headline yaitu tipe "Bagaimana" (Bagaimana Menjadi Webmaster dalam 30 Hari dan Menghasilkan Jutaan Rupiah dari Internet). Dan yang kedua yaitu tipe Berita ("Jadikan Diri Anda Seorang Webmaster hanya dalam waktu 30 hari Saja!!!").

Saya rasa anda sudah tahu keampuhan headline yang saya tulis. Mau buktinya? Ya..., yang sekarang anda baca adalah buktinya.

## **Sales Letter**

Setelah tugas penarik perhatian yang dilakukan oleh headline berhasil. Sekarang giliran sales letter yang melanjutkan presentasi produk kita. **Sales letter** merupakan sistem yang berupa tulisan pada website yang bertugas untuk mempresentasikan produk anda kepada pengunjung sebaik mungkin sehingga mereka memutuskan untuk membeli produk anda.

Karena sales letter bertugas mempresentasikan produk kita. Maka kita harus mendesainnya untuk sebaik mungkin dapat menarik hati konsumen. Sehingga akhirnya mereka mau memutuskan untuk membeli produk kita.

Ingat!, sebagus apapun produk anda nantinya, jika sales letter anda "payah" maka konsumen tidak akan pernah tahu manfaat yang diperoleh dari produk anda. Oleh karena itu jika anda menulis sales letter, berpikirlah dari sudut pandang konsumen. Karena dengan begitu mereka akan tahu manfaat terbesar setelah memiliki produk anda.

Sales letter menjadi penentu seberapa banyak pengunjung yang akan membeli produk anda dari 1000 orang pengunjung misalnya. Oleh karena itu sales letter harus kita desain agar menampilkan manfaat terbesar yang mereka peroleh setelah memiliki produk anda.

Dalam sales letter kita dapat membaginya pada beberapa bagian diantaranya:

- Pembukaan
- Manfaat
- Kredibilitas
- Harga
- Ciptakan kondisi
- Pemesanan
- Garansi
- Bonus
- Penutupan

### **Pembukaan**

Setelah anda mendapatkan perhatian mereka. Maka anda harus tetap menjaga minat mereka untuk tetap membaca sales letter anda. Pada pembukaan ini anda dapat menuliskan secara sekilas manfaat terbesar yang mereka dapatkan setelah memiliki produk anda.

Dalam beberapa paragraf awal anda dapat menyampaikan pada pengunjung bahwa anda tahu cara memecahkan masalah mereka. Atau anda pernah dalam posisi mereka.

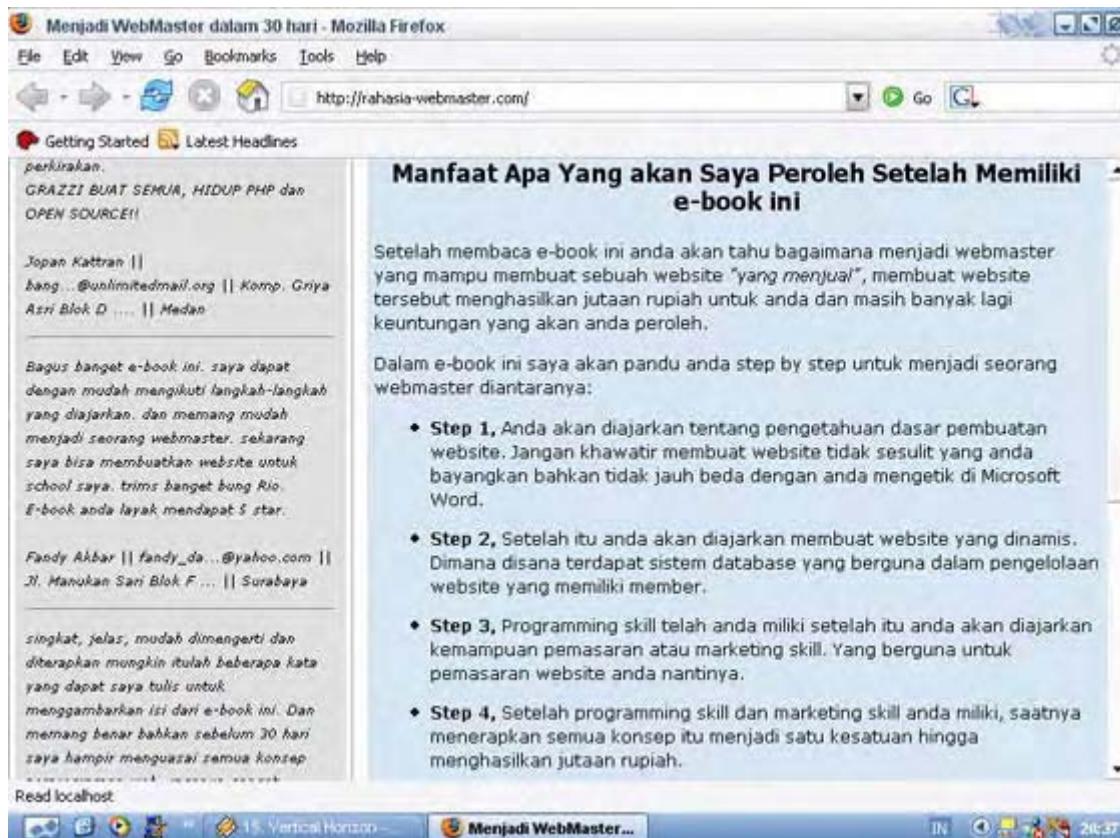
Namun sekarang kehidupan anda telah berubah. Dan dengan memiliki produk anda yakinkan mereka bahwa mereka akan mengalami hal yang sama dengan anda yaitu **kesuksesan**.

Dengan demikian rasa penasaran mereka untuk terus mengetahui produk anda semakin lama-lama membara. Jika sudah dalam keadaan seperti ini kita perlu memberitahukan manfaat apa saja yang akan mereka peroleh setelah memiliki produk anda.

## Daftar Manfaat

Setelah anda sudah menciptakan rasa keingintahuan yang lebih besar dari konsumen. Saatnya memberikan daftar manfaat yang akan mereka peroleh. Untuk mempermudah pengunjung dalam memahami manfaat produk anda.

Desainlah daftar manfaat tersebut pada setiap baris berbeda. Atau akan lebih baik jika anda menggunakan bullet list. Tag HTML yang dapat kita gunakan untuk bullet list adalah tag <ul></ul> dan <li></li>.



Gambar 5.1: Daftar manfaat pada rahasia-webmaster.com

Manfaat yang anda tulis juga sangat menentukan keputusan pengunjung dalam membeli produk anda. Jadi pastikan daftar manfaat terbaik yang anda tulis dalam daftar manfaat.

## Kredibilitas

Setelah kita mendapatkan perhatian mereka, kemudian lebih kita perkuat lagi dengan membeberi daftar manfaat produk kita pada mereka. Langkah selanjutnya adalah dengan menunjukkan beberapa hal yang berhubungan dengan kredibilitas.

Kita semua tahu jika kita membeli dari produsen terkemuka yang kredibilitasnya sudah jelas. Tentu produk dari mereka lebih dipercaya konsumen ketimbang produk dari produsen yang tidak jelas kredibilitasnya.

Kita dapat menunjukkan kredibilitas dengan memberikan informasi mengapa kita ahli dalam bidang tersebut. Jangan tutupi diri anda dengan tidak mencantumkan profil data diri anda dengan lengkap. Karena dengan mencantumkan profil diri anda secara lengkap. Kesan negatif terhadap anda dapat dihilangkan.

Selain itu cara yang paling efektif adalah dengan menunjukkan kepuasan (testimonial) konsumen terdahulu. Dengan demikian mereka dapat percaya kalau produk anda memang benar-benar berhasil.

## Harga

Ini bagian yang juga jangan anda lupakan. Masa, anda menulis panjang lebar dalam sales letter anda namun anda tidak mencantuk harga, kan tidak lucu.

Bisa-bisa pengunjung bertanya, "Ini produk dijual apa tidak sih?".

Dalam mencantuk harga anda harus memperhatikan beberapa hal. Diantaranya anda harus dapat membuat seolah-olah harga produk anda tidak pantas dihargai dengan harga tersebut karena manfaatnya begitu besar.

Misalnya anda memiliki sebuah produk yang sangat berkualitas. Dan seharusnya harganya mencapai jutaan. Namun anda hanya menjualnya seharga 100 ribu. Buatlah pengunjung merasa bahwa ia beruntung menemukan sebuah produk yang berkualitas namun dengan harga yang rendah.

## Ciptakan Kondisi

Setelah pengunjung mengetahui harga produk anda. Maka sekali lagi harus kita yakinkan agar segera membeli produk kita. Bagaimana caranya?

Caranya adalah dengan menciptakan kondisi ***keterburu-buruan***. Ya dengan menciptakan kondisi tersebut calon pembeli produk kita akan segera melakukan tindakan. Yaitu melakukan pemesanan produk kita dan akhirnya membeli.

Salah satu contoh untuk menciptakan rasa keterburu-buruan adalah dengan menggunakan ***deadline***. Lihat contoh berikut.

**Jika anda memesan sebelum hari Rabu, 19 Oktober 2005, maka dapatkan juga 8 SUPER BONUS!!!**

Contoh deadline diatas adalah deadline yang saya gunakan pada sales letter rahasia-webmaster.com. Saya yakin deadline diatas pasti memiliki pengaruh pada keputusan anda.

Untuk menciptakan deadline diatas kita sudah pernah membuatnya. Kita hanya memerlukan beberapa baris kode PHP untuk menciptakan deadline seperti diatas. Namun dengan beberapa baris kode tersebut, jutaan rupiah sudah dapat saya kantongi.

## Pemesanan

Pada bagian ini kita dapat beranggapan bahwa pengunjung telah memiliki minat dan mereka hendak membeli produk kita. Maka kita harus menunjukkan kepada mereka dengan jelas tata cara pemesanan produk kita.

Anda harus menyediakan jalan bagi mereka untuk menuju proses pemesanan. Dalam hal ini kita dapat menyediakan link pemesanan.

Dalam membuat link pemesanan pada sales letter anda. Buatlah link tersebut tampak jelas dengan cara memperbesar tulisan dan sebagainya. Dengan demikian link tersebut akan mendapat perhatian lebih dari pengunjung.

Letak dari link-link yang merujuk ke pemesanan juga harus anda perhatikan letaknya. Misalnya setelah daftar manfaat atau harga e-book anda. Di bawahnya anda beri link pemesanan seperti,

**PESAN SEKARANG JUGA!**

Dengan demikian pengunjung dapat langsung bereaksi melakukan pemesanan. Karena telah mengetahui daftar manfaat produk anda.

## **Garansi**

Jika kita berbicara tentang pembelian suatu produk. Maka faktor garansi jangan pernah anda abaikan. Membeli sesuatu semuanya terdapat resiko. Entah itu secara offline lebih-lebih secara online.

Untuk menghilangkan rasa ketakutan dan meningkatkan rasa keyakinan pengunjung kita wajib memberikan garansi.

Mungkin dalam pikiran anda, "Bagaimana jika mereka meminta pengembalian produk?". Pertanyaan diatas terletak pada seberapa bagus produk anda.

Dan saya yakin jika memang produk anda berkualitas dan memang bermanfaat untuk mereka. Mereka tidak akan meminta kembali uangnya.

Waktu garansi yang dapat anda berikan juga bervariasi. Namun saya sarankan jangan memberikan waktu garansi terlalu pendek semisal 7 hari atau 30 hari. Semakin pendek waktu garansi produk anda. Maka akan semakin membuat konsumen ingin cepat-cepat mencoba produk anda.

Jika mereka merasa tidak puas terhadap produk anda. Maka mereka dengan segera meminta kembali uangnya.

Saya sarankan anda untuk memberikan waktu garansi yang cukup lama. Misalnya 6 bulan, 8 bulan atau 1 tahun atau bahkan lebih. Dengan begitu mereka tidak akan terburu-buru mencoba produk anda. Dan kemudian jika mereka tidak puas, biasanya mereka menunda permintaan garansi mereka. Dan sudah wajar kalau akhirnya mereka lupa.

## **Bonus**

Mungkin bonus menjadi salah satu hal yang penting dalam peran serta mempengaruhi keputusan pengunjung untuk membeli produk kita. Karena dengan menambahkan bonus dapat meningkatkan minat calon pembeli kita menjadi berlipat-lipat.

Anda dapat memberikan bonus berupa e-book gratis, produk berlisensi hak jual ulang, atau sebuah servis. Seperti pada rahasia-webmaster.com yaitu servis iklan gratis dan gratis update seumur hidup. Atau bahasa kerennya LIFE TIME.

Saya rasa anda sudah tahu pengaruh bonus terhadap produk utama anda. Jika belum ingat-ingat kembali waktu anda belum memiliki e-book ini. Dengan adanya bonus seperti itu keinginan anda untuk membeli akan semakin besar, betul?.

Namun perlu anda ingat, jangan pernah menjual produk yang memiliki lisensi hak cipta. Seperti program-program shareware. Jika anda menyertakan suatu program shareware dalam bonus anda.

Entah itu masih trial atau sudah ***dicrack***. Anda dapat dikenai sanksi berupa pidana. Karena anda dianggap telah melanggar undang-undang hak cipta orang lain.

Jadi jika ada seseorang yang memikat hati konsumen dengan embel-embel software yang jelas-jelas berlisensi hak cipta. Maka pengetahuannya akan dunia bisnis ataupun TI ***perlu*** dipertanyakan.

## **Penutupan**

Pada bagian yang terakhir adalah penutupan. Pada bagian ini tegaskan sekali lagi bahwa

produk anda mempunyai manfaat yang begitu besar.

Juga Ingatkan pada pengunjung bahwa penawaran seperti ini memiliki keterbatasan waktu. Sehingga mereka harus cepat memutuskan untuk bertindak. Nah, setelah itu kita tinggal menunggu mereka melakukan transfer. Rajin-rajin berdoa OK.

## Sistem Pelayanan Otomatis

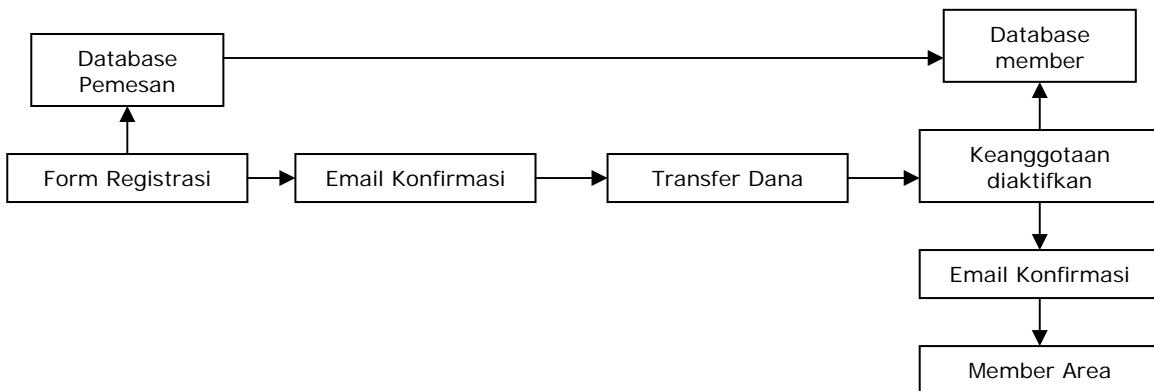
Pelayanan otomatis? Ini menggunakan robot atau apa. Sudah dibilang dari tadi, maksudnya otomatis disini adalah website yang kita bangun menggunakan script PHP. Kalau mau beli robot, mesti jual rumah dulu dong?!!.

Pada bagian ini website kita desain sedemikian rupa sehingga proses mulai dari registrasi, database dan pengiriman email berjalan secara otomatis. Tugas kita hanyalah mengaktifkan keanggotaan user yang telah melakukan transfer dana ke rekening kita.

Skenario untuk sistem pelayanan otomatis adalah sebagai berikut:

1. User telah memutuskan untuk memesan produk anda
2. User mengisi form registrasi yang telah disediakan
3. User menerima email konfirmasi pendaftaran
4. User melakukan transfer ke pengelola dan reseller
5. Keanggotaan user diaktifkan oleh pengelola
6. User menerima email pemberitahuan bahwa keanggotaannya telah diaktifkan
7. User masuk ke member area dan mendownload produk utama

Setelah anda mengatahui alur pelayanan otomatis. Maka dengan sedikit gambaran kita dapat menciptakan sketsa alur script yang akan kita buat.



Gambar 5.2: Flowchart sistem pelayanan otomatis

### Penjelasan Flowchart(Diagaram)

Dari flowchart diatas dapat kita lihat. Bahwa dalam sistem pelayanan ini kita harus menyediakan dua tabel. Satu untuk pemesan dan satu member.

Ketika user mensubmit form registrasi. Maka secara profil dari user tersebut akan kita simpan pada tabel pemesan. Dan secara otomatis sebuah email yang berisi konfirmasi pendaftaran akan terkirim ke alamat email user tersebut.

Email konfirmasi yang terkirim ke alamat email user berisi data transfer rekening pengelola dan reseller lengkap dengan alamat email dan nomor telpon reseller.

Setelah user melakukan transfer data ke alamat rekening pengelola dan resellernya. Maka sekarang tugas anda sebagai pengelola hanyalah mengecek kecocokan dana yang ditransfer.

Jika dana tersebut cocok dengan yang ada didatabase. Anda tinggal menekan tombol untuk melakukan aktivasi terhadap user tersebut.

Setelah anda mengaktifkan user tersebut. Secara otomatis sebuah email berisi pemberitahuan bahwa keanggotaannya telah diaktifkan. Dengan demikian ia dapat login ke member area untuk mendownload produk.

Data user tersebut pada tabel pemesan sekarang sudah dihapus. Dan dipindah ke tabel member.

Selain melakukan aktivasi user. Anda sebagai pengelola juga dapat memblokir keanggotaan seorang member. Anda mungkin menemukan seorang member yang ternyata belum mentransfer dana namun sudah diaktifasi. Nah, dengan adanya fasilitas blokir member tersebut. Kita dapat memblokir keanggotaannya sehingga ia tidak dapat login.

Kemungkinan lain adalah jika suatu saat user meminta garansi. Dan kemudian anda sudah mengembalikan uangnya. Maka anda dapat melakukan pemblokiran atau menghapus permanen user tersebut dari database.

## **Sistem Follow-Up Otomatis**

Seringkali kita mengunjungi suatu website dan kadang kita tidak pernah kembali ke website tersebut. Kenyataan seperti diatas jangan sampai terjadi pada website anda. Dimana seorang pengunjung yang semestinya membeli produk kita. Ternyata hanya terlewat begitu saja.

Untuk itu kita memerlukan suatu metode agar kita tetap dapat mengontak user tersebut. Meskipun ia belum memutuskan untuk membeli.

### **Caranya?**

Satu-satunya cara yang harus kita lakukan adalah dengan menangkap email dari user tersebut. Kita memerlukan alamat email ini karena kita akan terus melakukan follow-up pada user ini sampai ia memutuskan untuk membeli.

Follow-up merupakan suatu cara untuk tetap berhubungan dengan prospek kita sampai ia memutuskan untuk membeli produk kita. Follow-up disini tentunya adalah dengan menggunakan media email.

### **Menangkap Email Pengunjung**

Salah satu cara yang efektif untuk mendapatkan email dari user adalah dengan memberikan sesuatu yang gratis untuk mereka. Itu bisa berupa e-book atau software.

Anda tentu ingat ketika mengunjungi [rahasia-webmaster.com](http://rahasia-webmaster.com). Disamping kiri halaman selalu terdapat link download e-book gratis.

Namun untuk mendapatkannya user harus memasukkan nama dan email mereka. Dengan demikian secara otomatis alamat email user berada dalam target follow-up anda.

Sistem follow-up sendiri dapat kita bagi menjadi beberapa bagian yaitu:

- Follow-up untuk prospek: serangkaian email yang kita kirimkan berupa materi atau ajakan sampai akhirnya terjadi pembelian
- Follow-up untuk reseller: Serangkian email yang kita kirimkan berupa materi strategi promosi yang benar dan sebagainya

Untuk melakukan follow-up secara otomatis kita memerlukan suatu servis yang dinamakan **autoresponder**.



Gambar 5.3: Form untuk menangkap email prospek

Autoresponder merupakan suatu servis atau layanan yang memungkinkan anda untuk mengirimkan email dalam interval waktu tertentu sesuai settingan anda.

Jika anda tanya seberapa besar pengaruh follow up terhadap jumlah pembelian produk anda. Tentu follow up sangatlah penting. Karena biasanya orang harus diyakinkan beberapa kali sebelum akhirnya memutuskan membeli.

Dengan menggunakan autoresponder kita dapat terus mengirimkan email yang berbeda-beda guna meyakinkan calon konsumen kita.

Jika servis web hosting anda tidak menyediakan fasilitas autoresponder. Maka **FreeAutobot** (<http://www.freeautobot.com/>) adalah pilihan alternatif yang sangat saya anjurkan.

Meskipun FreeAutobot menyediakan servis gratis untuk autoresponder yaitu sebanyak 12 kali pengiriman email. FreeAutobot tidak memasang iklan pada setiap email yang dikirim sehingga email kita lebih tampak professional.

Dalam 12 kali pengiriman email tersebut anda dapat menjadwal interval waktu masing-masing email. Misalnya email ke-2 dikirim 3 hari setelah email pertama. Email ke-3 dikirim 5 hari setelah email ke-2 dan seterusnya. Semua itu dapat anda atur pada halaman member freeautobot.com.

### Tips Isi Follow-up Prospek

Prospek disini adalah user atau pengunjung yang telah mengunjungi website kita namun belum melakukan pemesanan pada kali pertama. Email dari para prospek ini kita dapat ketika mereka memasukkan nama dan email untuk mendapatkan e-book gratis.

Isi dari follow-up bagi para prospek harus anda rancang untuk dapat membangkitkan kembali minat mereka untuk membeli produk anda. Biasanya anda harus melakukan berkali-kali follow-up agar mereka akhirnya membeli produk anda.

Namun jangan khawatir karena dengan autoresponder pengiriman follow-up tersebut terkirim secara otomatis. Yang perlu anda lakukan adalah merancang isi dari email yang anda kirim.

Contoh isi email untuk follow-up prospek antara lain:

1. Pada pesan pertama ini berikan link download untuk produk gratis yang telah anda janjikan sebelumnya. Jangan selalu cantumkan alamat website anda diakhir setiap pesan.
2. Pada follow-up ke-2 berikan suatu informasi atau masalah yang dapat diatasi oleh produk anda.
3. Pada email ke-3 tonjolkan manfaat produk lebih banyak.
4. Masukkan daftar manfaat lebih banyak lagi dan produk-produk gratis berkualitas jika mereka membeli produk anda.
5. Cantumkan testimonial dari member yang telah membeli produk anda lebih dulu.
6. Masukkan lebih banyak testimonial lagi dan buatlah alasan mengapa mereka harus membeli sekarang.

Dari 6 pesan follow-up seperti contoh diatas. Berilah interval waktu yang cukup lama dan bervariasi. Misalnya antara 4 hari sampai 7 hari. Karena kita ingat tidak semua orang mengecek emailnya setiap hari.

Jadi jika mereka melihat di *inbox* penuh dengan email dari kita. Maka yang kita dapat dari mereka bukannya rasa minat untuk membeli. Tetapi rasa jengkel karena banyaknya email yang mereka terima. Namun, jika kita mengirimnya dengan interval waktu yang cukup lama. Kemungkinan satu email kita untuk dibaca setiap waktu user mengecek inboxnya akab lebih besar.

### **Tips Follow-up Reseller/member**

Para reseller merupakan senjata utama anda dalam menghasilkan income yang besar. Karena dengan reseller anda tidak perlu susah payah melakukan promosi.

Tugas promosi anda digantikan oleh para reseller yang dengan rajin mempromosikan iklan anda ke berbagai media iklan.

Karena fungsi reseller sangat penting dalam bisnis anda. Maka anda perlu melakukan training terhadap mereka. Agar nantinya mereka dapat melakukan promosi dengan baik. Sehingga anda sebagai pengelola memperoleh keuntungan. Begitu pula dengan mereka para reseller.

Training yang dapat anda berikan pada mereka adalah dengan menyediakan suatu halaman khusus bagi reseller seputar strategi promosi. Selain itu anda juga dapat melakukan follow-up ke semua reseller lewat email.

Contoh isi email untuk follow-up reseller:

1. Sampaikan terima kasih anda atas bergabungnya mereka menjadi member website anda. Berikan penjelasan bahwa mereka juga otomatis menjadi reseller website anda dan berkesempatan menghasilkan uang.
2. Berikan penjelasan mengenai tips-tips seputar promosi. Bimbing mereka agar promosi yang mereka lakukan efektif.
3. Berikan strategi promosi yang lebih detil.
4. Lanjutkan mengenai startegi promosi.
5. dan seterusnya

Dengan melakukan follow-up seperti diatas mereka akan merasa bahwa anda memperhatikan member anda. Dengan demikian rasa percaya mereka kepada anda akan lebih besar. Dengan begitu, ketika anda ingin meluncurkan produk baru. Anda sudah dapat memiliki target market yang positif.

## Sistem Reseller Otomatis

Jika anda ingin menghasilkan traffic yang melimpah ke website anda. Salah satu cara yang paling efektif adalah dengan menggunakan reseller. **Reseller** adalah user yang telah menjadi member anda dan berhak mendapat komisi sekitar % dari produk anda sesuai perjanjian.

Dengan konsep tersebut yaitu mengubah setiap member menjadi reseller. Pekerjaan promosi anda akan sangat ringan. Karena sebagian besar telah dilakukan oleh para reseller anda.

Sistem pelayanan reseller otomatis ini memiliki konsep sebagai berikut:

1. Setiap member adalah reseller
2. Setiap reseller memiliki website duplikasi sendiri-sendiri
3. Pengaturan komisi secara otomatis
4. Pengelola dapat memantau para reseller
5. Reseller dapat memantau sendiri statistiknya

Anda sebagai member rahasia-webmaster.com juga merupakan reseller kami. Anda juga memiliki website duplikasi sendiri-sendiri sesuai username anda.

Contoh URL reseller sebagai berikut:

[http://www.rahasia-webmaster.com?ref=username\\_anda](http://www.rahasia-webmaster.com?ref=username_anda)

Dari website duplikasi diatas secara otomatis ketika terjadi transaksi penjualan. Rekening anda yang akan muncul. Hal itu yang akan kita terapkan pada contoh website reseller nantinya.

Anda sebagai administrator atau pengelola dapat memantau aktivitas dari para reseller. Berapa traffic yang dihasilkan, berapa jumlah pengunjung, komisi dan sebagainya. Para reseller juga dapat memantau sendiri aktivitasnya melalui member area.

Untuk sistem bagi hasil anda sebagai pemilik terserah mau menetapkan berapa. Sebagai referensi, pada rahasia-webmaster.com. Sistem bagi hasil adalah 50:50, yaitu 50% untuk saya sebagai pengelola dan 50% untuk reseller. Anda dapat menerapkan hal yang serupa pada website anda.

Untuk bagi hasil sebaiknya gunakan cara langsung yaitu dengan membaginya saat pemesanan. Dimana pemesan harus mentransfer ke dua rekening yaitu rekening anda sebagai pengelola dan reseller anda.

Script yang akan kita buat untuk sistem reseller otomatis sebenarnya cukup simpel. Kita hanya perlu menerapkan beberapa hal yaitu:

### #Halaman non Member

1. Mengambil isi dari variabel pada URL untuk mendapatkan username member. Menggunakan `$_GET`.
2. Melakukan query SQL untuk mendapatkan data dari database berupa nama bank, no. Rekening dan sebagainya.
3. Mengirim email konfirmasi ke user atas pemesanan produk yang dilakukan

### #Halaman member

1. Lakukan query untuk mendapatkan daftar pemesan berdasarkan username member tersebut.
2. Lakukan looping while atau for untuk menampilkan semua pemesan.
3. Tampilkan hasilnya dalam bentuk tabel
4. Sediakan pilihan blokir berupa *checkbox* dan tombol blokir untuk melakukan pemblokiran member.

Tugas yang diemban oleh reseller sebenarnya sederhana. Yaitu mendatangkan sebanyak mungkin pengunjung ke website anda. Para reseller tidak perlu membuat website rumit dan mendesain produk.

Mereka cukup mereferensikan URL khusus yang anda berikan. Dan otomatis mereka akan mendapat komisi sekitar % dari anda jika terjadi penjualan.

## Meta Tags

Ketika anda mencari suatu informasi pada search engine, Google misalnya. Hasil dari pencarian tersebut biasanya mencantumkan judul(title) dari website tersebut sebagai link utama. Kemudian ada sedikit penjelasan setelahnya.

Hal yang sama juga berlaku pada website anda ketika website anda muncul di google. Judul website anda akan digunakan sebagai nama link pada daftar hasil pencarian.

### Lalu Apa Hubungannya?

Meta tags merupakan kumpulan tag yang diletakkan diantara tag <head></head> yang mendeskripsikan website tersebut. Seperti keyword, deskripsi, dibuat oleh program apa dan sebagainya. Disini yang perlu anda perhatikan pada pembuatan meta tag adalah meta tag **keywords** dan meta tag **description**.

Keberhasilan suatu website dalam mencapai ranking teratas pada search engine tidak lepas dari penggunaan kedua tag tersebut.

```
<html>
<head>
<title>Menjadi Webmaster dalam 30 Hari</title>

<meta name="keyword" content="rahasia webmaster, lowongan pekerjaan, bisnis
internet, peluang usaha">

<meta name="description" content="Temukan rahasia bagaimana menjadi webmaster
hanya dalam waktu 30 hari saja!. Dan bagaimana menghasilkan jutaan rupiah
melalui internet">

</head>
```

Contoh diatas adalah meta tags yang saya gunakan pada website rahasia-webmaster.com.

Dengan menggunakan meta tags anda dapat mengontrol tampilan yang akan ditampilkan search engine pada hasil pencarian.

Meskipun demikian saya anjurkan anda juga mendaftarkan website anda pada google agar dapat diindex. Untuk mendaftarkan URL website anda klik ke <http://www.google.com/addurl>. Ikuti petunjuk yang diberikan. Setelah melakukan pendaftaran URL butuh waktu sekitar 1 minggu atau lebih agar website anda dapat diindex google.

#### Tips

Pada meta tag keywords lebih baik gunakan frase daripada kata tunggal. Karena pencarian kebanyakan menggunakan frase bukan kata tunggal. Juga gunakan kata yang bervariasi misalnya e-book, E-book, ebook, dan sebagainya.

Jika anda memiliki kompetitor yang lebih unggul dalam hal rangking di search engine. Anda dapat membuka alamat website kompetitor anda tersebut lalu klik **View > Source**. Coba lihat bagian meta keywords dan descriptionnya, anda dapat mencontoh dari kompetitor tersebut.

Jika anda ingin mendapatkan target yang besar. Daftarkan website anda ke search engine yang populer seperti Google atau Yahoo!.

## Strategi Promosi

Selain dengan menggunakan search engine, cara lain yang sangat efektif diterapkan pada target market indonesia adalah **iklan baris** dan **Milis** atau **Newsgroup**.

### Iklan Baris

Di Indonesia situs iklan baris memiliki peranan yang penting dalam hal promosi. Karena kebanyakan netter Indonesia melakukan promosi produknya melalui situs-situs iklan baris. Mengapa? Jawabnya simpel saja. Karena iklan baris kebanyakan bersifat gratis dan instan. Bahkan banyak diantara situs iklan baris yang memiliki pengunjung ribuan per harinya. Ini tentu suatu "*tambang emas*" bagi pemilik produk seperti anda untuk memasarkan produknya.

Di bawah ini adalah daftar situs iklan baris yang memiliki pengunjung terbanyak:

[www.bekas.com](http://www.bekas.com)  
[www.iklanbaris.com](http://www.iklanbaris.com)  
[www.iklaniklan.com](http://www.iklaniklan.com)  
[www.bursaiklan.com](http://www.bursaiklan.com)  
[www.iklanpojok.com](http://www.iklanpojok.com)  
[www.iklanmini.com](http://www.iklanmini.com)  
[www.jual-beli.com](http://www.jual-beli.com)

Untuk situs-situs lainnya anda dapat mencarinya di search engine dengan mengetikkan keyword "iklan baris" atau "iklan mini".

### Mengoptimalkan Pemasangan Iklan

Anda tentu menyadari kalau memasang iklan di situs-situs iklan baris kita harus mengisi banyak form. Seperti judul iklan, isi iklan dan sebagainya. Hal ini tidak masalah jika kita hanya ingin mengisi beberapa iklan saja.

Namun bagaimana jika kita ingin memasang lebih dari tiga kali. Dan itu kita lakukan di situs iklan baris yang berbeda.

Hal tersebut tentu akan menguras tenaga dan menimbulkan kejemuhan. Oleh karena itu kita harus berusaha untuk mengoptimalkan pemasangan iklan kita. Bagaimana caranya?

Salah satu cara yang cukup efektif dan tidak membutuhkan biaya sama sekali adalah dengan memanfaatkan software pengisi form otomatis.

Saya pribadi dalam mengisi form-form pada situs-situs iklan baris memanfaatkan software **RoboForm** ([www.roboform.com](http://www.roboform.com)). Dengan roboform proses pengisian form dapat anda lakukan dalam 1x klik mouse. Yang anda perlukan hanyalah merancang isi dari setiap form yang akan diisi secara otomatis oleh RoboForm.

Penggunaan RoboForm sangat mudah. Karena setelah diinstall secara otomatis RoboForm akan terintegrasi dengan toolbar dari Internet Explorer atau FireFox.

RoboForm hadir dalam dua versi yaitu versi Shareware dan Freeware. Saya pribadi menggunakan versi freeware karena saya rasa kemampuan roboform sudah cukup baik pada versi ini. Meskipun ada beberapa keterbatasan dibanding dengan versi PRO-nya.

Untuk lebih jelasnya silahkan anda baca manual atau help dari RoboForm.

## Meningkatkan Nilai Iklan

Yang dimaksud meningkatkan nilai iklan disini adalah banyaknya orang yang mengklik iklan anda dari sekian orang yang membaca iklan anda.

Jika ada 100 orang yang membaca iklan anda dan ada 10 orang yang mengklik iklan anda. Maka perbandingannya adalah 10%, semakin besar perbandingannya maka semakin baik pula nilai iklan anda.

Satu hal yang perlu anda pegang dalam melakukan promosi adalah jangan pernah melakukan SPAM. Karena hal tersebut melanggar privasi setiap orang.

SPAM merupakan pengiriman email ke alamat email seseorang yang orang tersebut tidak menghendakinya.

\*\*\*\*\*

# Web Design

Karena kita disini berbicara masalah bisnis. Maka kita tidak boleh mengesampingkan urusan tampilan website. Memang untuk tampilan website tidak terlalu penting dalam penjualan produk kita.

Namun, ingat kata pepatah, "Dari Mata Turun ke Hati". Kata mutiara tersebut saya rasa juga berlaku untuk website kita. Karena jika tampilan website kita sudah tidak *sedap* dipandang. Pengunjung akan berpikir dua kali untuk membacanya secara keseluruhan.

Website yang bagus tidak harus memakai animasi yang hi-tech seperti flash dan sebagainya. Yang penting adalah kerapian dan kemudahan navigasi pada website tersebut harus diperhatikan.

Berilah gambar secukupnya saja dan jangan terlalu banyak. Karena hal itu hanya akan memperlambat proses loading website anda.

Ada beberapa kriteria yang harus anda perhatikan dalam mendesain website khususnya website komersial. Kriteria-kriteria tersebut antara lain:

- Tampilan website
- Kesesuaian isi website dengan tampilan
- Kemudahan navigasi

Dalam mendesain suatu website bisnis tampilan website tersebut harus mencerminkan bidang yang ia promosikan.

Kita ambil contoh sebuah website perbankan. Dalam mendesainnya kita harus memikirkan aspek kerapian dan kemudahan navigasi bagi user. Dan jangan menset backgroundnya dalam warna-warna gelap seperti hitam. Jangan-jangan nanti malah dianggap website underground.

Pilihlah warna-warna yang cerah untuk background namun tidak terlalu kontras. Seperti putih, abu-abu, dan lainnya.

Aspek kerapian adalah hal yang juga harus anda perhatikan. Meskipun website anda tidak memiliki banyak gambar. Namun jika anda mendesainnya dengan rapi, website tersebut akan kelihatan bagus.

Kerapian yang dimaksud disini meliputi tata letak gambar, tata letak tulisan, jenis huruf yang digunakan dan sebagainya.

Usahakan untuk meletakkan gambar seperlunya saja dan jangan terlalu berlebihan. Karena dengan gambar yang terlalu banyak, waktu loading website anda pun akan menjadi lama. Cukup tambahkan gambar yang sekiranya dapat mendukung isi dari website anda.

Tata letak tulisan juga perlu anda pertimbangkan pendesainannya. Lebih-lebih jenis huruf yang anda gunakan.

Dalam memilih jenis huruf untuk sebuah website bisnis. Pilihlah huruf-huruf yang mudah dibaca dan jelas. Saya lebih menganjurkan untuk anda menggunakan jenis huruf verdana atau arial. Hindarilah menggunakan huruf-huruf yang tidak lazim. Meskipun menurut anda huruf tersebut *keren*.

Aspek kemudahan navigasi juga merupakan elemen penting dalam sebuah website. Website yang bagus juga tidak lepas dari masalah pendesainan navigasi. Navigasi harus kita desain agar pengunjung dapat dengan mudah berpindah-pindah antar halaman yang ia inginkan.

Tanpa adanya navigasi seorang user akan kesulitan dalam berbindah-pindah antar halaman. Sebaiknya link navigasi diletakkan pada bagian atas halaman dibawah gambar utama(banner). Atau diletakkan disamping kiri atau kanan.

Kadang kala kita memerlukan gambar yang akan diletakkan pada website kita. Untuk itu kita perlu memiliki sedikit keahlian dalam menjalan program pengolah gambar seperti Adobe(R) Photoshop(R).

**Catatan:** Anda tidak harus menggunakan photoshop, anda dapat menggunakan program lain, misalnya Paint Shop Pro. Namun jika anda menginginkan program pengolah gambar gratis namun tidak kalah dengan photoshop **The Gimp** pada linux adalah jawabannya.

Untuk dapat mempelajari program pengolah seperti Adobe(R) Photoshop(R). Anda dapat membeli buku-buku yang membahas program tersebut. Atau jika anda mau yang lebih murah. Cukup manfaat pustaka terbesar di muka bumi apalagi kalau bukan INTERNET.

Saya rasa tidak sulit untuk mempelajari program tersebut. Dan dalam waktu satu hari saja saya rasa anda sudah dapat menguasai teknik-teknik yang umum digunakan.

Jika orientasi anda adalah website reseller seperti yang akan kita buat. Ada beberapa point penting yang perlu anda perhatikan.

Karena prospek adalah segala-segalanya maka dalam mendesain tampilan website tersebut. Letakkan form untuk menangkap email prospek hampir disetiap halaman. Ini akan selalu mengingatkan user untuk mengisi form tersebut. Lihat [rahasia-webmaster.com](http://rahasia-webmaster.com).

Di internet banyak sekali website yang bisa anda jadikan inspirasi sebagai bahan pertimbangan untuk desain website anda.

Salah satu tempat yang paling menarik untuk dikunjungi adalah di <http://www.freewebstemplate.com/>.

Di website tersebut terdapat ratusan template website yang bisa menginspirasi anda. Mulai dari kategori hi-tech, bisnis, multimedia dan masih banyak lagi.

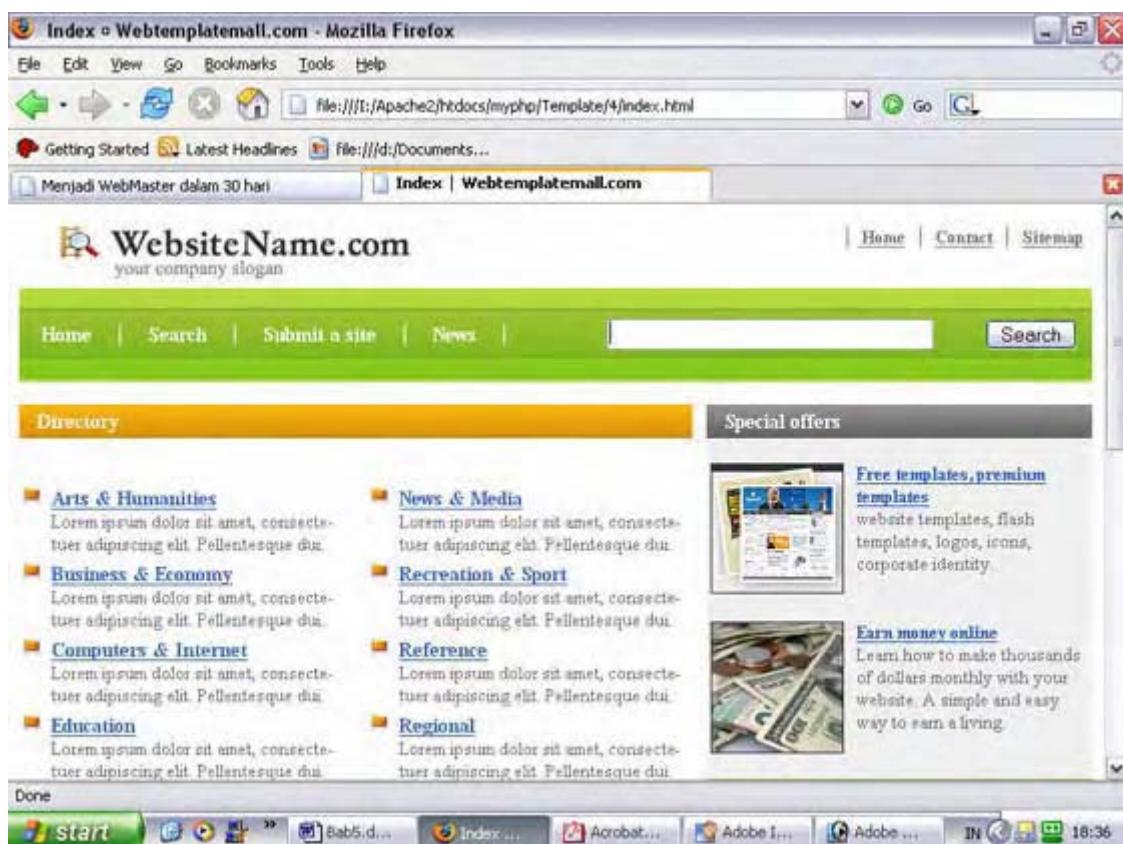
Berikut ini adalah beberapa contoh template yang saya ambil dari freewebstemplate.com



Gambar 5.4



Gambar 5.5



Gambar 5.6



Gambar 5.7

Contoh-contoh template diatas sebagian besar adalah website bisnis. Coba perhatikan tampilan dan tata letak gambarnya. Rapi dan menarik bukan?, jadi jika suatu saat anda

mendapatkan job untuk membuatkan website untuk perusahaan. Dan anda kehabisan inspirasi anda dapat mengunjungi freewebstemplate.com.

\*\*\*\*\*

## Kalender

27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

## Hari Ke-26, 27, 28 & 29

Sebelum melanjutkan seperti biasa lakukan hal berikut:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab5
- Buat folder baru nama folder tersebut **hari-26-27-28-29**

# ladangbisnis.localhost

Pada pembahasan sebelumnya kita telah membahas bagaimana membuat sebuah website yang menjual. Website yang mampu menghasilkan jutaan rupiah untuk kita. Kali ini kita akan membuat website tersebut.

Website yang akan kita buat adalah sebuah website reseller. Dimana user yang telah menjadi member dari website kita secara otomatis dia akan menjadi reseller website kita. Sesuai dengan pembahasan sebelumnya bahwa dalam membuat website yang menjual kita memerlukan beberapa strategi. Diantaranya headline website, sales letter yang menarik, tampilan website yang bersih dan rapi dan sebagainya.

Di bawah ini adalah gambar jadi website yang akan kita buat.



Gambar 5.8: Halaman utama ladangbisnis.localhost

Bagaimana tampilannya? lebih baik daripada proyek-proyek terdahulu kita bukan. Pada ladangbisnis.localhost saya hanya membuat satu gambar yaitu gambar banner yang terletak pada bagian atas website tersebut. Gambar ini saya desain sesuai dengan kebutuhan website ladangbisnis.localhost. Saya membuatnya menggunakan program **The Gimp 2.2** pada linux.

Pada windows anda dapat menggunakan program **Photoshop**. Anda dapat menggantinya dengan gambar lainnya sesuai keinginan anda. Asal warnanya anda sesuaikan dengan content website ladangbisnis.localhost juga ukurannya yaitu 798x100 pixel. Jika anda tidak dapat membuatnya cukup download materi pada bab 5 di alamat berikut:  
<http://materi.rahasia-webmaster.com/bab5/gambar>

Selanjutnya kita akan membahas teknis pembuatan website reseller yaitu ladangbisnis.localhost. Kita akan membuat website ini dalam waktu empat hari kita mulai pada hari ke-26. Syukur-syukur jika anda dapat menyelesaiannya lebih cepat.

Agar anda dapat lebih memahami gambaran umum tentang ladangbisnis.localhost, berikut adalah sketsa proyeknya.

**NB:** Karena pada ladangbinis.localhost banyak melibatkan pengiriman email. Maka jika anda tidak memiliki program mail server atau anda belum di server sebenarnya. Anda dapat memberi komentar pada perintah `mail()` agar tidak dieksekusi oleh PHP sehingga tidak muncul error.

## Sketsa Proyek

Website yang akan kita buat adalah website reseller. Dimana website ini memiliki tiga bagian utama, yaitu bagian pengunjung, bagian member, dan bagian administrator. Untuk bagian member dan administrator user harus mengisikan username dan password untuk dapat masuk.

Harga produk kita yaitu e-book "Internet Sebagai Ladang Bisnis" adalah Rp. 100.000,-. Dan komisi untuk reseller adalah 50% atau Rp. 50.000,-

Setiap member yang masuk lewat rekomendasi reseller kita. Maka secara otomatis ia(reseller) akan memperoleh komisi 50% dari harga produk kita. Mari kita bahas satu per satu dari masing-masing bagian tersebut.

### Bagian Administrator

Pada bagian ini adalah halaman-halaman yang dikhususkan untuk administrasi website ladangbisnis.localhost diantaranya:

- Untuk masuk ke admin area administrator harus mengisi username dan password.
- Admin dapat memantau statistik lengkap seputar ladangbisnis.localhost seperti banyaknya member yang bergabung, banyaknya traffic ke website, banyaknya pemesan, banyaknya member yang diblokir, banyaknya keuntungan yang diperoleh.
- Admin dapat mengaktifasi user yang telah melakukan transfer dana atau yang telah diblokir.
- Admin dapat memblokir keanggotaan member yang dirasa telah melanggar kesepakatan.
- Admin dapat menghapus member, pemesan, prospek dan member terblokir.
- Admin dapat melakukan pencarian data baik member, pemesan, atau member terblokir berdasarkan kriteria yang diinginkan.
- Admin dapat mengirim email ke semua user yaitu member, pemesan, terblokir dan prospek atau secara spesifik ke satu user saja.
- Admin dapat menambah, mengedit atau menghapus testimonial.

### Bagian Member Area

Pada bagian ini adalah halaman-halaman yang khususkan untuk member ladangbisnis.localhost diantaranya:

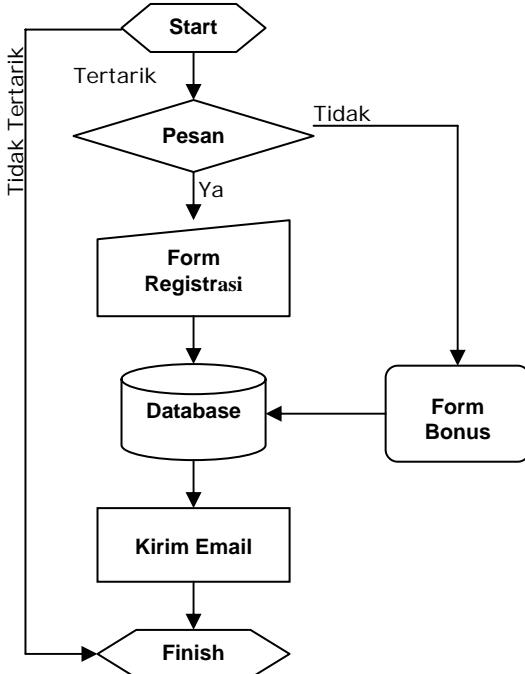
- Untuk masuk ke member area user harus mengisikan username dan password.
- Member dapat memantau statistiknya sendiri seperti banyaknya pemesan, banyaknya member, banyaknya member terblokir, dan banyaknya prospek yang masuk lewat rekomendasinya.
- Member dapat mengetahui traffic dan banyaknya komisi yang diperoleh.
- Member dapat mendownload produk utama.
- Member dapat memblokir member lain yang telah diaktifasi admin namun belum mentransfer dana ke rekeningnya.
- Member dapat memberikan testimonialnya di member area.
- Member dapat mengetahui halaman strategi promosi.

## Bagian Pengunjung

Pada bagian ini adalah halaman-halaman yang dikhususkan untuk pengunjung/non-member diantaranya:

- User dapat membaca/review dari produk kita.
- User dapat mendownload e-book gratis yang kita sediakan.
- User dapat memesan produk kita.
- Pada saat halaman transfer dana, rekening dibagi 2 satu untuk pengelola yaitu anda dan satu lagi untuk reseller anda.
- User dapat melihat daftar testimonial dan bonus.
- User dapat mengontak admin lewat form kontak.
- User dapat mengetahui manfaat menjadi reseller website anda.

Flowchart dari alur kerja website ladangbisnis.localhost dapat anda lihat pada gambar 5.9.



Gambar 5.9: Flowchart alur kerja dari ladangbisnis.localhost

Pada flowchar gambar 5.9 kita sudah mendapatkan gambaran umum tentang cara kerja ladangbisnis.localhost. Pertama kita menghadapi kemungkinan apakah user tertarik atau tidak pada ladangbisnis.localhost. Jika tertarik maka proses dilanjutkan namun jika tidak maka proses akan berakhir yang artinya kita gagal mendapatkan user tersebut.

Kemungkinan kedua user tertarik namun ia akan memesan atau tidak masih menjadi pertanyaan. Jika tidak memesan mungkin user tertarik pada bonus yang kita berikan dan ia mengisi form bonus. Setelah itu user akan masuk pada daftar prospek di database kita. Selanjutnya kita mengirim email berupa alamat link download bonus.

Kemungkinan kedua user memesan produk kita, maka proses selanjutnya adalah user harus mengisi form registrasi. Kemudian data user tersebut masuk pada daftar pemesan di database. Halaman yang berisi daftar transfer dana yaitu pengelola dan reseller ditampilkan. Yang terakhir adalah proses pengiriman email konfirmasi ke user yang baru saja mendaftar.

Sebelum mulai mengerjakan proyek ini lakukan beberapa konfigurasi berikut:

Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab5\hari-26-27-28-29

- Buat folder baru, dan namai folder tersebut **reseller**.
- Masuk pada direktori **reseller** yang baru anda buat.
- Buat beberapa folder baru diantaranya **gambar**, **include**, **member**, **\_my\_cp\_**, dan **template**.

Kita tidak menamakan direktori untuk file-file administrasi dengan nama *admin* melainkan kita ganti dengan *\_my\_cp\_*. Hal ini kita lakukan agar user atau para attacker yang akan mencoba masuk ke admin area sedikit mengalami kesulitan dalam menemukan form login untuk administrator. Anda dapat menggantinya dengan kata yang lain yang menurut anda sulit ditebak.

Daftar file yang akan kita buat untuk website ladangbisnis.localhost dapat anda lihat pada tabel 5.x.

**Tabel 5.x** Daftar file untuk website ladangbisnis.localhost.

<b>Direktori reseller\_my_cp_</b>	
<b>Nama File</b>	<b>Fungsi</b>
cari.php	untuk melakukan pencarian data
index.php	form login dan halaman utama admin area
mail_to.php	untuk mengirim email ke seluruh user
proses_db.php	memproses data berupa pengaktifan, pemblokir user dan lainnya
statistik.php	menampilkan statistik user
testimonial.php	menambah, mengedit, dan menghapus testimonial
<b>Direktori reseller\member</b>	
<b>Nama File</b>	<b>Fungsi</b>
cari.php	untuk pencarian data
download.php	menampilkan halaman download
index.php	halaman utama member area
profil.php	untuk mengupdate profil member
statistik.php	menampilkan statistik dari member
strategi.php	menampilkan halaman strategi promosi
testimonial.php	menampilkan form testimonial dan mengirimnya ke admin
<b>Direktori reseller\include</b>	
<b>Nama File</b>	<b>Fungsi</b>
admin_var.php	menyimpan kumpulan variabel untuk halaman administrator
class_halaman.php	class untuk menampilkan navigasi halaman
class_skin.php	class untuk mengganti tampilan halaman
class_waktu.php	class untuk menampilkan waktu dan deadline pembelian
fungsi.php	menyimpan kumpulan fungsi yang diperlukan
index.php	ladangbisnis.localhost
konfig.php	mengelola redirect ke halaman utama
member_var.php	berisi konfigurasi halaman ladangbisnis.localhost
var_mail.php	menyimpan kumpulan variabel untuk halaman member
var_utama.php	menyimpan kumpulan variabel untuk template email
	menyimpan kumpulan variabel untuk halaman non-member
<b>Direktori reseller\template</b>	
<b>Nama File</b>	<b>Fungsi</b>
index_skin.php	file skin untuk halaman non-member
member_skin.php	file skin untuk halaman member dan administrator
style.css	file CSS untuk semua halaman
<b>Direktori reseller\Gambar</b>	
<b>Nama File</b>	<b>Fungsi</b>
banner.jpg	Gambar utama ladangbisnis.localhost (opsional)
index.php	Meredirect ke halaman utama
<b>Direktori reseller</b>	
<b>Nama File</b>	<b>Fungsi</b>
bonus.php	menampilkan daftar bonus
db_reseller.sql	untuk membangun database reseller
index.php	halaman utama ladangbisnis.localhost (sales letter)

kontak.php	menampilkan form kontak dan memprosesnya
insert.php	memproses login user
pesan.php	menampilkan form registrasi dan memprosesnya
reseller.php	menampilkan keuntungan menjadi reseller ladangbisnis.localhost
testimonial.php	menampilkan daftar testimonial member
thankyou.php	memproses data registrasi dan mengirim email ke user

Seperti proyek-proyek terdahulu kita mulai mengerjakannya dari file databasenya terlebih dahulu. Untuk itu klik menu **File > New > SQL** pada PHP DESIGNER 2006, kemudian salin kode berikut.

#### Script dari file db\_reseller.sql

```
-- buat database reseller
-- hapus jika sudah ada

DROP DATABASE IF EXISTS reseller;
CREATE DATABASE reseller;

-- aktifkan database reseller
USE reseller;

-- buat tabel tbl_admin
CREATE TABLE tbl_admin
(
 username varchar(16) PRIMARY KEY,
 password varchar(32)
) TYPE='MyISAM';

-- buat tabel tbl_blokir
CREATE TABLE tbl_blokir
(
 tgl varchar(10),
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 nama varchar(50),
 email varchar(30),
 kota varchar(20),
 reseller varchar(16),
 transfer int(10)
) TYPE='MyISAM';

-- buat tabel tbl_member
CREATE TABLE tbl_member
(
 tgl varchar(10),
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 nama varchar(50),
 email varchar(30),
 kota varchar(20),
 reseller varchar(16),
 transfer int(10)
) TYPE='MyISAM';

-- buat tabel tbl_pemesan
CREATE TABLE tbl_pemesan
(
 tgl varchar(10),
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 nama varchar(50),
 email varchar(30),
 kota varchar(20),
```

```

reseller varchar(16),
transfer int(10)
) TYPE='MyISAM';

-- buat tabel tbl_profil
CREATE TABLE tbl_profil
(
username varchar(16) PRIMARY KEY,
password varchar(32),
nama varchar(50),
email varchar(30),
kota varchar(20),
reseller varchar(16),
alamat varchar(100),
telpon varchar(20),
bank varchar(100)
) TYPE='MyISAM';

-- buat tabel tbl_prospek
CREATE TABLE tbl_prospek
(
tgl_masuk varchar(10),
email varchar(30) PRIMARY KEY,
nama varchar(50),
reseller varchar(16)
) TYPE='MyISAM';

-- buat tabel tbl_testimonial
CREATE TABLE tbl_testimonial
(
tm_id int(4) PRIMARY KEY auto_increment,
username varchar(16),
testimonial text
) TYPE='MyISAM';

-- buat tabel tbl_traffic
CREATE TABLE tbl_traffic
(
username varchar(16) PRIMARY KEY,
traffic int(6)
) TYPE='MyISAM';

CREATE TABLE tbl_transfer
(
nilai int(6) PRIMARY KEY
) TYPE='MyISAM';

-- masukkan account admin ke tbl_admin
-- password => _super_pass_ = 1835a30bcdedf3c4fa4fca7b6d9092113
-- setelah dienkrip dengan balik_md5()
INSERT INTO tbl_admin VALUES
(''_super_user_','1835a30bcdedf3c4fa4fca7b6d9092113');

-- masukkan beberapa data ke tbl_member, tbl_pemesan, dan tbl_profil
-- masukkan member (kita asumsikan passwordnya pas123)
-- pass123 = 0da656c1352c7ec167b91122dd6d1aff (setelah dienkrip balik_md5())
INSERT INTO tbl_member VALUES
('26-10-2005', 'joni', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Joni',
'joni@localhost.localdomain', 'Surabaya', 'admin', 50000),
('26-10-2005', 'jono', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Jono',
'jono@localhost.localdomain', 'Surabaya', 'joni', 50001),
('26-10-2005', 'joko', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Joko',
'joko@localhost.localdomain', 'Surabaya', 'joni', 50002);

```

```
-- masukkan beberapa data ke tbl_profil
INSERT INTO tbl_profil VALUES
('joni', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Joni',
'joni@localhost.localdomain', 'Surabaya', 'admin', 'Jl. Apache 80',
'080808080', 'Bank Localhost - Rek. 9876543210 - a.n Mr. Joni'),
('jono', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Jono',
'jono@localhost.localdomain', 'Surabaya', 'joni', 'Jl. Mail 21',
'070707070', 'Bank Localhost - Rek. 0123456789 - a.n Mr. Jono'),
('joko', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Joko',
'joko@localhost.localdomain', 'Surabaya', 'joni', 'Jl. Netbios 139',
'060606060', 'Bank Localhost - Rek. 5432167890 - a.n Mr. Joko');

-- masukkan ke tbl_traffic
INSERT INTO tbl_traffic VALUES
('admin', 15), ('joni', 10), ('jono', 5), ('joko', 4);

-- masukkan pemesan
INSERT INTO tbl_pemesan VALUES
('26-10-2005', 'jojo', 'pas123', 'Mr. Jojo', 'jojo@localhost.localdomain',
'Surabaya', 'jono', 50003),
('26-10-2005', 'josi', 'pas123', 'Mr. Josi', 'josi@localhost.localdomain',
'Surabaya', 'joko', 50004);

-- masukkan beberapa orang ke prospek
INSERT INTO tbl_prospek VALUES
('26-10-2005', 'gatot@kaca.com', 'Gatot Kaca', 'joni'),
('26-10-2005', 'semar@mendem.com', 'Semar Sakti', 'joni'),
('26-10-2005', 'os@linux.com', 'Mr. Open Source', 'joko');

-- set nilai transfer ke 50005 karena yang terakhir adalah 50004
-- yaitu si josi
INSERT INTO tbl_transfer VALUES (50005);

-- buat privilege untuk database ini
GRANT SELECT, INSERT, DELETE, UPDATE ON reseller.*

TO res_user@localhost IDENTIFIED BY 'reseller123';
```

Simpan pada direktori paling atas yaitu direktori **reseller** dengan nama **db\_reseller.php**.

Sekarang kita perlu mengeksekusi file ini untuk menciptakan database reseller. Untuk yang pertama kita menggunakan MySQL monitor. Login pada MySQL monitor kemudian ketik kode berikut.

```
C:\>mysql -u root -p [ENTER]
ENTER PASSWORD: [PASSWORD ANDA]
```

```
mysql> source C:/Apache2/htdocs/webmaster/Bab5/hari-26-27-28-29/reseller/db_reseller.sql
```

Alternatif yang kedua kita dapat menggunakan phpMyAdmin. Pada phpMyAdmin klik tab **SQL** lalu klik tombol browser arahkan ke alamat seperti pada MySQL Monitor diatas.

Selanjutnya kita akan membuat file-file yang berada pada direktori **include**. Karena file-file pada direktori ini merupakan file library yang dipanggil hampir oleh semua halaman. Untuk yang pertama kita akan membuat file *index.php*. Tugas file ini hanyalah melakukan redirect jika ada yang mencoba masuk ke direktori *include*.

Pada pada PHP DESIGNER 2006 klik menu **File > New > PHP** untuk membuat file PHP baru, kemudian ketik kode berikut.

Script dari file **index.php**

```
<?php
```

---

```

** Nama File.....: index.php ***
** Penulis.....: Rio Astamal ***
** Tanggal.....: 20-10-2005 ***
** Penjelasan....: Untuk meredirect ke halaman utama ***
*****/

// jika user mencoba masuk ke halaman inc
// kembalikan ia ke halaman utama
header ("Location: ../index.php");
exit;

?>
```

---

Simpan pada direktori **reseller/include** dengan nama **index.php**.

File berikutnya adalah *konfig.php*, tugas file ini menyimpan berbagai variabel untuk tampilan suatu halaman. Seperti berapa testimonial yang harus ditampilkan per halaman dan sebagainya. Buat file PHP baru klik **File > New > PHP**. Kemudian salin kode berikut,

#### Script dari file **konfig.php**

```
<?php

** Nama File.....: konfig.php ***
** Penulis.....: Rio Astamal ***
** Tanggal.....: 17-10-2005 ***
** Penjelasan....: menyimpan konfigurasi baik untuk ***
** halaman member dan admin ***
*****/

// cegah pengaksesan langsung dari browser
if (eregi('konfig.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php'); // kembalikan ke halaman utama
 exit;
}

// KARENA DALAM DEVELOPMENT MAKA ANGKANYA KITA SET KECIL SAJA MISAL 2

/* KONFIGURASI UNTUK ADMIN */

// ganti jumlahnya sesuai keinginan anda
$a_pph = 2; // konfigurasi umum per halaman
$a_tmph = 2; // testimonial per halaman

$deadline = 2; // deadline hari

/* KONFIGURASI UNTUK USER/MEMBER */

$u_tmph = 2; // testimonial per halaman
$u_pph = 2; // konfigurasi umum per halaman

// AKHIR KONFIGURASI //
```

---

Simpan pada direktori **reseller/include** dengan nama **konfig.php**.

File berikutnya yang akan kita buat adalah file yang menyimpan semua fungsi yang dipakai pada semua halaman ladangbisnis.localhost. Langsung saja klik menu **File > New > PHP** untuk membuat file php baru kemudian ketik kode berikut.

## Script dari file ***fungsi.php***

---

```

/*
** Nama File.....: fungsi.php
** Penulis.....: Rio Astamal
** Tanggal.....: 20-10-2005
** Penjelasan....: menyimpan kumpulan fungsi
*/

// cegah pengaksesan langsung dari browser
if (eregi('fungsi.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php');
 exit;
}

// fungsi untuk mengkoneksikan ke MySQL server
function koneksi_db()
{
 // untuk username = secure dan password = secure123
 @$koneksi = mysql_connect('localhost', 'res_user', 'reseller123');
 // jika gagal melakukan koneksi tampilkan pesan kesalahannya
 if (!$koneksi)
 {
 // jika kode anda sudah dirilis sebaiknya jangan menampilkan pesan
 // kesalahannya seperti berikut.
 // Gunakan metode ini hanya jika dalam tahap development saja.
 echo "Error: ".mysql_errno()."
\n";
 echo "Keterangan: ".mysql_error()."
\n";
 exit;
 }
 else
 {
 // pilih database yang digunakan
 mysql_select_db('reseller');
 return true;
 }
}

//cek setiap field apa ada yang kosong
function cek_field($var)
{
 foreach ($var as $field)
 {
 if ($field == '' || !isset($field))
 return false;
 }
 return true;
}

//cek kevalidan email
function cek_email($email)
{
 //fungsi untuk mengecek kevalidan email
 if (ereg('^[a-zA-Z0-9_\-]+@[a-zA-Z0-9\-_]+\.[a-zA-Z0-9\-_\.]+\$', $email))
 return true;
 else
 return false;
}

// fungsi untuk menyaring string selain alfabet, numerik dan -
// karakter pada parameter ke-2 tidak difilter jika diisi

```

```

function filter_str($string, $lainnya=' ')
{
 if ($lainnya == '')
 $filter = ereg_replace('[^a-zA-Z0-9_]', '', $string);
 else
 $filter = ereg_replace("[^a-zA-Z0-9_$lainnya]", '', $string);

 return $filter;
}

// fungsi untuk mengenkripsi string dengan metode MD5
// dan membalik urutannya
function balik_md5($string)
{
 // untuk membalik urutan string digunakan fungsi strrev()
 $chiper_text = strrev(md5($string));
 return $chiper_text;
}

// fungsi untuk mengecek session
function cek_session($nama_ses)
{
 // jika session kosong
 if (!isset($_SESSION[$nama_ses]))
 return false; // kembalikan nilai false
 else
 // jika tidak kosong
 return true; // kembalikan nilai true
}

// fungsi untuk login
function login($tabel, $username, $password)
{
 // lakukan query untuk mencocokkan data
 $hasil = mysql_query("SELECT * FROM $tabel WHERE username='$username'
 AND password='$password'");

 // cek baris yang dikembalikan
 if (mysql_num_rows($hasil) == 0)
 return false; // data tidak cocok
 else
 return true; // kembalikan nilai true
}

// fungsi untuk membuat password secara acak
// digunakan untuk mengirim password pada form lupa password
function pass_acak($panjang=8)
{
 $kar = "ABCDEFGHIJKLMNPRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz";
 // acak karakter
 srand((double)microtime() * 1000000);
 // lakukan looping sebanyak $panjang
 for ($i=0; $i<$panjang; $i++) // default diulang sebanyak 8x
 {
 $nom_acak = rand() % 53; // untuk mendapatkan nomor acak, pada substr()
 $pass .= substr($kar, $nom_acak, 1); // ambil satu karakter
 }

 return $pass; // kembalikan hasil
}

// fungsi untuk logout (menghapus session)
function logout($nama_ses)
{
}

```

```

// jika session kosong
if (!isset($_SESSION[$nama_ses]))
 return false; // kembalikan false
else
{
 // jika tidak kosong hancurkan session tersebut
 unset($_SESSION[$nama_ses]);
 session_destroy();
 return true; // kembalikan nilai true
}

// ***** FUNGSI-FUNGSI BARU *****
// fungsi untuk mengupdate nilai transfer atau traffic
function update($pilih=0, $username='')
{
 if ($pilih == 0) // transfer
 $query = "UPDATE tbl_transfer SET nilai=nilai+1";
 else if ($pilih == 1) // visit
 $query = "UPDATE tbl_traffic SET traffic=traffic+1 WHERE
 username='$username'";
 else
 return "Error: Pilihan 0 untuk transfer dan 1 untuk visit";
 $hasil = mysql_query($query);
 // cek status
 if (!$hasil)
 return false;
 else
 return true;
}

// fungsi untuk mendapatkan username reseller secara acak
function acak_reseller()
{
 // ambil username reseller dari database
 $hasil = mysql_query("SELECT username FROM tbl_member");

 // looping untuk membuat array $reseller
 while ($data = mysql_fetch_array($hasil))
 $reseller[] .= $data[0];

 srand((double)microtime() * 1000000);
 shuffle($reseller); // acak array

 for ($i=0; $i<count($reseller); $i++)
 return $reseller[$i]; // kembalikan nilai
}

// fungsi untuk langsung memecah hasil query menjadi array
function SQL_array($query)
{
 $hasil = mysql_query($query); // query SQL
 $my_array = mysql_fetch_array($hasil); // pecah menjadi array

 return $my_array; // kembalikan hasil
}

//fungsi untuk mengkonversi nilai rupiah ke format Rp. xxx.xxx,-
function rupiah($rp)
{
 $rupiah = 'Rp. '.number_format($rp, 0, ',', '.').',-';
 return $rupiah;
}

```

```

}

// fungsi untuk mencetak kolom pada suatu tabel
// cukup berguna menghemat energi jari-jari kita?!
function buat_kolom($datanya, $style='td', $lainnya='')
{
 // cek nilai dari style apakah kolom header atau biasa
 if ($style != 'td')
 $style = 'th';

 // parameter input berupa array
 reset($datanya); // pastikan index array dimulai dari awal
 $isi = current($datanya); // tunjuk index array yang aktif
 // looping untuk mencetak semua kolom/array
 while ($isi)
 {
 $kolom .= "<$style $lainnya>$isi</$style>\n";
 $isi = next($datanya); // index array berikutnya
 }

 return $kolom; // kembalikan hasil
}

// fungsi untuk memindah data antar tabel yang strukturnya sama
function pindah_data($tb_awal, $tb_tujuan, $opsi, $isi)
{
 // lakukan query gabungan INSERT dan SELECT
 $hasil = mysql_query("INSERT INTO $tb_tujuan SELECT * FROM $tb_awal
 WHERE $opsi='$isi'");
 // cek status
 if (!$hasil)
 return false; // kembalikan false jika ada error
 else {
 // hapus data dari tabel awal
 $hasil2 = mysql_query("DELETE FROM $tb_awal WHERE $opsi='$isi'");
 // cek status
 if (!$hasil2)
 return false;
 else
 return true;
 }
}

?>

```

Simpan pada direktori **reseller/include** dengan nama **fungsi.php**. Ada beberapa fungsi yang sudah pernah kita tulis pada proyek sebelumnya. Jadi anda tinggal memodifikasi dan menambah fungsi baru. Jangan lupa ganti username, password dan database pada fungsi `konek_db()`.

Ada satu fungsi yang kita buang dari proyek sebelumnya yaitu fungsi `show_tgl()`. Fungsi ini nantinya kita ganti dengan metode baru pada class\_waktu.

## PENJELASAN SCRIPT

Beberapa fungsi baru yang kita tambahkan kali ini antara lain `update()`, `acak_reseller()`, `SQL_array()`, `rupiah()`, `buat_kolom()` dan `pindah_data()`. Kita akan membahasnya satu per satu, kita mulai dari fungsi `update()`.

Pada fungsi `update()` kita memberikan 2 parameter dua-duanya bersifat opsional. Jika tidak ada yang diisi maka secara default fungsi `update()` akan mengupdate nilai transfer. Namun jika parameter pertama kita isi selain 0 maka fungsi ini akan mengupdate nilai traffic yang diisikan pada parameter ke dua.

Fungsi berikutnya adalah `acak_reseller()`. Fungsi ini cukup menarik karena dengan fungsi ini kita dapat mendapatkan username dari member secara acak. Caranya adalah pertama kita lakukan query pada tabel member untuk mendapatkan daftar username.

Kemudian kita gunakan looping while untuk membuat array yang isinya adalah semua username yang ada pada tabel member. Oleh karena itu kita gunakan tanda `[]` pada `$reseller[]` sebagai identitas bahwa data yang kita inginkan berbentuk array. Kemudian kita acak index array dengan `srand(double())` dan `shuffle()`.

Sekarang susunan array `$reseller` yang berisi username sudah teracak. Langkah terakhir adalah mengambil satu username yaitu dengan melakukan looping for. Sebenarnya kita juga dapat menggunakan fungsi array seperti `current()` atau `end()` dalam menunjuk index array. Namun dari hasil perbandingan yang saya lakukan hasilnya kurang random dibanding menggunakan looping for.

Fungsi selanjutnya yaitu `SQL_array()`. Fungsi ini memiliki satu parameter yang harus diisi yaitu query SQL yang dijalankan. Kemudian kita memecah hasil query tersebut dengan `mysql_fetch_array()` dan mengembalikan hasilnya.

Fungsi `rupiah()` bertugas untuk mengubah bentuk dari angka biasa xxxx ke format Rp. x.xxx,-. Untuk melakukan itu kita menggunakan fungsi `number_format()`. Fungsi ini memiliki empat parameter. Untuk menghasilkan string x.xxx parameter yang harus diisi adalah yang pertama. Yaitu string yang akan dirubah formatnya. Kemudian parameter ke-4 yaitu tanda pemisah antar ribuan.

Fungsi berikutnya yaitu fungsi `buat_kolom()`. Seperti namanya tugas fungsi ini adalah membuat kolom. Tujuan pembuatan fungsi ini adalah agar kita tidak perlu menulis berulang-ulang tag `<td></td>` selama looping. Fungsi `buat_kolom()` memiliki tiga parameter. Satu bersifat wajib dan dua opsional.

Parameter pertama merupakan data yang akan kita masukkan pada kolom tabel. Data pada parameter pertama ini berupa array. Parameter kedua adalah jenis kolom apakah header kolom(`th`) atau kolom biasa(`td`). Parameter ketiga kita isi mana kita ingin memasukkan attribut-attribut tambahan pada kolom seperti `align`, `class` dan sebagainya.

Untuk mencetak kolom pertama kita harus pastikan bahwa index array dari data yang akan dipakai dimulai dari awal. Untuk itu kita gunakan fungsi `reset()` yang akan mengembalikan index array dari awal. Kemudian kita membuat sebuah variabel yaitu `$isi` yang isinya adalah index array yang sedang aktif. Untuk mencetak kolom maka kita perlu menggunakan looping dalam kasus ini kita gunakan while.

Pada badan looping kita membuat variabel `$kolom` yang isinya adalah elemen array yang akan dicetak sebagai kolom. Setelah itu kita set index array agar berpindah ke posisi berikutnya.

Fungsi yang terakhir yaitu `pindah_data()`. Fungsi ini berguna untuk memindah data pada tabel yang strukturnya sama. Pada database reseller tabel yang strukturnya sama adalah `tbl_member`, `tbl_pemesan`, dan `tbl_blokir`. Karena pada `ladangbisnis.localhost` nantinya terdapat proses aktivasi dan pemblokiran maka data-data memerlukan beberapa baris kode jika menggunakan cara query biasa.

Jadi untuk mempersingkatnya kita membuat fungsi ini. Fungsi `pindah_data()` memiliki empat parameter yang semuanya harus diisi. Parameter pertama adalah tabel awal, parameter kedua adalah tabel tujuan, parameter ketiga adalah field/kolom, dan yang keempat adalah isi field/kolom.

Untuk memindah data yang strukturnya sama kita hanya perlu melakukan query `INSERT` digabung dengan query `SELECT` dalam satu baris perintah. Contohnya kita akan memindah data dari `tbl_pemesan` ke `tbl_member`,

```
INSERT INTO tbl_member SELECT * FROM tbl_pemesan WHERE kolom='isi';
```

Setelah kita memindah data maka kita hapus data dari tabel sebelumnya menggunakan query DELETE.

File dari direktori include berikutnya yang akan kita buat adalah file *class\_skin.php*. File ini sudah sering kita buat jadi saya rasa anda sudah sangat hafal dengan file ini. Klik **File > New > PHP** pada PHP DESIGNER 2006 lalu ketik kode berikut.

#### Script dari file *class\_skin.php*

```
<?php

/*
** Nama File.....: class_skin.php
** Penulis.....: Rio Astamal
** Tanggal.....: 20-10-2005
** Penjelasan....: Class untuk mengganti tag
*/

class skin
{
 // deklarasikan properti
 var $tag = array();
 var $file_skin;
 var $halaman;

 // metode untuk menentukan nama tag yang diganti
 // dan string penggantinya
 function ganti_tag($namatag, $str_pengganti)
 {
 $this->tag[$namatag] = $str_pengganti;
 }

 // metode untuk menentukan file template yang digunakan
 function ganti_skin($namafile)
 {
 $this->file_skin = $namafile;
 }

 // metode untuk mengganti tampilan
 function ganti_tampilan()
 {
 // buka file template menggunakan file()
 $this->halaman = file($this->file_skin);

 // gabungkan setiap baris dengan implode()
 $this->halaman = implode("", $this->halaman);

 // gunakan looping foreach() untuk mengganti setiap tag
 foreach($this->tag as $str_dicari => $str_baru)
 {
 $this->halaman = ereg_replace($str_dicari, $str_baru, $this->halaman);
 }

 echo $this->halaman; // tampilkan ke layar
 }
}
?>
```

Simpan pada direktori **reseller/include** dengan nama *class\_skin.php*.

Selanjutnya adalah file yang kita gunakan untuk menampilkan menampilkan link nomor halaman. File ini sudah pernah kita buat pada proyek sebelumnya. Hanya saja sekarang

terdapat sedikit perubahan. Buat file PHP baru pada PHP DESIGNER 2006 kemudian ketik kode berikut.

#### Script dari file ***class\_halaman.php***

```
<?php

/*
** Nama File.....: class_halaman.php
** Penulis.....: Rio Astamal
** Tanggal.....: 20-10-2005
** Penjelasan....: Class untuk menampilkan link nomor
** halaman
***** */

// cegah pengaksesan langsung dari browser
if (eregi('class_forum.php', $_SERVER['PHP_SELF']))
{
 // kembalikan ke halaman utama
 header('Location: ../index.php');
 exit; // keluar dari rutin script
}

class halaman
{

 // deklarasikan properti untuk class
 var $tabel, $page, $file, $per_halaman, $jml_data, $jml_hal, $hasil_query;
 var $record, $last_page, $sebelumnya, $berikutnya, $tampilkan_hal;

 // metode untuk memilih tabel yang digunakan
 function set_tabel($nama_tabel)
 {
 $this->tabel = $nama_tabel;
 }

 // metode untuk menentukan data yang di-post
 function set_page($halaman)
 {
 $this->page = $halaman;
 }

 // metode untuk menentukan banyaknya pesan per halaman
 function set_pph($angka)
 {
 $this->per_halaman = $angka;

 return $this->per_halaman;
 }

 // metode untuk mendapatkan jumlah record yang harus ditampilkan
 // pada query SQL. jumlah record = halaman x pesan per halaman
 function get_record()
 {
 $hasil = $this->page * $this->per_halaman;
 $this->record = $hasil;

 return $this->record;
 }

 // metode untuk melakukan query sql. digunakan untuk mendapatkan jumlah
 // data, jumlah record dan sebagainya
 function query_SQL($q = 1, $id='', $isi='', $lainnya='')
 {
```

```

// jika parameter pertama berisi 1 lakukan query berikut
if ($q == 1)
 $query = mysql_query("SELECT * FROM $this->tabel");
else if ($q == 2) // jika 2 lakukan query berikut
 $query = mysql_query("SELECT * FROM $this->tabel WHERE $id='$isi'");
else if ($q == 3) // jika 3 lakukan query berikut
 $query = mysql_query("SELECT * FROM $this->tabel WHERE $id='$isi' ORDER BY
 $lainnya DESC LIMIT $this->record, $this->per_halaman");
else
 $query = mysql_query($q); // jika bukan 1,2 atau 3 lakukan query berikut

$this->hasil_query = $query;

return $this->hasil_query; // kembalikan hasil dari query
}

// metode untuk mendapatkan jumlah data pada database
function get_jml_data()
{
 $jumlah = mysql_num_rows($this->hasil_query);
 $this->jml_data = $jumlah;

 return $this->jml_data;
}

// metode untuk mendapatkan jumlah halaman
function get_jml_hal()
{
 // untuk menghitung jumlah halaman digunakan fungsi ceil
 // dimana >> Jumlah data : pesan per halaman
 // jika hasilnya koma, maka dibulatkan ke atas
 $jumlah = ceil($this->jml_data / $this->per_halaman);

 $this->jml_hal = $jumlah;

 return $this->jml_hal;
}

// metode untuk mendapatkan halaman
// paling awal, sebelumnya, berikutnya dan paling akhir
function set_hal()
{
 // halaman terakhir kita kurangi satu karena pada nomor
 // karena nilai jml_hal lebih besar 1 dari halaman terakhir
 $this->last_page = $this->jml_hal - 1;

 // link halaman sebelumnya didapat dengan mengurangi nilai
 // halaman sekarang(yang aktif) dengan satu
 $this->sebelumnya = $this->page - 1;

 // link halaman berikutnya didapat dengan menambahkan nilai
 // halaman sekarang dengan satu
 $this->berikutnya = $this->page + 1;
}

// metode untuk menampilkan halaman
function show_page($URL)
{
 // jika jumlah halaman lebih dari satu tampilkan selain itu jangan
 if ($this->jml_hal > 1)
 {
 // jika halaman sekarang 0 (paling awal) jangan tampilkan link
 // first dan before ganti dengan tulisan biasa(warna abu-abu)
 if ($this->page == 0)

```

```

 {
 $first = "<<" ;
 $back = "<" ;
 }
 else
 {
 $first = "<<" ;
 $back = "sebelumnya'><" ;
 }

 // jika halaman sekarang sama dengan nilai halaman terakhir
 // jangan tampilkan link last dan next
 if ($this->page == $this->last_page)
 {
 $last = ">>" ;
 $next = ">>" ;
 }
 else
 {
 $last = "last_page'>>>" ;
 $next = "berikutnya'>>>" ;
 }

 // tampilkan
 $halaman = "$first $back \n" ;

 // gunakan looping untuk menampilkan setiap nomor halaman
 for ($i=0; $i<$this->jml_hal; $i++)
 {
 // jika nomor halaman sama dengan halaman yang sedang
 // dibuka tebalkan angka tersebut dan hapus link
 // nomor halaman kita tambah 1 agar nomor awal tidak 0
 if ($i == $this->page)
 $halaman .= " ".intval($i + 1)
 ."" ;
 else
 $halaman .= " ".intval($i + 1). " \n" ;
 }

 $halaman .= " $next $last\n" ;

 // output dari halaman jika jumlah halamannya lebih dari satu
 // kurang lebih seperti berikut, dimana x adalah nomor halaman
 // << < x x x > >>
}
else
 $halaman = '' ; // artinya halaman yang ada hanya satu

$this->tampilkan_hal = $halaman;

// kembalikan nilai karena akan dicetak ke layar
return $this->tampilkan_hal;
}

} // akhir dari class halaman
?>

```

Simpan pada direktori **reseller/include** dengan nama **class\_halaman.php**. Yang berubah dari file ini hanyalah tulisan seperti << First < Before xxx Next > Last >> kita ganti dengan yang lebih sederhana yaitu << < xxx > >>. Itu saja? Ya itu saja.

File berikutnya adalah **class\_waktu.php**. File ini juga pernah kita buat sebelumnya namun disini

akan ada penambahan baru. Klik **File > New > PHP** kemudian ketik kode berikut.

Script dari file ***class\_waktu.php***

---

```
<?php
```

```

** Nama File.....: class_waktu.php ***
** Penulis.....: Rio Astamal ***
** Tanggal.....: 20-10-2005 ***
** Penjelasan....: class untuk memanipulasi tanggal ***
** baik dikurangi,ditambah atau tetap ***

```

```
class waktu
{
 // deklarasikan properti
 var $date, $bulan, $year, $jml_hari, $is_kabisat, $mode;

 // metode untuk memberi nilai tanggal, bulan, tahun, jumlah
 // hari dalam bulan sekarang
 function set_date()
 {
 $this->date = date('d'); // tanggal 01-28/29/30/31
 $this->bulan = date('m'); // bulan 01-12
 $this->year = date('Y'); // tahun 4 digit
 $this->jml_hari = date('t'); // jumlah hari
 $this->is_kabisat = date('Y') % 4; // jika 0 = kabisat selain itu tidak
 }

 // metode untuk menentukan tanggal dikurangi atau ditambah
 // 0 = mengurangi, 1 = menambah, 2 = tetap
 function set_mode($pilihan)
 {
 $this->mode = $pilihan;
 }

 // metode untuk menambah, mengurangi tanggal atau tetap
 // format 0 = tanggal biasa, 1 = detik(timestamp)
 function set_tgl($format = 0, $angka = 1)
 {

 if ($angka > 31)
 return "Error: Angka terlalu tinggi(max. 31), keluar dari fungsi.";

 /* ----- kode tambahan ----- */
 $this->angka = $angka;
 /* ----- *-*-*-*-*-* -----*/

 $bulan = $this->bulan;
 $thn = $this->year;

 if ($this->mode == 0) // jika mode pengurangan
 {
 $tgl = $this->date - $angka; // tanggal dikurangi

 if ($tgl <= 0) // jika kurang/sama dengan nol, maka bulan sebelumnya
 {
 $bulan = $this->bulan - 1; // kurangi bulan dengan 1

 if ($this->jml_hari == 30) // bulan yang tanggalnya sampai 30
 $tgl = (31 + $this->date) - $angka; // tanggal pada bulan yang tanggalnya
31
 else if ($this->bulan == 1) // januari

```

```

{
 $tgl = (31 + $this->date) - $angka; // tanggal pada bulan DESEMBER
 $bulan = 12; // desember
 $thn = $this->year - 1; // tahun dikurangi 1
}
else if ($this->bulan == 3) // maret
{
 if ($this->is_kabisat == 0) // jika kabisat
 $tgl = (29 + $this->date) - $angka; // tanggal pada februari
 else
 $tgl = (28 + $this->date) - $angka; // tanggal pada februari non kabisat
}
else if ($this->bulan == 8) // Agustus
 $tgl = (31 + $this->date) - $angka; // tanggal pada bulan juli
else if ($this->bulan == 2) // februari
 $tgl = (31 + $this->date) - $angka; // tanggal pada februari
else
 $tgl = (30 + $this->date) - $angka; // tanggal pada bulan yang tanggalnya
30
}
}
else if ($this->mode == 1)
{
 $tgl = $this->date + $angka; // angka ditambah

 if ($this->jml_hari == 31 && $bulan != 12) // bulan bertanggal 31 selain
desember
 {
 if ($tgl > 31) // jika melebihi tanggal 31
 {
 $tgl = ($this->date - 31) + $angka; // kurangi 31 dan tambah sesuai
parameter
 $bulan = $bulan + 1; // bulan ditambah 1
 }
 }
 else if ($this->jml_hari == 30) // bulan bertanggal 30
 {
 if ($tgl > 30)
 {
 $tgl = ($this->date - 31) + $angka;
 $bulan = $bulan + 1;
 }
 }
 else if ($bulan == 12) // bulan desember
 {
 if ($tgl > 31) // jika melebihi tanggal 31
 {
 $tgl = ($this->date - 31) + $angka;
 $bulan = 1; // bulan kita set januari
 $thn = $thn + 1; // tahun kita tambah satu
 }
 }
 else if ($bulan == 2) // jika bulan februari
 {
 if ($this->is_kabisat == 0) // jika kabisat
 {
 if ($tgl > 29) // jika lebih dari tanggal 29
 {
 $tgl = ($this->date - 29) + $angka;
 $bulan = $bulan + 1; // bulan ditambah 1
 }
 }
 else
 if ($tgl > 28) // jika non kabisat

```

```

 {
 $tgl = ($this->date - 28) + $angka;
 $bulan = $bulan + 1;
 }
 }
else
 $tgl = $this->date; // tetap

if (strlen($tgl) < 2)
 $tgl = "0$tgl"; // agar berformat 2 digit

if (strlen($bulan) < 2)
 $bulan = "0$bulan"; // agar berformat 2 digit

// cek format yang diinginkan
if ($format == 0 || $format != 1)
 $this->hasil = "$tgl - $bulan - $thn"; // tanggal biasa
else
 $this->hasil = mktime(0, 0, 0, $bulan, $tgl, $thn); // detik

return $this->hasil; // kembalikan hasil
} // akhir dari metode set_tgl()
} // akhir dari class waktu

// class untuk menampilkan deadline hari
// merupakan turunan dari class waktu
class deadline extends waktu
{
 var $the_day;

// metode untuk menampilkan deadline atau hari biasa.
function show_deadline()
{
 // buat array untuk nama hari
 $nm_hari = array("Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jumat",
 "Sabtu");

 // buat array untuk nama bulan
 $nm_bulan = array("Januari", "Februari", "Maret", "April", "Mei", "Juni",
 "Juli", "Agustus", "September", "Oktober", "Nopember", "Desember");

 // pecah isi dari $this->hasil
 $pecahan = explode(" - ", $this->hasil);
 $tgl = intval($pecahan[0]); // tanggal
 $bln = $pecahan[1] - 1; // bulan (dikurangi 1 agar index awal 0)
 $thn = $pecahan[2]; // tahun

 // set maksimal deadline 7 hari
 if ($this->angka > 7)
 return "Error: Maksimal deadline adalah 7 hari.';

 // hari dalam angka, 0=minggu, 1=senin, dst.
 $int_hari = date('w');

 if ($this->mode == 1)
 $int_hari = $int_hari + $this->angka;

 // jika $int_hari melebihi 6
 if ($int_hari > 6)
 $int_hari = (date('w') - 7) + $this->angka; // agar angka kembali ke awal
}

```

```

// isi nilai dari $hari dengan array $nm_hari berdasarkan $int_hari
$hari = $nm_hari[$int_hari];
// isi nilai dari $bulan dengan array $nm_bulan berdasarkan $bln
$bulan = $nm_bulan[$bln];
$tahun = $thn;

// susun format tampilan
$day = "$hari, $tgl $bulan $tahun";

// kembalikan hasil
return $this->the_day = $day;

} // akhir dari show_deadline();

} // akhir dari class deadline

?>

```

---

Simpan pada direktori **reseller/include** dengan nama **class\_waktu.php**. Perubahan ada pada metode set\_tgl(). Dimana kita menambahkan kode berikut

```

/* ----- kode tambahan ----- */
$this->angka = $angka;
/* ----- *-*-*-*-*-* ----- */

```

### PENJELASAN SCRIPT

Selain perubahan diatas kita juga menambahkan class baru pada file ini yaitu class deadline. Class ini merupakan turunan dari class waktu. Jadi semua fungsionalitas dari class waktu dimiliki oleh class deadline namun tidak sebaliknya. Tujuan pembuatan class deadline adalah untuk menciptakan format tanggal Hari, dd bulan yyyy.

Dalam menjalankan kerjanya class deadline meminjam semua metode dari class waktu. Class deadline hanya memiliki satu metode asli yaitu show\_deadline(). Pertama kita membuat array yang berisi hari dari Minggu hingga Sabtu dan kita masukkan pada variabel \$nm\_hari. Selanjutnya kita membuat array untuk nama bulan dan kita simpan pada variabel \$nm\_bulan.

Untuk mendapatkan tanggal maka kita mendapatkannya dari metode set\_tgl(). Namun karena kita dalam class maka yang kita ambil adalah \$this->hasil. Format tanggal dari \$this->hasil adalah format dalam angka dan spasi yaitu dd – mm – yyyy. Jadi kita perlu untuk memecahnya jika ingin mendapatkan tanggal, seperti berikut ini

```
$pecahan = explode(" - ", $this->hasil);
```

Setelah itu kita buat variabel \$tgl, \$bln, dan \$thn untuk menampung hasil pecahannya. Khusus untuk \$tgl kita gunakan intval() karena akan kita lakukan perhitungan jadi kita ubah menjadi integer.

Kemudian kita mengecek nilai dari \$this->angka. Yang merupakan isi dari parameter ke dua pada metode set\_tgl(). Kita set maksimal 7 karena jika melebihi 7 kalkulasi hari akan sulit dilakukan. Variabel \$int\_hari menyimpan hasil dari fungsi date('w') yang mengembalikan angka sebagai pengenal hari 0 untuk minggu, 1 untuk senin dan seterusnya.

Bagaimana jika nilai \$int\_hari melebihi 6 yang tidak terdefinisi?. Oleh karena itu kita gunakan statemen if berikut untuk menghandelnya.

```
if ($int_hari > 6)
 $int_hari = (date('w') - 7) + $this->angka;
```

Jika melebihi 6 maka agar nilai kembali ke antara 0 – 6 diperlukan kalkulasi seperti diatas. Contoh jika saat ini tanggal Sabtu, 22 Oktober 2005 dan kita ingin deadlinenya tiga hari ke depan maka deadlinenya haruslah Selasa, 25 Oktober 2005. Tanpa statemen if kalkulasinya

adalah sebagai berikut,

```
hari = 6 + 3 (sabtu = 6) ==> hasil 9
```

hasilnya adalah 9 sedangkan index hari hanya 0 sampai 6. Jadi hasilnya adalah NULL atau tidak terdefinisi. Dengan statemen if diatas kalkulasinya akan menjadi seperti dibawah ini,

```
hari = (6 - 7) + 3 ==> hasil 2
```

Nilai 2 dalam index hari adalah Selasa. Jadi terbukti dengan kalkulasi diatas deadline hari dapat ditentukan. Selanjutnya kita tinggal memasukkan \$int\_hari ke array \$nm\_hari untuk mendapatkan hari yang diinginkan.

Berikutnya file yang akan kita buat adalah file yang menyimpan kumpulan variabel untuk halaman non-member. Untuk membuatnya klik menu **File > New > PHP** pada PHP DESIGNER 2006 untuk membuat file PHP baru, lalu ketik kode berikut.

#### Script dari file ***var\_utama.php***

---

```
<?php

/*
 * NamaFile: var_utama.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk menyimpan variabel yang dipakai seluruh halaman
 * non member
 */

//cegah pengaksesan langsung dari browser
if (eregi('var_utama.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php');
 exit;
}

// panggil file-file yang diperlukan
include ('include/fungsi.php');
include ('include/class_waktu.php');
include ('include/konfig.php');

konek_db(); // koneksikan PHP dengan MySQL server

// jika anda ingin adil anda dapat menggunakan cara berikut
// untuk menentukan reseller yang mendapat komisi jika user mengunjungi
// ladangbisnis.localhost tanpa sponsor
// namun jika tidak anda dapat menentukan user mana yang mendapat komisi
// jika ada pengunjung yang datang tanpa sponsor

// (NAMANYA JUGA BISNIS YANG DICARI YA KEUNTUNGAN!!!)

// cek isi dari res pada URL untuk menentukan reseller
// jika res tidak diset/kosong
$res = filter_str($_GET['res']);
if ($res == '')
{
 // panggil fungsi acak_reseller() untuk mengambil username
 // secara acak/random
 $res = acak_reseller();
}
else
{
 // jika isi res tidak kosong namun username tidak terdaftar
 $hasil = mysql_query("SELECT * FROM tbl_member WHERE username='$res'");
```

```

if (mysql_num_rows($hasil) == 0)
 $res = acak_reseller();
}

// dapatkan nama dan kota dari reseller (berupa array)
$data_res = SQL_array("SELECT * FROM tbl_member WHERE username='$res'");

// buat session untuk menampung username
$_SESSION['reseller'] = $res;

// panggil class deadline (turunan waktu)
$waktu = new deadline;
$waktu->set_date();
$waktu->set_mode(2); // tanggal tetap
$waktu->set_tgl(); // format tanggal biasa => Hari, TGL BULAN TAHUN
// variabel untuk menampilkan hari ini
$hari_ini = $waktu->show_deadline();

$dead = new deadline;
$dead->set_date();
$dead->set_mode(1); // tanggal ditambah
$dead->set_tgl(0, $deadline); // ada di konfig.php
// variabel untuk menampilkan deadline hari
$hari_deadline = $dead->show_deadline();

// variabel untuk menyimpan tampilan link menu
$menu =
<table border='0' cellpadding='4' width='100%' bgcolor="#ba7b31'>
 <tr><td align='center'>
 H O M E</td>
 <td align='center'>
 PESAN</td>
 <td align='center'>
 TESTIMONIAL</td>
 <td align='center'>
 LOGIN</td>
 <td align='center'>
 RESELLER</td>
 <td align='center'>
 BONUS</td>
 <td align='center'>
 KONTAK</td></tr>
</table>";

$reseller =
<table border='0' cellpadding='4' width='100%'>
 <tr><td>
 <p class='sf' align='right'>
 Reseller: $data_res[nama] - $data_res[kota]</p>
 </tr></td>
</table>";

$bonus =
<table border='0' cellpadding='4' width='100%'>
 <tr><td bgcolor="#ba7b31">
 E-BOOK GRATIS!
 </tr></td>
</table>
<form action='bonus.php?res=$res&proses=ebook' method='post'>
<table border='0' cellpadding='4'>
 <tr><td colspan='2' class='putdih'>
 <p class='sf'>Dapatkan e-book tentang bagaimana membangun mailing list
 berskala global dengan cepat dan mudah secara GRATIS!!.</p>
 <p class='sf'>Cukup isikan nama dan email anda di form berikut.</p>
 </td>
</tr>
</table>

```

```

Alamat link download akan kami kirim ke email anda saat ini juga!.</p>
</td></tr>
<tr>
<td>Nama: </td>
<td><input type='text' name='nama'></td></tr>
<tr>
<td>Email: </td>
<td><input type='text' name='email'></td></tr>
<tr>
<td></td><td><input type='submit' value='Kirim'></td></tr>
</table>
</form>";

// variabel untuk profil pemilik website
$me = "
<table border='0' cellpadding='4' width='100%'>
<tr><td bgcolor='#ba7b31'>
PROFIL ADMIN
</tr></td>
</table>
<table border='0' cellpadding='4'>
<tr><td>
<p class='sf'>Rio Astamal
Administrator of
ladangbisnis.localhost

Jl. Ganti Dengan AL AD
Surabaya 63210

Call: 0808080808</p>
</td></tr>
</table>";

// variabel untuk footer
$footer = "© Copyright 2005 ladangbisnis.localhost";

// variabel untuk menyimpan link kembali
$kembali = '

<< Kembali' ;

?>

```

Simpan pada direktori **reseller\include** dengan nama ***var\_utama.php***.

File yang akan kita buat selanjutnya adalah file yang menyimpan kumpulan variabel yang digunakan untuk halaman member. Buat file PHP baru kemudian ketik kode berikut.  
Script dari file ***member\_var.php***

```

<?php

/*
** NamaFile: member_var.php
** Penulis : Rio Astamal
** Tanggal : 25-10-2006
** Fungsi : menyimpan variabel yang dipakai pada semua halaman
** member area
*/
//cegah pengaksesan langsung dari browser
if (eregi('var_utama.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php');
 exit;
}

// user yang sedang login
$user = $_SESSION['member'];

// panggil file-file yang diperlukan

```

```

include ('../include/fungsi.php');
include ('../include/class_waktu.php');
include ('../include/konfig.php');

koneksi_db(); // koneksikan PHP dengan MySQL server

$menu = "


```

```

<tr><td class='sf'>Jumlah Prospek</td>
 <td class='sf'>: $jml_prospek user.</td></tr>
<tr><td class='sf'>Komisi Anda</td>
 <td class='sf'>: ".rupiah($komisi)."</td></tr>
<tr><td colspan='2' class='sf'>
Jumlah kunjungan ke website anda:
$traj[1] kali.</td></tr>
<tr><td bgcolor='#ba7b31' height='25' colspan='2'></td></tr>
</table>";

// variabel yang berisi javascript untuk menampilkan kotak dialog
$java_konfirm =
<script language="javascript">
function konfirmasi(pesan)
{
 var tanya = confirm(pesan);

 return tanya;
}
</script>';

?>

```

Simpan pada direktori **reseller\include** dengan nama **member\_var.php**.

Masih berupa file yang menyimpan kumpulan variabel. Kali ini file yang akan kita buat adalah file yang menyimpan kumpulan variabel untuk halaman administrator. Buat file PHP baru pada PHP DESIGNER 2006 kemudian ketik kode berikut.

Script dari file **admin\_var.php**

```

<?php

/*
 * NamaFile: admin_var.php
 * Penulis : Rio Astamal
 * Tanggal : 26-10-2006
 * Fungsi : menyimpan variabel halaman administrator
 */

//cegah pengaksesan langsung dari browser
if (eregi('admin_var.php', $_SERVER['PHP_SELF']))
{
 header('Location: ../index.php');
 exit;
}

// panggil file-file yang diperlukan
include ('../include/fungsi.php');
include ('../include/class_waktu.php');
include ('../include/konfig.php');

koneksi_db(); // koneksikan PHP dengan MySQL server

// panggil class deadline (turunan waktu)
$waktu = new deadline;
$waktu->set_date();
$waktu->set_mode(2); // tanggal tetap
$waktu->set_tgl(); // format tanggal biasa => Hari, TGL BULAN TAHUN
// variabel untuk menampilkan hari ini
$hari_ini = $waktu->show_deadline().' '.date('H:i');

$menu =
<table border='0' cellpadding='4' width='100%' bgcolor='#ba7b31'>
 <tr><td align='center'>
```

```

 H O M E</td>
<td align='center'>
 PENCARIAN</td>
<td align='center'>
 KIRIM EMAIL</td>
<td align='center'>
 TESTIMONIAL</td>
<td align='center'>
 LOGOUT</td></tr>
</table>";

// variabel untuk footer
$footer = "© Copyright 2005 ladangbisnis.localhost";

// variabel untuk menyimpan link kembali
$kembali = '

<< Kembali';

// variabel yang menyimpan statistik website
$q = "SELECT * FROM ";
$jml_pemesan = mysql_num_rows(mysql_query($q."tbl_pemesan"));
$jml_member = mysql_num_rows(mysql_query($q."tbl_member"));
$jml_blokir = mysql_num_rows(mysql_query($q."tbl_blokir"));
$jml_prospek = mysql_num_rows(mysql_query($q."tbl_prospek"));

// hitung jumlah traffic ke ladangbisnis.localhost
$hasil = mysql_query("SELECT * FROM tbl_traffic");
while ($data = mysql_fetch_array($hasil))
{
 $web_traffic += $data['traffic']; // total traffic user
}

$komisi = 50000 * $jml_member; // komisi yang diterima member

$web_stat =
<table border='0' cellpadding='4'>
<tr><td bgcolor='#ba7b31' class='putih' colspan='2'>
 STATISTIK WEBSITE</td></tr>
<tr><td colspan='2'>
 <p class='sf'>Berikut ini adalah statisk dari website
 ladangbisnis.localhost.</p></td></tr>
<tr><td class='sf'>Jumlah pemesan</td>
 <td class='sf'>: $jml_pemesan user.</td></tr>
<tr><td class='sf'>Jumlah member</td>
 <td class='sf'>: $jml_member user.</td></tr>
<tr><td class='sf'>Jumlah terblokir</td>
 <td class='sf'>: $jml_blokir user.</td></tr>
<tr><td class='sf'>Jumlah Prospek</td>
 <td class='sf'>: $jml_prospek user.</td></tr>
<tr><td class='sf'>Profit Anda</td>
 <td class='sf'>: ".rupiah($komisi)."</td></tr>
<tr><td colspan='2' class='sf'>Jumlah kunjungan ke
 http://ladangbisnis.localhost:
 $web_traffic kali.</td></tr>
<tr><td bgcolor='#ba7b31' height='25' colspan='2'></td></tr>
</table>";

?>

```

Simpan pada direktori **reseller\include** dengan nama **admin\_var.php**.

File terakhir pada direktori *include* yang akan kita buat adalah file yang menyimpan template untuk keperluan pengiriman email. Seperti biasa klik menu **File > New > PHP** pada PHP DESIGNER 2006 lalu ketik kode berikut.

### Script dari file ***var\_mail.php***

---

```
<?php

/*
 * NamaFile: var_mail.php
 * Penulis : Rio Astamal
 * Tanggal : 26-10-2006
 * Fungsi : menyimpan variabel yang digunakan untuk mengirim emeil
 */

// cegah pengaksesan langsung dari browser
if (eregi('var_mail.php', $_SERVER['PHP_SELF']))
{
 header('Location: index.php');
 exit();
}

// variabel untuk isi email konfirmasi pendaftaran user
$mail_daftar = "
Saudara/i {NAMA_U}, \n
Terima kasih atas pendaftaran anda untuk menjadi member
ladangbisnis.localhost. Supaya Anda dapat segera masuk ke member area
dan mendownload produk kami. Maka segeralah mentransfer dana sebesar
{RUPIAH} x2 yang masing-masing ditujukan kepada Saya selaku
pemilik program dan Reseller Anda. Berikut ini adalah data profilnya.\n
PROFIL PENGELOLA
=====
Nama : Rio Astamal
Transfer: {RUPIAH}
Bank : BANK Virtual - Rek. 0987654321 - a.n Rio Astamal
Telp./HP: 0808080808
Email : admin@ladangbisnis.localhost
=====\\n
PROFIL RESELLER
=====
Nama : {NAMA_R}
Transfer: {RUPIAH}
Bank : {BANK}
Telp./HP: {TELPON}
Email : {EMAIL}
=====\\n
Karena minat dan respon untuk e-book ini meningkat pesat. Maka harga
Rp. 100.000,- akan segera saya naikkan menjadi Rp. 200.000,-. Kemungkinan
harga akan saya naikkan pada hari $waktu_naik. Saat ini saya hanya
melayani 11 pemesan saja.\n
Oleh karena itu, jika anda saat ini telah memesan maka segeralah
melakukan transfer. Agar anda tetap memperoleh harga terendah yaitu
Rp. 100.000,-\\n\\n
=====
Rio Astamal - Administrator
http://ladangbisnis.localhost
=====";;

// variabel untuk isi email member/terblokir yang dihapus admin
$mail_del = "
Saudara {NAMA}, \n
Karena saudara telah melakukan pelanggaran terhadap ketentuan-ketentuan
yang telah disepakati bersama. Maka dengan terpaksa keanggotaan saudara
di ladangbisnis.localhost kami hapus.\n
Dengan demikian saudara tidak lagi dapat masuk member area dan mengikuti
program reseller kami. Jika anda ingin kembali dapat masuk ke member area
maka anda harus mendaftar kembali sebagai member baru.\n\\n
=====
```

```

Rio Astamal - Administrator
http://ladangbisnis.localhost
=====";

// variabel untuk isi email member yang diblokir admin
$mail_blokir = "
Saudara {NAMA},\n
Karena anda telah melakukan beberapa hal yang menurut saya (administrator)
merugikan member lain atau website ladangbisnis.localhost. Maka untuk
sementara waktu keanggotaan anda kami blokir.\n
Dengan demikian anda tidak dapat masuk ke member area dan juga tidak
dapat mengikuti program reseller ladangbisnis.localhost.\n
Jika anda ingin keanggotaan anda diaktifkan kembali, maka reply email ini.
Setelah keanggotaan anda diaktifkan kembali, informasinya akan saya
kirim ke email anda.\n
=====
 Rio Astamal - Administrator
 http://ladangbisnis.localhost
=====";

// variabel untuk isi email (Blokir yang dilakukan reseller dari
// member area)
$mail_res_blokir = "
Saudara/i {NAMA_U},\n
Kami mohon maaf karena keanggotaan anda di ladangbisnis.localhost untuk
sementara kami blokir. Pemblokiran ini dikarenakan anda belum menyelesaikan
pembayaran kepada reseller anda yaitu Saudara/i {NAMA_R}. Namun jika
anda sudah merasa mentransfer ke reseller anda tersebut. Sebaiknya anda
menguhungungi dia untuk konfirmasi.\n
Namun jika anda memang belum mentransfer, segeralah untuk mentransfer agar
keanggotaan anda dapat diaktifkan kembali. Berikut ini adalah data diri
dari reseller anda.\n
DATA DIRI RESELLER
=====
Nama : {NAMA_R}
Username: {USERNAME}
Email : {EMAIL}
Telp/HP : {TELPON}
Bank : {BANK}
=====\
Segeralah mentransfer dana ke reseller anda. Agar keanggotaan anda aktif
dan anda dapat mengikuti kembali program reseller kami.\n
=====
 Rio Astamal (Administrator)
 http://ladangbisnis.localhost
=====";

// variabel untuk member yang diaktivasi ulang oleh admin
$mail_reaktivasi = "
Saudara {NAMA},\n
Reseller anda telah meminta saya selaku pemilik program untuk
mengaktivasi kembali keanggotaan anda. Dengan demikian anda dapat
kembali masuk ke member area.\n
Selain itu anda juga dapat mengikuti kembali program reseller
ladangbisnis.localhost yang sempat diblokir.\n
TERIMA KASIH.
=====
 Rio Astamal (Administrator)
 http://ladangbisnis.localhost
=====";

// variabel untuk member yang diaktivasi oleh admin
$mail_aktivasi = "
Saudara {NAMA},\n

```

Saya ucapkan selamat, karena keanggotaan anda di ladangbisnis.localhost telah diaktifkan. Sekarang anda dapat masuk ke member area dan mendownload produk utama kami.\n

---

```
=====
Username: {USERNAME}
Password: {PASSWORD}
=====\\n
Jika anda menginginkan penghasilan yang instan. Saya sarankan kepada anda untuk mengikuti program reseller kami. Setiap member yang masuk lewat rekomendasi anda. Maka anda akan mendapatkan komisi 50% atau Rp. 50.000,-.
\\nAlamat URL yang anda promosikan adalah:
=====\\n
http://ladangbisnis.localhost?res={USERNAME}\\n
Untuk keterangan lebih lengkap silahkan lihat pada member area.\\n
TERIMA KASIH karena anda telah membeli e-book ini.\\n\\n
=====\\n
Rio Astamal (Administrator)
http://ladangbisnis.localhost
=====";
```

---

```
// variabel untuk isi email pengiriman lupa password
$mail_lupa_pass = "
Saudara {NAMA},\\n
Pada tanggal {DATE} anda meminta kami untuk melakukan reset password anda di ladangbisnis.localhost. Dan berikut ini adalah password baru anda untuk masuk ke member area.\\n
=====\\n
Username: {USERNAME}
Password: {PASSWORD}
=====\\n
Gunakan password baru anda tersebut untuk login dan update kembali password anda agar mudah diingat.\\n
=====\\n
Rio Astamal - Administrator
http://ladangbisnis.localhost
=====";
```

?>

---

Simpan pada direktori **reseller\include** dengan nama **var\_mail.php**. Tag-tag {XXX} pada variabel diatas nantinya akan kita ganti sesuai dengan data yang kita inginkan saat mengirim email.

Akhir dari file yang berada pada direktori **reseller\include**.

---

Untuk direktori berikutnya kita beralih ke direktori *reseller/gambar*. Hanya terdapat dua file pada direktori ini yaitu file *index.php* dan file gambar utama yaitu *banner.jpg*. Untuk membuat file *index.php* klik menu **File > New > PHP** pada PHP DESIGNER 2006 kemudian salin kode berikut.

#### Script dari file *index.php*

---

```
<?php
/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 20-10-2005
** Penjelasan....: Untuk meredirect ke halaman utama
*/
// jika user mencoba masuk ke halaman inc
// kembalikan ia ke halaman utama
```

```

header ("Location: ../index.php");
exit;

?>

```

---

Simpan pada direktori **reseller\gambar** dengan nama ***index.php***.

File berikutnya yaitu *banner.jpg*, seperti yang sudah pernah kita bahas sebelumnya. Anda dapat membuat menggunakan Photoshop atau The Gimp atau program pengolah gambar lainnya. Yang penting adalah ukurannya 798x100 pixel. Meskipun anda tidak membuat file ini pun tidak masalah dan tidak mengganggu jalannya script ladangbisnis.localhost. Hanya saja tampilannya kurang menarik.



Gambar 5.10: File *banner.jpg*

Jika anda tidak bisa atau malas membuatnya cukup download gambarnya di <http://materi.rahasia-webmaster.com/bab5/gambar/>

Akhir dari file yang berada pada direktori reseller\gambar.

---

Selanjutnya file-file yang akan kita buat adalah file yang berada pada direktori *reseller/template*. Pada direktori ini hanya terdapat empat file. File-file ini digunakan sebagai skin template pada semua halaman ladangbisnis.localhost.

Untuk file yang pertama adalah *index.php*. Tugas file ini hanya melakukan redirect halaman saja jika ada yang mencoba masuk ke direktori *reseller/template*. Klik **File > New > PHP** pada PHP DESIGNER 2006, kemudian ketik kode berikut.

#### Script dari file ***index.php***

---

```

<?php

/*
** Nama File.....: index.php
** Penulis.....: Rio Astamal
** Tanggal.....: 20-10-2005
** Penjelasan....: Untuk meredirect ke halaman utama
*/
// jika user mencoba masuk ke halaman inc
// kembalikan ia ke halaman utama
header ("Location: ../index.php");
exit;

?>

```

---

Simpan pada direktori **reseller\template** dengan nama ***index.php***.

Untuk file berikutnya kita akan membuat file CSS. Pada PHP DESIGNER 2006 klik menu **File > New > CSS** kemudian ketik kode berikut.

#### Script dari file ***style.css***

---

```

body{font-family: verdana, arial, sans-serif, font-size: 12px; margin-top: 0;
margin-left: 0; margin-right: 0; margin-bottom: 0}
td,li,th{font-family: verdana, arial, sans-serif; font-size: 12px}

```

---

---

```

table{border-collapse: collapse; border-color: #cgcgcg}
a{color: #ba7b31; text-decoration: none}
a:hover{font-weight: bold; text-decoration: underline}
p{line-height: 17px}
.besar{font-family: verdana, arial, sans-serif; font-size: 18px; text-align: center; line-height: 20px; font-weight: bold; color: #ba7b31}
.a.menu{font-weight: bold; color: #ffffff; text-decoration: none}
.a.menu:hover{text-decoration: underline; color: #ffffff; font-weight: bold}
.a.tabel:hover{text-decoration: underline; font-weight: normal}
.sf{font-family: verdana, arial, sans-serif; font-size: 11px}
.putih{color: #ffffff}
.error{color: red; text-align: center}

```

---

Simpan pada direktori **reseller\template** dengan nama **style.css**.

File berikutnya yang akan kita buat adalah file skin template untuk semua halaman non-member. Klik **File > New > PHP** pada PHP DESIGENR 2006 lalu ketik kode berikut.

Script dari file **index\_skin.php**

---

```

<html>
<head>
<title>Jadikan Internet Ladang Bisnis Paling Menguntungkan</title>

<!-- OPTIMASI META TAG -->
<meta name="keywords" content="peluang bisnis, peluang usaha, lowongan kerja, internet marketing, mlm, bisnis, web programmer, programmer">
<meta name="description" content="Temukan cara paling ampuh untuk mengubah internet menjadi ladang bisnis paling menguntungkan">
<meta name="title" content="ladangbisnis.localhost">
<meta name="author" content="Rio Astamal">
<meta name="publisher" content="ladangbisnis.localhost">
<meta name="owner" content="ladangbisnis.localhost">
<meta name="distribution" content="global">
<meta name="robots" content="index, follow">
<meta name="revisit-after" content="30 days">

<!-- AKHIR META TAG -->

<!-- PANGGIL FILE style.css -->
<link rel="stylesheet" type="text/css" href="template/style.css">

</head>
<body>
<!-- Tabel untuk gambar utama -->
<table border="0" cellpadding="0" width="100%" bordercolor="#000000">
<tr>
<td background="gambar/banner.jpg" width="798" height="100"></td>
</tr>
</table>

<!-- Tabel untuk sales letter disebelah kanan -->
<table border="1" cellpadding="0" width="100%" bordercolor="#000000">
<tr class="putih">
<td width="70%" valign="top">{MENU}</td>
<td width="30%" valign="middle" align="right" bgcolor="#ba7b31">{TODAY}</td>
</tr>
<tr>
<td width="70%" valign="top">{UTAMA}</td>
<td width="30%" valign="top" bgcolor="#dea057">{KANAN}</td>
</tr>
</table>

```

---

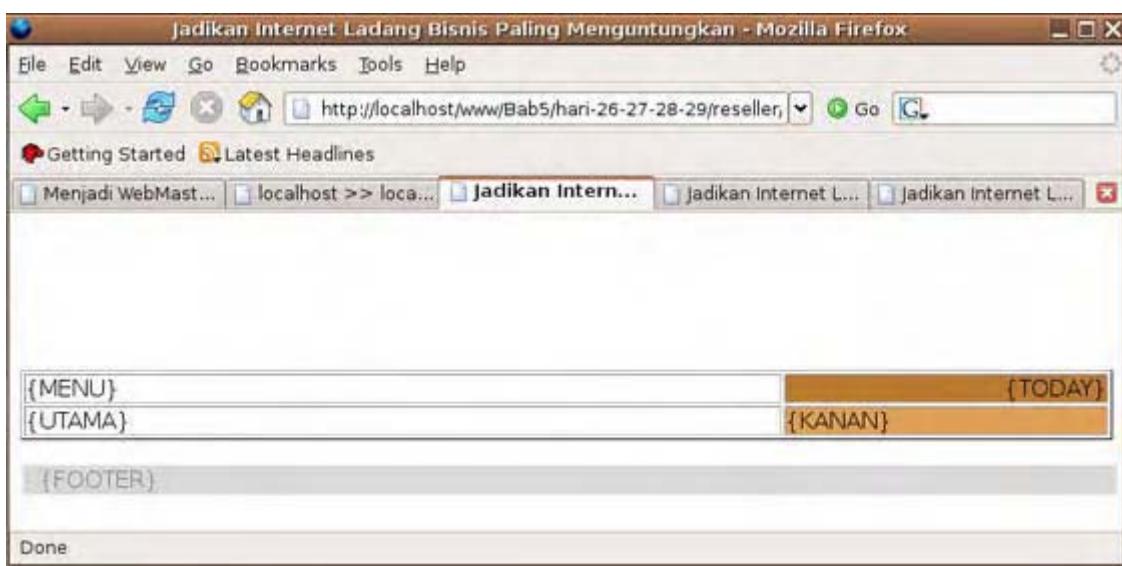
```

<!-- beri jarak -->
<p></p>

<!-- buat tabel untuk footer -->
<table border="0" width="100%">
<tr>
<td bgcolor="#d9d9d9">
<marquee scrolldelay="1" scrollamount="2">{FOOTER}</marquee></td>
</tr>
</table>
</body>
</html>

```

Simpan pada direktori **reseller\index\_skin.php** dengan nama **index\_skin.php**. Untuk optimasi search engine maka kita juga memasukkan beberapa **meta tag** tambahan. Jika anda ingin melihat tampilan awalnya arahkan browser anda ke [http://localhost/webmaster/bab5/hari-26-27-28-29/reseller/template/index\\_skin.php](http://localhost/webmaster/bab5/hari-26-27-28-29/reseller/template/index_skin.php).



Gambar 5.11: Tampilan file index\_skin.php

File template berikutnya kita gunakan untuk halaman-halaman member dan admin. Untuk membuatnya klik **File > New > PHP** lalu ketik kode berikut.

Script dari file **member\_skin.php**

```

<html>
<head>
<title>Jadikan Internet Ladang Bisnis Paling Menguntungkan</title>

<!-- PANGGIL FILE style.css -->
<link rel="stylesheet" type="text/css" href="../template/style.css">

</head>
<body>

<!-- Tabel untuk gambar utama -->
<table border="0" cellpadding="0" bordercolor="#000000">
<tr>
<td background="../gambar/banner.jpg" width="798" height="100"></td>
</tr>
</table>

<!-- Tabel untuk sales letter disebelah kanan -->
<table border="1" cellpadding="0" bordercolor="#000000">

```

---

```

<tr class="putih">
 <td width="30%" valign="middle" align="left" bgcolor="#ba7b31">{TODAY}</td>
 <td width="70%" valign="top">{MENU}</td>
</tr>
<tr>
 <td width="30%" bgcolor="#dea057" valign="top">{KIRI}</td>
 <td width="70%" valign="top">{UTAMA}</td>
</tr>
</table>

<!-- beri jarak -->
<p></p>

<!-- buat tabel untuk footer -->
<table border="0" width="100%">
 <tr>
 <td bgcolor="#d9d9d9">
 <marquee scrolldelay="1" scrollamount="2">{FOOTER}</marquee></td>
 </tr>
</table>
</body>
</html>

```

---

Simpan pada direktori **reseller\template** dengan nama ***member\_skin.php***. Perbedaan file ini dengan file *index\_skin.php* hanya terletak pada lebar kolomnya. Jika pada *index\_skin.php* kolom yang lebih lebar adalah sebelah kiri dan yang sempit adalah sebelah kanan ({TODAY} dan {KANAN}). Maka pada *member\_skin.php* kolom yang lebih lebar adalah yang sebelah kanan dan yang sempit adalah sebelah kiri ({TODAY} dan {KIRI}).

Akhir dari file pada direktori **reseller\template**.

---

Selanjutnya kita akan bekerja pada direktori untuk halaman-halaman administrasi yaitu pada direktori *reseller/\_my\_cp\_*. Karena berhubungan dengan hak/permission maka semua halaman pada direktori ini menggunakan session sebagai pengenal.

Langsung saja kita akan membuat file yang pertama yaitu *index.php*. Jalankan PHP DESIGNER 2006 klik **File > New > PHP** kemudian salin script berikut.

Script dari file ***index.php***

---

```

<?php

/*
 * NamaFile: index.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk membuat form login dan halaman utama admin
 */

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/admin_var.php');

$proses = filter_str($_GET['proses']);
if ($proses == '')
 $proses = 'utama';

// handel setiap kondisi dengan switch dan case
switch ($proses)
{

```

```

/*
case 'utama':
/*
// cek apaka user sudah login atau belum
if (!cek_session('admin'))
{
// kosongkan nilai menu dan web_stat
$menu = '' ; $web_stat = '';
// buat form login
$utama = "

| | |
|---|--|
| <p class="besar">Admin Area - Mohon Login</p> <center> <form action='index.php?proses=proses_form' method='post'> <table border='1' cellpadding='5' width='50%' bordercolor='#cgcgcg'> <tr bgcolor='#ba7b31' align='center'> <td class='putih' colspan='2'>FORM LOGIN</td></tr> <tr bgcolor='#f4f4f4'><td>Username</td> <td><input type='text' name='username'></td></tr> <tr bgcolor='#ffffff'><td>Password</td> <td><input type='password' name='password'></td></tr> <tr><td><td><input type='submit' value='Login'></td></tr> </table></form></center> </td></tr> </table>" ; } else { // tampilkan menu dan statistik dan link untuk manage database \$utama = " <table border="0" cellpadding="5" width="100%"> <tr><td> <p class="besar">Selamat Datang di Admin Area</p> <p>Silahkan klik pada tabel berikut untuk melihat daftar membernya.</p> <table border='1' cellpadding='0' width='50%'> <tr><td> !cek_session()="" >="" >\$jml_blokir="" >\$jml_member="" >\$jml_pemesan="" >\$jml_prospek="" >".rupiah(\$komisi)."<="" ><td>jumlah="" <="" <table="" <td><a="" <td>jumlah<="" <td>statistik<="" <td><font="" <tr="" ;="" <="" <tr="" a><="" akhir="" b><="" bgcolor="#f4f4f4" border="0" break;="" cellpadding="5" class="putih" color="#ba7b31" dari="" else="" font><="" height="30" href="prospek.php" member<="" pemesan<="" pre="" profit<="" prospek<="" table>="" table>"="" td>="" td><="" terblokir<="" tr>="" user<="" width="100%" }=""> </tr><td>></td></tr></table></center> | <p class="besar">Selamat Datang di Admin Area</p> <p>Silahkan klik pada tabel berikut untuk melihat daftar membernya.</p> <table border='1' cellpadding='0' width='50%'> <tr><td> !cek_session()="" >="" >\$jml_blokir="" >\$jml_member="" >\$jml_pemesan="" >\$jml_prospek="" >".rupiah(\$komisi)."<="" ><td>jumlah="" <="" <table="" <td><a="" <td>jumlah<="" <td>statistik<="" <td><font="" <tr="" ;="" <="" <tr="" a><="" akhir="" b><="" bgcolor="#f4f4f4" border="0" break;="" cellpadding="5" class="putih" color="#ba7b31" dari="" else="" font><="" height="30" href="prospek.php" member<="" pemesan<="" pre="" profit<="" prospek<="" table>="" table>"="" td>="" td><="" terblokir<="" tr>="" user<="" width="100%" }=""> </tr><td>> |
| <p class="besar">Selamat Datang di Admin Area</p> <p>Silahkan klik pada tabel berikut untuk melihat daftar membernya.</p> <table border='1' cellpadding='0' width='50%'> <tr><td> !cek_session()="" >="" >\$jml_blokir="" >\$jml_member="" >\$jml_pemesan="" >\$jml_prospek="" >".rupiah(\$komisi)."<="" ><td>jumlah="" <="" <table="" <td><a="" <td>jumlah<="" <td>statistik<="" <td><font="" <tr="" ;="" <="" <tr="" a><="" akhir="" b><="" bgcolor="#f4f4f4" border="0" break;="" cellpadding="5" class="putih" color="#ba7b31" dari="" else="" font><="" height="30" href="prospek.php" member<="" pemesan<="" pre="" profit<="" prospek<="" table>="" table>"="" td>="" td><="" terblokir<="" tr>="" user<="" width="100%" }=""> </tr><td>> | |


```

```

***** */
***** */
case 'pros_form':
***** */

// ambil data yang dipost
$username = filter_str($_POST['username'], "\^"); // ^ diperbolehkan
$password = filter_str($_POST['password']);
// enkripsi password
$password = balik_md5($password);

if (!login('tbl_admin', $username, $password))
{
 // kosongkan menu dan statistik
 $menu = ''; $web_stat = '';
 $utama = "<p class='error'>Username atau password salah.$kembali</p>";
}
else
{
 // buat session untuk user ini
 $_SESSION['admin'] = $username;
 header('Location: index.php');
}

break;
***** */

***** */
case 'logout':
***** */

if (!logout('admin'))
 $utama = "<p class='error'>Anda belum login.
 Login";
else
{
 $utama = "<p class='error'>Anda telah logout dari sistem.
 Login kembali.";
}
break;
***** */

} // akhir dari switch

$skin = new skin; // buat objek skin
$skin->ganti_skin('../template/member_skin.php'); // tentukan file template
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $web_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **reseller\\_my\_cp\_** dengan nama **index.php**. Jika admin belum login otomatis session belum terisi jadi yang ditampilkan adalah form login. Namun jika session sudah terisi berarti yang ditampilkan adalah halaman utama admin area.



Gambar 5.12: Tampilan halaman utama admin area

Untuk login gunakan username `_^_super_user_^_` dan untuk password gunakan `_super_pass_`. Username dan password ini sesuai dengan apa yang kita masukkan pada file db\_reseller.sql. Hanya saja string chiper pada file db\_reseller.sql saya peroleh dengan menampilkan dulu hasil enkripsi. Menggunakan echo balik\_md5(`_^_super_user_^_`) dan echo balik\_md5(`_super_pass_`) kemudian kita tinggal copy and paste saja.

Variabel seperti \$jml\_pemesan, \$jml\_member dan lainnya. Sudah kita set nilainya pada file admin\_var.php, jadi kita tinggal memanggilnya lewat include(). Tampilan dari halaman utama admin area dapat anda lihat pada gambar 5.12.

Untuk melihat statistik lengkap daftar user seperti tanggal join, reseller, transfer dan lain sebagainya. Anda dapat mengklik link x user pada tabel statistik.

Pada kolom sebelah kiri untuk halaman-halaman administrator. Sebagian besar nantinya akan ditampilkan tentang statistik website ladangbisnis.localhost. Statistik tersebut meliputi banyaknya pemesan e-book, banyaknya member yang telah bergabung, banyaknya prospek, banyaknya member yang terblokir dan jumlah kunjungan ke ladangbisnis.localhost.

File berikutnya yang akan kita buat adalah file yang akan menampilkan daftar statistik dari pemesan, member, terblokir, dan prospek. Untuk membuatnya klik menu **File > New > PHP** untuk membuat file baru kemudian ketik script berikut.

#### Script dari file **statistik.php**

```
<?php

/*
 * NamaFile: statistik.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : menampilkan daftar statistik user
 */

session_start();
```

```

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/admin_var.php');
include ('../include/class_halaman.php');

// cek apakah user sudah login atau belum
if (!cek_session('admin'))
{
 header('Location: index.php');
 exit;
}
else
{

// ambil data dari URL
$page = filter_str($_GET['page']);
if ($page == '')
 $page = 0;

// kosongkan hari dan statistik agar tabel lebih lebar
$hari_ini = ''; $web_stat = '';

// ambil data yang dipost
$tabel = filter_str($_GET['tabel']);
if ($tabel == '')
 $tabel = 'tbl_pemesan';

// untuk tabel pemesan dan terblokir (tampilan sama)
$chk_name = "aktivasi[]"; // nama checkbox
$tombol = "Aktivasi"; // label Tombol
$status = "Aktivasi?"; // label pada header kolom
$order_by = "transfer"; // untuk query SQL selain tbl_prospek

// cek nama tabel
if ($tabel == 'tbl_member')
{
 $jdl = "
 <p class='besar'>Daftar Member</p>
 <p>Total member yang sudah bergabung: $jml_member user.

 Statistik berikut menunjukkan daftar member ladangbisnis.locahost</p>";
 $chk_name = "blokir[]";
 $tombol = "Blokir";
 $status = "Blokir?";
 $form = "proses_db.php?proses=member";
}

else if ($tabel == 'tbl_pemesan')
{
 $jdl = "
 <p class='besar'>Daftar Pemesan</p>
 <p>Total pemesan e-book: $jml_pemesan user.

 Statistik berikut menunjukkan daftar pemesan e-book \"Internet
 Sebagai Ladang Bisnis\".</p>";
 $form = "proses_db.php?proses=pemesan";
}

else if ($tabel == 'tbl_blokir')
{
 $jdl = "
 <p class='besar'>Daftar Terblokir</p>
 <p>Total member yang keanggotaannya diblokir: $jml_blokir user.

 Statistik berikut menunjukkan daftar member ladangbisnis.locahost
 yang keanggotaannya diblokir.</p>";
 $form = "proses_db.php?proses=terblokir";
}

else if ($tabel == 'tbl_prospek')

```

```

{
 $jdl = "
 <p class='besar'>Daftar Prospek</p>
 <p>Total prospek yang berhasil didapat: $jml_prospek user.

 Statistik berikut menunjukkan daftar prospek ladangbisnis.localhost</p>";
 $form = "proses_db.php?proses=prospek";
 $order_by = "tgl_masuk"; // urut berdasarkan tgl_masuk
}

// buat array untuk header kolom
if ($tabel == 'tbl_prospek')
 $header_arr = array("Tgl. Join", "Email", "Nama", "reseller", "Hapus?");
else
 $header_arr = array("Tgl. Join", "Username", "Nama", "Email", "Kota",
 "Reseller", "Transfer", $status, "Hapus?");

$query = "SELECT * FROM $tabel";

// panggil class halaman
$hal = new halaman;
$hal->set_tabel($tabel);
$hal->set_page($page);
$pph = $hal->set_pph($u_pph); // ada di konfig.php

// lakukan query untuk mendapatkan banyaknya data dan jumlah halaman
$hal->query_SQL($query);
$jml_data = $hal->get_jml_data();
$jml_hal = $hal->get_jml_hal();

$utama = "

| |
|--|
| \$jdl <form action='\$form' method='post'> <table border='1' cellpadding='5' bordercolor='#cgcgcf' width='775'> <tr bgcolor='#ba7b31' class='putih'> .buat_kolom(\$header_arr, 'th')."</tr>"; // lakukan query ulang \$record = \$hal->get_record(); \$hasil = \$hal->query_SQL(\$query." ORDER BY \$order_by DESC LIMIT \$record, \$pph"); // lakukan looping untuk menampilkan hasil while (\$data = mysql_fetch_array(\$hasil)) { // agar setiap baris warnanya silih berganti if (\$color == '#f4f4f4') \$color = '#ffffff'; else \$color = '#f4f4f4'; \$chk_akt = "<input type='checkbox' name='chk_name' value='".\$data[1].'">"; \$chk_del = "<input type='checkbox' name='hapus[]' value='".\$data[1].'">"; // buat link email \$l_mail = "<a href='mail_to.php?to=\$data[4]' class='tabel'\$data[4]"; // buat array kolom if (\$tabel == 'tbl_prospek') { \$l_mail = "<a href='mail_to.php?to=\$data[1]' class='tabel'\$data[1]"; \$kolom_arr = array(\$data[0], \$l_mail, \$data[2], \$data[3], \$chk_del); } else \$kolom_arr = array(\$data[0], \$data[1], \$data[3], \$l_mail, \$data[5], } |
|--|


```

```

$data[6], rupiah($data[7]), $chk_akt, $chk_del);

$utama .= "
<tr bgcolor='\$color' height='25'>
.buat_kolom($kolom_arr, 'td', 'align="center" class="sf"')." </tr>;
} // akhir while

$hal->set_hal(); // beri nilai untuk navigasi halaman

// cek jika tabelnya prospek maka cukup sediakan satu tombol
if ($tabel == 'tbl_prospek')
$utama .= "
<tr bgcolor='#ba7b31' align='center'><td colspan='4'></td>
<td><input type='submit' value='Hapus'></td></tr>
</table></form>
";
else
$utama .= "
<tr bgcolor='#ba7b31' align='right'><td colspan='8'>
<input type='submit' name='sub_ganda' value='\$tombol'></td>
<td colspan='9'><input type='submit' name='sub_ganda'
value='Hapus'></td></tr>
</table></form>
";

$utama .= $hal->show_page("statistik.php?tabel=$tabel")."$kembali
</td></tr>
</table>";

} // akhir else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $web_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan()
?>

```

Simpan pada direktori **reseller\my\_cp\_** dengan nama **statistik.php**. Output dari file ini dapat anda lihat pada gambar 5.13.

Tugas anda sebagai administrator hanyalah mencocokkan data transfer yang ada di statistik dengan yang ada di bank anda. Jika cocok maka cukup klik tombol **Aktivasi** dengan segera user tersebut sudah teraktivasi.

## PENJELASAN SCRIPT

Karena merupakan file admin pertama kita mengecek apakah admin sudah login atau belum dengan mengecek session admin. Jika belum maka tampilkan form login.

Data pada URL yang kita handel adalah tabel. Dimana isi dari tabel inilah yang menentukan aksi yang akan dijalankan. Untuk tabel pemesan dan blokir tampilannya sama. Yang berbeda hanya form dan isi halaman saja.

Karena kita menghadapi kemungkinan 4 tabel, maka kita harus memilih variabel mana yang sama dan banyak digunakan oleh semua tabel agar kita tidak perlu menuliskannya ulang pada statemen if.

Untuk variabel \$chk\_name, \$tombol dan \$status tampilan untuk tabel pemesan dan terblokir sama jadi kita meletakkannya diluar statemen if. Begitu pula dengan variabel \$order\_by yang menyimpan kolom mana yang akan di-sort pada query SQL nantinya. Dari keempat tabel hanya tabel prospek yang strukturnya berbeda. Maka kita juga meletakkannya diluar if.

The screenshot shows a Firefox browser window with the title "Jadikan Internet Ladang Bisnis Paling Menguntungkan - Mozilla Firefox". The address bar shows the URL "http://localhost/www/Bab5/hari-26-27-28-29/reseller/statistik.php". The page content is titled "Jadikan internet sumber penghasilan anda" and "Daftar Member". It displays a table with member data:

Tgl. Join	Username	Nama	Email	Kota	Reseller	Transfer	Blokir?	Hapus?
26-10-2005	jono	Mr. Jono	jono@localhost.localdomain	Surabaya	joni	Rp. 50.001,-	<input type="checkbox"/>	<input type="checkbox"/>
26-10-2005	joni	Mr. Joni	joni@localhost.localdomain	Surabaya	admin	Rp. 50.000,-	<input type="checkbox"/>	<input type="checkbox"/>

Below the table are navigation links: '<< < 1 2 > >>' and '<< Kembali'. At the bottom left is a "Done" button.

Gambar 5.13: Output statistik.php saat menampilkan daftar member

Langkah berikutnya adalah kita membuat array untuk header kolom. Array ini nantinya kita cetak menjadi header kolom menggunakan fungsi buat\_kolom(). Variabel array ini kita simpan pada \$header\_arr. Karena tabel untuk prospek strukturnya berbeda maka kita juga harus mengubahnya. Oleh karena itu gunakan statemen if untuk menghandel nilai \$header\_arr.

Kemudian kita membuat variabel query yang isinya adalah statement SQL untuk memilih semua data dari tabel yang diinginkan.

Untuk menampilkan semua data dan juga nomor halaman. Seperti proyek sebelumnya kita juga menggunakan class halaman. Pada metode pertama yaitu set\_tabel() kita menghandel nilai dari \$tabel. Metode ke dua yaitu set\_page() kita menggunakan nilai dari \$page. Untuk menentukan berapa data yang ditampilkan kita menggunakan metode set\_pph(). Parameter dari metode ini kita isikan dengan \$a\_pph yang ada pada file konfig.php.

Selanjutnya kita membuat variabel \$utama yang akan menampilkan hasil dari statistik. Pada variabel \$utama kita membuat form yang tujuannya sesuai dengan isi dari variabel \$form. Isi dari \$form sendiri ditentukan oleh nilai dari \$tabel. Setelah itu kita membuat header kolom dengan perintah berikut.

```
buat_kolom($header_arr, 'th')
```

th pada parameter kedua pada fungsi buat\_kolom mengindikasikan bahwa yang ingin kita cetak adalah header kolom bukan kolom biasa.

Setelah itu kita melakukan query ulang untuk mendaftar user yang ada. Query SQL yang kita gunakan adalah sebagai berikut.

```
$hasil = $hal->query_SQL($query . " ORDER BY $order_by DESC LIMIT $record, $pph");
```

Untuk menampilkan semuanya maka kita perlu melakukan looping dari hasil query diatas. Pada badan looping kita membuat beberapa seperti \$chk\_akt untuk checkbox aktivasi dan \$chk\_del untuk checkbox hapus.

Kita juga membuat link email untuk setiap alamat email user dan kita simpan pada variabel \$l\_mail. Hal ini kita lakukan karena mungkin kita hanya ingin mengirim pesan ke satu user saja dan tidak perlu semuanya. Maka kita cukup klik link email user tersebut dan kita akan dibawa ke form email(akan kita buat nanti).

Untuk membuat kolom-kolom maka kita gunakan fungsi buat\_kolom(). Isi dari buat kolom ini yang berbeda hanya untuk prospek dimana hanya terdapat 5 kolom. Sedangkan untuk tabel lainnya terdapat 9 kolom. Karena kita ingin mencetak kolom biasa maka kita mengisikan td pada parameter ke dua. Parameter ketiga kita dengan attribut tambahan misalnya class dan align.

Setelah looping kita menentukan nilai-nilai navigasi halaman dengan metode \$hal->set\_hal(). Setelah itu kita menentukan jumlah tombol dengan mengidentifikasi nilai \$tabel.

Jika \$tabel adalah tbl\_prospek tombol yang kita sediakan hanya satu saja yaitu tombol hapus. Selain itu kita menyediakan dua tombol yaitu (Blokir/Aktivasi) dan Hapus. Kedua tombol ini memiliki nama yang sama yaitu sub\_ganda. Jadi nanti untuk menghandel aksi yang dilakukan oleh user maka kita mengidentifikasi nilainya yaitu nilai dari value.

Langkah yang paling akhir adalah mencetak link navigasi halaman menggunakan \$hal->show\_page(). Parameter kita isi dengan statistik.php?tabel=\$tabel agar link halaman valid.

File berikutnya adalah *proses\_db.php*. File ini bertugas mengolah data yang dikirim dari file statistik.php. Pada PHP DESIGNER 2006 klik menu **File > New > PHP** untuk membuat file PHP abru kemudian ketik script berikut.

#### Script dari file *proses\_db.php*

```
<?php

/*
 * NamaFile: proses_db.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk memproses aktivasi, blokir, penghapusan user
 */

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/admin_var.php');

// cek apakah user sudah login atau belum
if (!cek_session('admin'))
{
 header('Location: index.php');
 exit;
}
else
{
 // dapatkan data dari URL
 $proses = filter_str($_GET['proses']);
 if ($proses == '')
 $proses = 'pemesan';

 // panggil file var_mail.php dan pastikan hanya dipanggil satu kali
 include_once ('../include/var_mail.php');

 // handel setiap proses dengan switch dan case
 switch ($proses)
 {

 }
}
```

```

case 'pemesan':
/*************/
// ambil data yang dipost
$pilihan = $_POST['sub_ganda'];

if ($pilihan == 'Aktivasi')
{
 $aktivasi = $_POST['aktivasi']; // berupa array
 if ($aktivasi == '')
 $utama = "<p class='error'>Anda belum memilih user untuk diaktivasi.
 $kembali</p>";
 else
 {
 // lakukan looping untuk mengaktifkan setiap member yang dicek
 foreach ($aktivasi as $member_baru)
 {
 // panggil fungsi pindah_data untuk memindahkan user
 // dari tabel pemesan ke tabel member
 $hasil = pindah_data('tbl_pemesan', 'tbl_member', 'username', $member_baru);
 if (!$hasil)
 $utama .= "<p class='error'>Gagal memasukkan data ke database.

 Error: ".mysql_error()."

Kontak Admin</p>";
 else
 {
 $pass = SQL_array("SELECT * FROM tbl_member WHERE
username='$member_baru'");
 $pass_baru = balik_md5($pass['password']); // password yg terenkripsi

 // update tabel dengan password yang telah terenkripsi
 mysql_query("UPDATE tbl_member SET password='$pass_baru' WHERE
username='$member_baru'");

 // masukkan user ke tabel profil
 $data = SQL_array("SELECT * FROM tbl_member WHERE
username='$member_baru'");
 mysql_query("INSERT INTO tbl_profil VALUES('$data[username]',
'$data[password]', '$data[nama]', '$data[email]', '$data[kota]',
'$data[reseller]', '', '', '')");

 // masukkan user ke tabel traffic
 mysql_query("INSERT INTO tbl_traffic VALUES('$data[username]', 0)");

 // kirim email kepada user yang baru saja diaktivasi
 $to = $data['email'];
 $subject = "Keanggotaan anda di ladangbisnis.localhost telah Aktif";
 $header = "From: admin@ladangbisnis.localhost";

 // ganti tag {NAMA}, {USERNAME}, {PASSWORD}
 $isi = str_replace('{NAMA}', $data['nama'], $mail_aktivasi);
 $isi = str_replace('{USERNAME}', $member_baru, $isi);
 $isi = str_replace('{PASSWORD}', $pass['password'], $isi);

 // kirim email
 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail ($to, $subject, $isi, $header);
 $utama .= " $member_baru berhasil diaktivasi
";
 }
 } // akhir dari foreach
 $utama .= $kembali;
 } // akhir dari else $aktivasi==''
}

```

```

else
{
 $hapus = $_POST['hapus']; // bertipe array
 if ($hapus == '')
 $utama = "<p class='error'>Anda belum memilih user yang akan dihapus.
 $kembali</p>";
 else
 {
 // gunakan looping untuk menghapus setiap member yang dicek
 foreach ($hapus as $sampah)
 {
 $hasil = mysql_query("DELETE FROM tbl_pemesan WHERE username='$sampah'");
 if (!$hasil)
 $utama .= "<p class='error'>Gagal menghapus $sampah dari database.</p>";
 else
 $utama .= " $sampah berhasil dihapus.
";
 } // akhir foreach
 $utama .= $kembali;
 } // akhir else $hapus == ''
} // akhir $pilihan == 'Aktivasi'
break;
/************/

/************/
case 'member':
/************/

// ambil data yang dipost
$pilihan = $_POST['sub_ganda'];

if ($pilihan == 'Blokir')
{
 $blokir = $_POST['blokir']; // bertipe array
 if ($blokir == '')
 $utama = "<p class='error'>Anda belum memilih user yang akan diblokir.
 $kembali</p>";
 else
 {
 // lakukan looping untuk mengaktifkan setiap member
 foreach ($blokir as $member)
 {
 // panggil fungsi pindah_data untuk memindahkan user
 $hasil = pindah_data('tbl_member', 'tbl_blokir', 'username', $member);
 if (!$hasil)
 $utama = "<p class='error'>Gagal memblokir member $member.

 Ket: ".mysql_error()." $kembali</p>";
 else
 {
 $data = SQL_array("SELECT * FROM tbl_blokir WHERE username='$member'");
 // kirim email
 $to = $data['email'];
 $subject = "Keanggotaan anda di ladangbisnis.localhost Kami BLOKIR!";
 $header = "From: admin@ladangbisnis.localhost";
 // isi email dari $mail_blokir namun tag {NAMA} diganti
 $isi = str_replace('{NAMA}', $data['nama'], $mail_blokir);

 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail($to, $subject, $isi, $header);

 $utama = " $member berhasil diblokir.
";
 } // akhir else !$hasil
 } // akhir foreach
}

```

```

 $utama .= $kembali;
 } // akhir else $blokir == ''
}

else // $pilihan == Hapus
{
 $hapus = $_POST['hapus'];
 if ($hapus == '')
 $utama = "<p class='error'>Anda belum memilih user yang akan dihapus.
 $kembali</p>";
 else
 {
 // looping foreach untuk menghapus setiap member yang dicek
 foreach ($hapus as $sampah)
 {
 // data user yang akan dihapus
 $data = SQL_array("SELECT * FROM tb_member WHERE username='$sampah'");

 // hapus user dari tabel member, profil dan traffic
 $hasil = mysql_query("DELETE FROM tbl_member WHERE username='$sampah';
 DELETE FROM tbl_profil WHERE username='$sampah';
 DELETE FROM tbl_traffic WHERE username='$sampah'");

 if (!$hasil)
 $utama = "<p class='error'>Gagal menghapus member $sampah.

 KET: ".mysql_error()."
";
 else
 {
 // kirim email ke user yang baru saja dihapus
 $to = $data['email'];
 $subject = "Keanggotaan anda di ladangbisnis.localhost DIHAPUS!";
 $header = "From: admin@ladangbisnis.localhost\n"
 ."Reply-To: no-reply@ladangbisnis.localhost";
 // ganti tag {NAMA} dengan nama dari user
 $isi = str_replace('{NAMA}', $data['nama'], $mail_del);

 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail ($to, $subject, $isi, $header);

 $utama = " Username $sampah berhasil dihapus.
";
 }
 } // akhir foreach

 $utama .= $kembali;
 } // akhir $pilihan == 'hapus'
} // akhir else $pilihan == 'blokir'
break;
***** */

***** */
case 'terblokir':
***** */
// ambil data yang dipost
$pilihan = $_POST['sub_ganda'];

if ($pilihan == 'Aktivasi')
{
 $aktivasi = $_POST['aktivasi']; // berupa array
 if ($aktivasi == '')
 $utama = "<p class='error'>Anda belum memilih user untuk diaktifasi
 kembali. $kembali</p>";
 else
 {

```

```

// lakukan looping untuk mengaktifkan setiap member
foreach ($aktivasi as $member_lama)
{
 // panggil fungsi pindah_data untuk memindahkan user
 // dari tabel pemesan ke tabel member
 $hasil = pindah_data('tbl_blokir', 'tbl_member', 'username', $member_lama);
 if (!$hasil)
 $utama .= "<p class='error'>Gagal memasukkan data ke database.

 Error: ".mysql_error()."
";
 else
 {
 $data = SQL_array("SELECT * FROM tbl_member WHERE
username='".$member_lama "'");
 // kirim email kepada user yang baru saja diaktivasi
 $to = $data['email'];
 $subject = "Keanggotaan anda di ladangbisnis.localhost telah "
 . "DIAKTIFKAN kembali";
 $header = "From: admin@ladangbisnis.localhost";
 // ganti tag {NAMA} dengan nama user
 $isi = str_replace('{NAMA}', $data['nama'], $mail_reaktivasi);
 // kirim email
 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail ($to, $subject, $isi, $header);
 $utama .= " $member_lama berhasil diaktivasi.
";
 }
}
// akhir dari foreach

$utama .= $kembali;
} // akhir dari else $aktivasi==''
}
else
{
 $hapus = $_POST['hapus']; // bertipe array
 if ($hapus == '')
 $utama = "<p class='error'>Anda belum memilih user yang akan dihapus.
 $kembali</p>";
 else
 {
 // gunakan looping untuk menghapus setiap member
 foreach ($hapus as $sampah)
 {
 $hasil = mysql_query("DELETE FROM tbl_pemesan WHERE username= '$sampah '");
 if (!$hasil)
 $utama .= "<p class='error'>Gagal menghapus $sampah dari database.</p>";
 else
 $utama .= "$sampah berhasil dihapus.
";
 } // akhir foreach

 $utama .= $kembali;
 } // akhir else $hapus == ''
} // akhir dari else $aktivasi==pilihan
break;
/************/

/************/
case 'prospek':
/************/

$hapus = $_POST['hapus']; // array
if ($hapus == '')
 $utama = "<p class='error'>Anda belum memilih user yang akan dihapus.
 $kembali";
else

```

```

{
 foreach ($hapus as $sampah)
 {
 $hasil = mysql_query("DELETE FROM tbl_prospek WHERE email='\$sampah' ");
 if (!$hasil)
 $utama = "<p class='error'>Gagal menghapus data. \$kembali</p>";
 else
 $utama .= " \$sampah berhasil dihapus.
";
 } // akhir foreach

 $utama .= $kembali;
}
break;

} // akhir switch
} // akhir else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $web_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampil();
?>

```

Simpan pada direktori **reseller\my\_cp\_** dengan nama **proses\_db.php**.

### PENJELASAN SCRIPT

File ini memiliki 4 case dan masing-masing ditujukan untuk tabel yang berbeda yaitu untuk pemesan, memebr, terblokir dan prospek. Jika pada halaman statistik user melihat daftar pemesan lalu ia mengklik tombol maka blok kode pada *case pemesan* akan dieksekusi.

Pada *case pemesan* kita mengambil data yang dipost yaitu nama tombol sub\_ganda. Kita masukkan nilainya pada \$pilihan. Kemudian kita gunakan statemen if untuk menghandel nilainya.

Jika isinya adalah *Aktivasi* berarti user mengklik tombol aktivasi. Namun sebelumnya kita mengecek kondisi apakah admin sudah memilih user yang akan diaktivasi. Jika ia belu mencentang satu user pun tampilkan pesan error.

```
$aktivasi = $_POST['aktivasi']; // berupa array
```

Karena data yang kita handel berupa checkbox yang berarti array. Untuk mengaktifasi setiap user kita perlu menggunakan looping. Looping yang paling cocok untuk kondisi demikian adalah looping *foreach()*.

```
foreach ($aktivasi as $member_baru)
```

Jadi untuk setiap data aktivasi nilainya kita masukkan pada \$member\_baru untuk setiap looping. Karena struktur dari *tbl\_pemesan* dan *tbl\_member* sama, maka kita memanggil fungsi *pindah\_data* untuk memindah data sesuai dengan username yang diaktivasi.

```
$hasil = pindah_data('tbl_pemesan', 'tbl_member', 'username', $member_baru);
```

Jika tidak ada error berarti proses pemindahan data berjalan sukses. Dan user telah berhasil diaktifasi. Kita tinggal mengirimkan email pemberitahuan bahwa keanggotaan user tersebut telah diaktifkan. Password yang kita kirimkan pada isi email adalah password yang belum terenkripsi.

Setelah itu kita update isi dari password agar berisi password yang terenkripsi dengan fungsi `balik_md5()`. Selain itu kita juga memasukkan data user tersebut ke `tbl_profil` dan `tbl_traffic`.

Untuk isi email kita menggunakan isi dari variabel `$mail_aktivasi` sebagai template email yang ada pada file `var_mail.php`. Kita harus mengubah tag `{XXX}` pada isi email dengan data yang sesuai seperti berikut.

```
$isi = str_replace('{NAMA}', $data['nama'], $mail_aktivasi);
$isi = str_replace('{USERNAME}', $member_baru, $isi);
$isi = str_replace('{PASSWORD}', $pass['password'], $isi);
```

Kita mengganti tag-tag tersebut menggunakan fungsi `str_replace()`. Parameter pertama merupakan string yang ingin kita ganti. Parameter kedua adalah string pengganti dan parameter ketiga kumpulan string yang terdapat string yang kita cari.

Isi dari variabel `$isi` yang pertama adalah mengganti tag `{NAMA}` dengan nama user yang diaktivasi. Tag tersebut kita cari dalam `$mail_aktivasi`. Untuk `$isi` yang kedua pada parameter ketiga kita tidak menggunakan lagi `$mail_aktivasi` karena sudah digantikan oleh variabel `$isi` yang tag `{NAMA}`-nya sudah terganti. Demikian juga untuk nilai `$isi` berikutnya.

Jika nilai dari `$pilihan` adalah *Hapus* maka user menekan tombol hapus. Blok kode pada `else` yang akan dieksekusi. Data yang kita ambil adalah *hapus* yang berupa array. Jadi kita menggunakan `foreach` untuk menghapus setiap user yang dicek.

Untuk *case* lainnya prinsip kerjanya hampir sama dengan *case pemesan*.

File berikutnya yang akan kita buat adalah file yang bertugas untuk melakukan pencarian data pada tabel pemesan, member dan terblokir. Prinsip kerja dari file ini hampir sama dengan `statistik.php`. Untuk membuatnya buat file PHP baru pada PHP DESIGNER 2006 kemudian ketik kode berikut.

#### Script dari file *cari.php*

```
<?php

/**
/* NamaFile: pemesan.php
/* Penulis : Rio Astamal
/* Tanggal : 25-10-2006
/* Fungsi : daftar seluruh pemesan e-book
**/

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/admin_var.php');
include ('../include/class_halaman.php');

// cek apakah user sudah login atau belum
if (!cek_session('admin'))
{
 header('Location: index.php');
 exit;
}
else
{
 // dapatkan data dari URL
 $proses = filter_str($_GET['proses']);
 if ($proses == '')
 $proses = 'utama';

 $page = filter_str($_GET['page']);
```

```

if ($page == '')
$page = 0;

switch ($proses)
{
 /*****
 case 'utama':
 *****/

$utama = "

| |
|---|
| <p class="besar">Pencarian Data</p> <center> <form action='cari.php' method='get'> <table border='1' cellpadding='5' bordercolor='#cgcgcg' bgcolor='#ffffff'> <tr bgcolor='#ba7b31' align='center'> <td colspan='2' class='putih'>FORM PENCARIAN</td></tr> <tr bgcolor='#f4f4f4'><td>Pilih Database</td> <td><select name='tabel'> <option value='tbl_pemesan'>Pemesan</option> <option value='tbl_member'>Member</option> <option value='tbl_blokir'>Terblokir</option> </select></td></tr> <tr><td>Pilih Kolom</td> <td><select name='kolom'> <option value='tgl'>Tanggal Join</option> <option value='username'>Username</option> <option value='nama'>Nama</option> <option value='email'>Email</option> <option value='kota'>Kota</option> <option value='reseller'>Reseller</option> <option value='transfer'>Transfer</option> </select></td></tr> <tr bgcolor='#f4f4f4'><td>Kata kunci</td> <td><input type='text' name='keyword'></td></tr> <tr><td></td><td><input type='submit' value='Cari'></td></tr> </table> <input type='hidden' name='proses' value='cari_data'> </form></center> <p class='sf'>* Khusus untuk kolom transfer cukup ketikkan angkanya saja. Misal anda ingin mencari Rp. 50.030,-. Maka cukup ketikkan 50030 saja.</p> \$kembali </td></tr> </table>"; break; ****/ case 'cari_data': // kosongkan hari dan statistik agar tabel lebih lebar \$hari_ini = ''; \$web_stat = ''; // ambil data yang dipost \$tabel = filter_str(\$_GET['tabel']); \$kolom = filter_str(\$_GET['kolom']); // spasi, titik, -, @ diperbolehkan \$keyword = filter_str(\$_GET['keyword'], " @\.-"); // sesuaikan form dengan tabel yang dicari </center> |
|---|


```

```

// untuk tabel pemesan dan terblokir (tampilan sama)
$chk_name = "aktivasi[]";
$tombol = "Aktivasi";
$status = "Aktivasi?";

// cek nama tabel
if ($tabel == 'tbl_member')
{
 $chk_name = "blokir[]";
 $tombol = "Blokir";
 $status = "Blokir?";
 $form = "proses_db.php?proses=member";
}
else if ($tabel == 'tbl_pemesan')
 $form = "proses_db.php?proses=pemesan";
else if ($tabel == 'tbl_blokir')
 $form = "proses_db.php?proses=terblokir";

// buat array untuk header kolom (pemesan dan terblokir)
$header_arr = array("Tgl. Join", "Username", "Nama", "Email", "Kota",
 "Reseller", "Transfer", $status, "Hapus?");

$query = "SELECT * FROM $tabel WHERE $kolom LIKE '%$keyword%'";

$utama = "
<table border='0' width='100%' cellpadding='5'>
<tr><td>
 <p class='besar'>Hasil Pencarian</p>

// panggil class halaman
$hal = new halaman;
$hal->set_tabel($tabel);
$hal->set_page($page);
$pph = $hal->set_pph($u_pph);

// lakukan query untuk mendapatkan banyaknya data dan jumlah halaman
$hal->query_SQL($query);
$jml_data = $hal->get_jml_data();
$jml_hal = $hal->get_jml_hal();

// cek jumlah baris yang dikembalikan
if ($jml_data == 0)
 $utama .= "<p class='error'>Maaf, data yang anda cari tidak ditemukan.
 $kembali</p>";
else
{
 $utama .= "
 <p>Total data yang ditemukan: $jml_data.

Berikut daftar data
 yang berhasil ditemukan.</p>
 <form action='$form' method='post'>
 <table border='1' cellpadding='4' width='785' bordercolor='#cgcgcf'>
 <tr bgcolor='#ba7b31' class='putih' height='30'>
 .buat_kolom($header_arr, 'th', "class='putih'")."</tr>

 // lakukan query ulang
 $record = $hal->get_record();
 $hasil = $hal->query_SQL($query." ORDER BY transfer DESC LIMIT $record,
 $pph");

 // lakukan looping untuk menampilkan hasil
 while ($data = mysql_fetch_array($hasil))
 {
 if ($color == '#f4f4f4')
 $color = '#ffffff';

```

```

else
 $color = '#f4f4f4';

$chk_akt = "<input type='checkbox' name='chk_name' value='".$data[1].'"";
$chk_del = "<input type='checkbox' name='hapus[]' value='".$data[1].'"";

// buat link email
$l_mail = "<a href='mail_to.php?to=$data[4]' class='tabel'$data[4]";

// buat array untuk kolom
$kolom_arr = array($data[0], $data[1], $data[3], $l_mail, $data[5],
 $data[6], rupiah($data[7]), $chk_akt, $chk_del);
$utama .= "
<tr bgcolor='".$color' height='25'>
 .buat_kolom($kolom_arr, 'td', 'align="center" class="sf"')."</tr>";
} // akhir while

$hal->set_hal();
$utama .= "
<tr bgcolor="#ba7b31' align='right'><td colspan='8'>
 <input type='submit' name='sub_ganda' value='".$tombol'></td>
 <td colspan='9'><input type='submit' name='sub_ganda'
 value='Hapus'></td></tr>
</table></form>

.$hal->show_page("cari.php?proses=cari_data&tabel=$tabel&kolom=
 $kolom=&keyword")."$kembali
</td></tr>
</table>";

} // akhir else $jml_data == 0

break;
*****/
} // akhir switch
} // akhir else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $web_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **reseller\my\_cp\_** dengan nama **cari.php**. Output dari file ini ditunjukkan oleh gambar 5.14.

### PENJELASAN SCRIPT

Agar pada hasil pencarian kita dapat berpindah-pindah antar halaman. Maka untuk attribut method pada form kita isi *GET* bukan *POST*. Jika kita menggunakan *POST* maka kita tidak bisa mengambil nama tabel yang digunakan dan keyword.

Hal itu dikarenakan data kan hilang jika kita berpindah pada halaman lainnya. Namun jika menggunakan *GET* kita tetap dapat menyimpannya di URL.



Gambar 5.14: Output dari file cari.php

Yang membedakan file ini dengan *statistik.php* adalah pada varibel \$query yang kita gunakan. Karena pencarian maka kita menggunakan query berikut.

```
$query = "SELECT * FROM $tabel WHERE $kolom LIKE '%$keyword%' ;"
```

File berikutnya yang akan kita buat adalah file yang bertugas untuk mengirimkan email ke seluruh user. Kita dapat mengirim email ke satu orang user atau seluruh user sesuai dengan pilihan. Buat file PHP baru kemudian ketik script berikut.

#### Script dari file *mail\_to.php*

```
<?php

/*
 * NamaFile: mail_to.php
 * Penulis : Rio Astamal
 * Tanggal : 26-10-2006
 * Fungsi : untuk mengirim email ke seluruh/sebagian user
 */

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/admin_var.php');

// cek apakah user sudah login atau belum
if (!cek_session('admin'))
{
 header('Location: index.php');
 exit;
}
else
```

```

// dapatkan data dari URL
$proses = filter_str($_GET['proses']);
if ($proses == '')
 $proses = 'utama';

switch ($proses)
{
 /*****
 case 'utama':
 ****/

 // dapatkan data dari URL
 $to = filter_str($_GET['to'], " @\.-"); // spasi, titik, - , @ diperbolehkan
 if ($to == '')
 $readonly = "readonly";

 $utama = "
<table border='0' width='100%' cellpadding='5'>
<tr><td>
 <p class='besar'>Kirim Email</p>
 <p>Pada halaman ini anda dapat mengirim email ke semua user. Baik member, pemesan, prospek dan terblokir. Juga pada user tertentu saja.</p>
 <form action='mail_to.php?proses=kirim' method='post'>
 <table border='1' cellpadding='5' bordercolor='#cgcgcf'>
 <tr bgcolor='#ba7b31' align='center'>
 <td colspan='2' class='putih'>FORM KIRIM EMAIL</td></tr>
 <tr bgcolor='#f4f4f4'><td>Kirim ke</td>
 <td><input type='checkbox' name='kepada[]' value='tbl_member'>
 Member
 <input type='checkbox' name='kepada[]' value='tbl_pemesan'>
 Pemesan
 <input type='checkbox' name='kepada[]' value='tbl_prospek'>
 Prospek
 <input type='checkbox' name='kepada[]' value='tbl_blokir'>
 Terblokir </td></tr>
 <tr bgcolor='#ffffff'><td>Spesifik (To)</td>
 <td><input type='text' name='to' value='$to' $readonly></td></tr>
 <tr bgcolor='#f4f4f4'><td>Subject</td>
 <td><input type='text' name='subject' size='60'></td></tr>
 <tr bgcolor='#ffffff'><td>Isi Pesan</td>
 <td><textarea name='isi' rows='10' cols='55'></td></tr>
 <tr bgcolor='#f4f4f4'><td><input type='submit' value='Kirim'></td></tr>
 </table></form>
 </td></tr>
</table>";
break;
 ****/
}

*****/
case 'kirim':
*****/

// ambil data yang dipost
$to = $_POST['to'];
$kepada = $_POST['kepada']; // array
$subject = $_POST['subject'];
$isi = $_POST['isi'];
// untuk menghindari mailer daemon, alamat reply kita isi sembarang
$header = "From: admin@ladangbisnis.localhost\n"
 ."Reply-To: no-reply@ladangbisnis.localhost";

```

```

// jika $to tidak kosong maka pengiriman secara spesifik(1 user)
if ($to != '')
{
 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail ($to, $subject, $isi, $header);
 $utama = "<p align='center'>Email berhasil dikirim ke $to.</p>";
}
else // email ke banyak user
{
 // gunakan looping untuk mengirim email ke seluruh user pada tabel
 foreach ($kepada as $tabel)
 {
 // lakukan query SQL
 $hasil = mysql_query("SELECT * FROM $tabel");
 $total += mysql_num_rows($hasil);
 // gunakan looping untuk mengirim email ke semua alamat
 while ($data = mysql_fetch_array($hasil))
 {
 $ke = $data['email'];
 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail ($ke, $subject, $isi, $header);
 }
 } // akhir foreach

 $utama = "<p align='center'>Email berhasil dikirim ke $total alamat
 email user.</p>";
} // akhir else $to != ''

break;
***** */
} // akhir switch

} // akhir else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $web_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();
?>

```

Simpan pada direktori **reseller\my\_cp\_** dengan nama **mail\_to.php**.

### PENJELASAN SCRIPT

Jika kita membuka halaman ini lewat klik yang kita lakukan pada alamat email salah satu user yang ada pada daftar di statistik. Maka field Spesifik secara otomatis terisi email user tersebut. Namun bila kita membuka halaman ini lewat link KIRIM EMAIL, maka field spesifik secara otomatis tidak dapat diisi atau bersifat readonly.

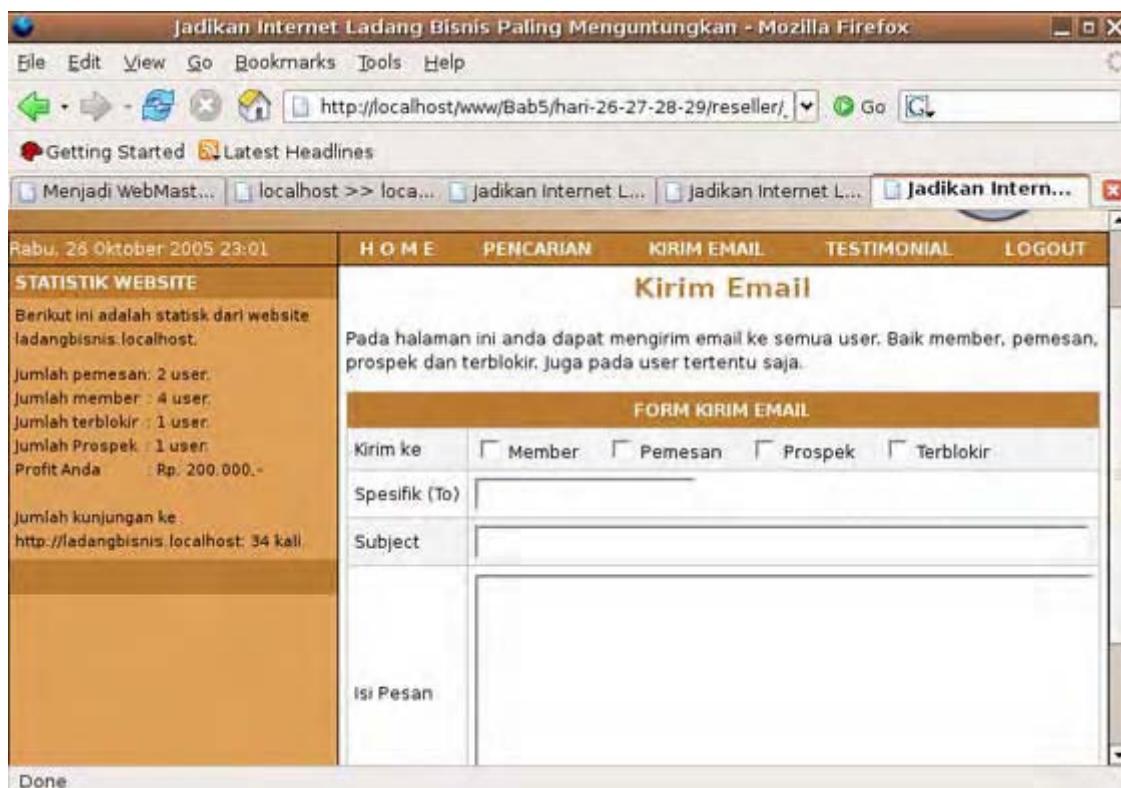
Untuk dapat mengirim email ke seluruh member pada daftar yang kita cek. Maka kita gunakan looping foreach().

```
foreach ($kepada as $tabel)
```

Pada blok looping foreach kita melakukan query berikut

```
$hasil = mysql_query("SELECT * FROM $tabel");
```

Jadi jika kita mengecek dua tabel misalnya member dan pemesan. Maka pada looping pertama email akan dikirim ke seluruh member dan pada looping kedua email akan dikirim ke seluruh pemesan.



Gambar 5.15: Output file mail\_to.php

File terakhir yang akan kita buat pada direktori `_my_cp_` ini adalah file yang kita gunakan untuk menambah, mengedit, dan menghapus testimonial. Disini pemasukan testimonial ke tabel `tbl_testimonial` kita lakukan secara manual. Dimana kita mengambilnya sesuai apa yang dikirimkan user lewat email.

Jika anda tidak suka cara demikian anda dapat mengganti kodennya agar testimonial yang dikirimkan member langsung tersimpan didatabase.

Tapi itu nanti saja, untuk sementara ikuti saya saja! OK. Buat file PHP dengan mengklik menu **File > New > PHP** pada PHP DESIGNER 2006 kemudian ketik kode berikut.

#### Script dari file **testimonial.php**

```
<?php

/*************************************
/* NamaFile: testimonial.php
/* Penulis : Rio Astamal
/* Tanggal : 26-10-2006
/* Fungsi : untuk menambah/menghapus/mengedit testimonial
/************************************/

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/admin_var.php');
include ('../include/class_halaman.php');

// cek apakah user sudah login atau belum
if (!cek_session('admin'))
```

```

{
 header('Location: index.php');
 exit;
}
else
{
 // dapatkan data dari URL
 $proses = filter_str($_GET['proses']);
 if ($proses == '')
 $proses = 'utama';

 $page = filter_str($_GET['page']);
 if ($page == '')
 $page = 0;

 switch ($proses)
 {

 //*****
 case 'utama':
 //*****

 // kosongkan statistik dan hari
 $hari_ini = ''; $web_stat = '';

 // panggil class halaman
 $hal = new halaman;
 $hal->set_tabel('tbl_testimonial');
 $hal->set_page($page);
 $pph = $hal->set_pph($a_tmph); // ada di konfig.php

 $hal->query_SQL(1); // lakukan query
 $jml_testim = $hal->get_jml_data();
 $jml_hal = $hal->get_jml_hal();

 // buat header kolom
 $header_arr = array("User", "Testimonial", "Edit?", "Hapus?");

 $utama = "
<table border='0' width='100%' cellpadding='5'>
<tr><td>
 <p class='besar'>Testimonial Member</p>
 <p>Total Terdapat: $jml_testim testimonial.

 Untuk mengedit silahkan lakukan perubahan langsung pada testimonial
 lalu klik Edit.</p>
 <table border='1' cellpadding='4' width='100%' bordercolor='#cgcgcg'>
 <tr bgcolor='#ba7b31' height='30'>
 .buat_kolom($header_arr, 'th', 'align="center" class="putih"')."</tr>";

 // lakukan query ulang
 $record = $hal->get_record();
 $hasil = $hal->query_SQL("SELECT * FROM tbl_testimonial ORDER BY tm_id
 DESC LIMIT $record, $pph");

 // looping untuk menampilkan testimonial
 while ($data = mysql_fetch_array($hasil))
 {
 if ($color == '#f4f4f4')
 $color = '#ffffff';
 else
 $color = '#f4f4f4';

 $utama .= "
 <tr bgcolor='$color'>

```

```

<td>$data[username]</td>
<td><form action='testimonial.php?proses=edit' method='post'>
 <textarea name='isi' rows='10'
 cols='80'>$data[testimonial]</textarea></td>
<td><input type='hidden' name='id' value='".$data[tm_id]">
 <input type='submit' value='Edit'></form></td>
<td><form action='testimonial.php?proses=hapus' method='post'>
 <input type='hidden' name='id' value='".$data[tm_id]">
 <input type='submit' value='Hapus'></form></td></tr>" ;
}

$hal->set_hal(); // nilai link halaman
$utama .= "
</table>
"
.$hal->show_page('testimonial.php?proses=utama')."

Tambah Testimonial
</td></tr>
</table>";

break;
***** */

***** */
case 'edit':
***** */

// ambil data yang dipost
$id = $_POST['id'];
$isi = addslashes($_POST['isi']);

// update database
$hasil = mysql_query("UPDATE tbl_testimonial SET testimonial='".$isi' WHERE
 tm_id=$id");
if (!$hasil)
 $utama = "<p class='error'>Gagal mengupdate testimonial.

 KET: ".mysql_error()."</p>";
else
 $utama = "<p align='center'>Testimoni berhasil diupdate.$kembali</p>";

break;
***** */

***** */
case 'hapus':
***** */
$id = $_POST['id'];

// lakukan query DELETE
$hasil = mysql_query("DELETE FROM tbl_testimonial WHERE tm_id=$id");
if (!$hasil)
 $utama = "<p class='error'>Gagal menghapus testimonial.

 KET: ".mysql_error()."</p>";
else
 $utama = "<p align='center'>Testimonial berhasil dihapus. $kembali</p>";

break;
***** */

***** */
case 'tambah':
***** */

$utama =
<table border='0' cellpadding='5' width='100%'>

```

```

<tr><td>
 <p class='besar'>Tambah Testimonial</p>
 <form action='testimonial.php?proses=pros_tambah' method='post'>
 <table border='1' cellpadding='5' width='100%' bordercolor='#cgcgcf'>
 <tr bgcolor='#ba7b31' align='center'><td colspan='2' class='putih'>
 FORM TAMBAH TESTIMONIAL</td></tr>
 <tr bgcolor='#f4f4f4'><td>User</td>
 <td><input type='text' name='user'></td></tr>
 <tr bgcolor='#ffffff'><td>Isi</td>
 <td><textarea name='isi' rows='10' cols='65'></td></tr>
 <tr bgcolor='#f4f4f4'><td></td>
 <td><input type='submit' value='Tambah'></td></tr>
 </table>
 </form>
 </td></tr>
</table>";

break;
/*************/
/* ****
case 'pros_tambah':
**** */

$user = filter_str($_POST['user']);
$isi = addslashes($_POST['isi']);

if (empty($user) || empty($isi))
 $utama = "<p class='error'>MASIH ADA FIELD YANG KOSONG. $kembali</p>";
else
{
 // masukkan ke database
 $hasil = mysql_query("INSERT INTO tbl_testimonial VALUES(0, '$user',
'$isi')");
 if (!$hasil)
 $utama = "<p class='error'>GAGAL MEMASUKKAN DATA KE DATABASE.

 KET: ".mysql_error()." $kembali</p>";
 else
 $utama = "<p align='center'>TESTIMONIAL BERHASIL DITAMBAHKAN. $kembali</p>";
}

break;
/************/
} // akhir switch
} // akhir else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $web_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();
?>

```

Simpan pada direktori **reseller\my\_cp\_** dengan nama **testimonial.php**. Output dari file ini ditunjukkan oleh gambar 5.16. Coba tambahkan beberapa testimonial. Isikan sembarang saja karena hanya untuk percobaan.

Tidak ada yang istimewa dari file ini. Kita menampilkan daftar testimonial dengan menggunakan class halaman, membuat header kolom menggunakan **buat\_kolom()** dan fungsi

standar lainnya.

Yang perlu anda perhatikan hanyalah format isi testimonial yang akan anda tulis. Misalnya letak Nama pengirim testimonial, emailnya, alamat dan sebagainya. Jangan lupa untuk menyamarkan alamat email dan alamat rumah user agar ia tidak menjadi korban SPAM dari pihak yang tidak bertanggung jawab.

User	Testimonial	Edit?	Hapus?
hacky	Baru kali ini saya menemukan suatu yang berharga diinternet!. Berkat e-book anda saat ini saya memiliki bisnis yang sangat menguntungkan!. SALAM SUKSES SELALU!!  Hacky Under # sp...@localhost.localdomain # Jl. Cyber Optic ... # Surabaya	<a href="#">Edit</a>	<a href="#">Hapus</a>

Gambar 5.16: Output file testimonial.php

Akhir dari file yang berada pada direktori **reseller\my\_cp\**.

Berikutnya kita akan membuat file-file yang ada pada direktori paling atas yaitu *reseller*. File-file ini bertugas untuk menampilkan sales letter, melayani pemesan dan masih banyak lagi. Dan untuk yang pertama kita akan membuat file yang sangat penting karena menentukan kesuksesan kita. File itu adalah *index.php* yang akan menampilkan headline website dan sales letter.

Lansung saja untuk membuatnya klik menu **File > New > PHP** pada PHP DESIGNER 2006 kemudian ketik script berikut ini.

Script dari file **index.php**

```
<?php
/*
 * NamaFile: index.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk menampilkan halaman utama dan sales letter
*/
// panggil file-file yang diperlukan
include ('include/class_skin.php');
include ('include/var_utama.php');

$utama = "
```

```

<table border='0' cellpadding='6' width='100%'>
<tr><td>

<!-- HeadLine / Judul Website (bagian terpenting) -->

<p class='besar'>Bagaimana Menyiasati Internet Agar Menghasilkan Profit
Yang Melimpah!</p>
<p></p>
<p class='besar'>Temukan rahasianya hanya di ladangbisnis.localhost</p>
<p></p>

<!-- Bagian sales letter (Pembukaan) -->

<p>Saudara netter, tahukah anda bahwa sesuatu yang kita sebut dengan
\\"INTERNET\\"
ternyata menyimpan segudang misteri. Jangan takut misteri ini bukan berhubungan
dengan hal-hal mistis. Melainkan sesuatu yang pasti anda kejar seumur hidup,
apalagi kalau bukan \\"UANG\\".</p>
<p>Dengan internet kita dapat menghasilkan penghasilan yang melimpah,
<u>hanya</u>
jika kita tahu CARANYA. Dan untuk itulah e-book \\"Internet Sebagai Ladang
Bisnis\\"
hadir. Untuk memantu anda dalam memecahkan masalah finansial.</p>
<p>Dengan e-book ini anda akan diajarkan bagaimana teknik-teknik dalam berbisnis
di internet DAN SETERUSNYA... DAN SETERUSNYA... DAN SETERUSNYA...</p>

<!-- Tampilkan Daftar Manfaat Produk Anda -->

<p class='besar'>Manfaat Apa Yang Saya Peroleh Dari E-book Ini</p>
<p>Setelah anda membaca e-book ini anda akan tahu bagaimana menghasilkan
profit yang melimpah dari internet. DAN SETERUSNYA... DAN SETERUSNYA...</p>

<!-- Gunakan bullet list untuk daftar manfaat -->
<p>E-book ini akan mengajarkan anda langkah demi langkah agar anda menjadi
seorang yang sukses.
<p>Manfaat 1, Tulis manfaat 1 disini.</p>
<p>Manfaat 2, Tulis manfaat 2 disini.</p>
<p>Manfaat 3, Tulis manfaat 3 disini.</p>
<p>DAN SETERUSNYA, DAN SETERUSNYA...</p>
</p>
<p>Itulah daftar manfaat yang akan anda peroleh setelah memiliki e-book
\\"Internet Sebagai Ladang Bisnis\\". DAN SETERUSNYA...</p>

<!-- CANTUMKAN HARGA PRODUK, BUAT SEOLAH-OLAH PRODUK ANDA BEGITU BERHARGA
NAMUN DENGAN HARGA YANG RENDAH -->

<p class='besar'>Berapa Harga E-book Ini?</p>
<p>\\"E-book ini harganya pasti sangat mahal!\\". Mungkin itulah ungkapan yang
terbesit dari benak anda jika sudah memikirkan harga e-book ini. Namun anda
salah jika mengira e-book ini sangat mahal. Dengan segala manfaat yang ada
e-book ini saya bandrol dengan harga Rp. 100.000,-.</p>
<p>Seharusnya harga Rp. 100.000,- untuk e-book ini tidak sesuai mengingat
manfaat yang akan anda peroleh dari e-book ini. DAN SETERUSNYA... DAN
SETERUNYA...</p>

<!-- sediakan link untuk memesan -->
<p align='center'>
Klik Disini Untuk Memesan</p>

<!-- Ciptakan Kondisi Keterburu-buruan Menggunakan DEADLINE -->

<p>Oops...! hampir terlupa,</p>
<p class='besar'>Jika Memesan Sebelum $hari_deadline
Maka Dapatkan Juga X Super Bonus!</p>

```

<p>Untuk keterangan lengkap tentang ke-x super bonus tersebut. Silahkan <a href='bonus.php?res=\$res'>klik disini.</a></p>

<p>Dalam bonus-bonus tersebut akan anda temukan berbagai informasi menarik tentang bisnis di internet. Juga berbagai kiat-kiat sukses para webmaster dalam menjalankan bisnisnya. Harga dari bonus-bonus tersebut bahkan melebihi harga produk utama.</p>

<!-- Beritahukan bahwa bonus akan hilang jika memesan melewati deadline -->

<p><b><font color='#ba7b31'>Catatan: Ke-x bonus tersebut akan hangus jika anda memesan melewati<br> &ampnbsp &nbs&nbsp; hari \$hari\_deadline.</font></b></p>

<!-- Tampilkan Garansi Produk Anda -->

<p class='besar'>Hilangkan Keraguan Anda dengan Garansi 100% Uang Kembali</p>

<p>Jika anda merasa bahwa produk ini sama sekali tidak membawa perkembangan bagi anda. Atau anda tidak puas dengan produk ini. Kami akan mengembalikan uang anda 100% tanpa pengurangan sedikitpun. Waktu garansi berlaku selama <b>12 bulan</b> terhitung sejak pembelian produk.</p>

<p>Jadi apa lagi yang perlu anda khawatirkan. Segera lakukan pemesan hari ini juga. <a href='pesan.php?res=\$res'><b>Klik disini untuk memesan.</b></a></p>

<!-- Beritahukan tentang Program Reseller Anda -->

<p class='besar'>Dapatkan Komisi 50% dari Program Reseller</p>

<p>Setelah anda melakukan pemesanan dan mentransfer dana. Kami akan mengaktifkan keanggotaan anda. Maka secara otomatis anda telah menjadi reseller ladangbisnis.localhost. </p>

<p>Anda akan kami beri URL khusus untuk mempromosikan situs ini. Dan jika ada pembeli yang masuk lewat URL anda, maka anda akan mendapatkan komisi sebesar 50%. Untuk keterangan lengkap program reseller <b><a href='reseller.php?res=\$res'>Klik disini.</a></b></p>

<!-- Lebih yakinkan pengunjung dengan daftar bonus anda -->

<p>Bayangkan hanya dengan investasi satu kali Rp. 100.000,- inilah yang akan anda dapatkan dari ladangbisnis.localhost:

<ol>

- <p><li>E-book yang revolusioner \"Internet Sebagai Ladang Bisnis\"</li></p>
- <p><li>Berbagai super bonus:</li>
  - <p><li>Super Bonus 1</li></p>
  - <p><li>Super Bonus 2</li></p>
  - <p><li>Super Bonus 3</li></p>
  - <p><li>DAN SETERUSNYA...</li></p>

</ul></p>

<p><li>Garansi <b>12 bulan</b> 100% uang kembali</li></p>

<p><li>Komisi 50% atau Rp. 50.000,- pada program reseller</li></p>

</ol></p>

<p>Hanya dengan Rp. 100.000,- semua poin-poin diatas dapat anda miliki. Jadi tunggu apa lagi. Segera lakukan pemesanan hari ini juga!!.</p>

<p align='center'><b><font size='3'><a href='pesan.php?res=\$res'>Klik Disini Untuk Melakukan Pemesanan</a></b></font></p>

</tr></td>

</table>";

```
// update nilai traffic user jika res kosong gunakan username admin
if ($_GET['res'] != '')
 update(1, $_GET['res']);
else
 update(1, 'admin');
```

```

$skin = new skin;
$skin->ganti_skin('template/index_skin.php');
$skin->ganti_tag('{TODAY}', "Hari ini: $hari_ini");
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KANAN}', $bonus.$me);
$skin->ganti_tag('{UTAMA}', $reseller.$utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampil();
?>

```

---

Simpan pada direktori **reseller** dengan nama ***index.php***. Output dari file ini ditunjukkan oleh gambar 5.8 pada awal pembuatan materi ini.

### PENJELASAN SCRIPT

Pada file ini kita hanya menampilkan headline dan mencoba membuat user tertarik untuk membeli dengan menampilkan sales letter.

Teori mengenai headline dan sales letter sudah pernah kita pelajari pada bahasan sebelumnya. Sebagai pengingat saja. Pada sales letter diatas kita membaginya dalam beberapa bagian. Pada contoh headline diatas kita menggunakan dua kalimat satu bersifat "Bagaimana ..." dan satu lagi bersifat "Berita".

Pada sales letter kita juga membaginya dalam beberapa bagian. Setiap bagian kita beri *sub-judul* agar alur yang kita berikan pada pengunjung lebih "*mengalir*". Bagian-bagian pada sales letter diatas diantaranya:

- Pembukaan
- Daftar Manfaat
- Harga
- Ciptakan Kondisi
- Bonus
- Garansi
- Program Reseller
- Daftar Bonus

Dalam variabel \$utama terdapat beberapa variabel seperti \$hari\_biasa, \$hari\_deadline, \$res. Perlu anda ingat variabel-variabel tersebut sudah pernah kita definisikan pada file *var\_utama.php*.

Pada bagian akhir kita mengupdate nilai dari traffic pada tabel *tbl\_traffic* menggunakan fungsi *update()*. Jika pada URL res kosong maka traffic dari admin yang akan ditambah.

Berikutnya file yang akan kita buat adalah file yang akan menangani proses pemesanan. File ini juga yang berfungsi membagi transfer antara admin dan resellernya. Pada proses pemesanan pembeli dapat mengetahui kemana dan berapa nilai transfer yang harus kirimkan. Buat file PHP baru kemudian ketik script berikut.

### Script dari file ***pesan.php***

---

```

<?php
/*
 * NamaFile: pesan.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk memproses pemesanan dari user
*/
session_start();

```

```

// panggil file-file yang diperlukan
include ('include/class_skin.php');
include ('include/var_utama.php');

// dapatkan data dari URL
$proses = filter_str($_GET['proses']);
if ($proses == '')
 $proses = 'utama';

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'utama':
 ****/

 // untuk persetujuan antara user dan pemilik program (EULA)
 $utama = "
<table border='0' cellpadding='6' width='100%'>
<tr><td>

Bab 5 - Memulai Bisnis di Internet, Web Design, dan Membuat Website Reseller


```

```


Pengaktifkan keanggotaan anda maksimal 24 jam setelah anda melakukan melakukn transfer.
</td></tr>
</table>

<center>
<p>Isi Form Berikut Lalu tekan Proses untuk melanjutkan</p>
<form action='pesan.php?proses=pros_form&res=$res' method='post'>
<table border='1' cellpadding='0' width='70%'>
<tr height='35' align='center'>
 <td bgcolor='#ba7b31' colspan='2'>
 FORM REGISTRASI</td></tr>
<tr><td>
<table border='0' cellpadding='5' width='100%'>
<tr bgcolor='#f4f4f4'>
 <td>Username</td>
 <td><input type='text' name='username' maxlength='16'></td></tr>
<tr bgcolor='#ffffff'>
 <td>Password</td>
 <td class='sf'><input type='password' name='password1' maxlength='16'>
 6 - 16 karakter.</td></tr>
<tr bgcolor='#f4f4f4'>
 <td>Ulangi</td>
 <td><input type='password' name='password2' maxlength='16'></td></tr>
<tr bgcolor='#ffffff'>
 <td>Nama Lengkap</td>
 <td><input type='text' name='nama' size='30'></td></tr>
<tr bgcolor='#f4f4f4'>
 <td>Kota</td>
 <td><input type='text' name='kota'></td></tr>
<tr bgcolor='#ffffff'>
 <td>Email</td>
 <td><input type='text' name='email' size='30'></td></tr>
<tr bgcolor='#f4f4f4'>
 <td></td><td><input type='submit' value='Proses'></td></tr>
</table>
</td></tr>
</table>
</form></center>

</td></tr>
</table>";

break;
/************/
/* **** */
case 'pros_form':
/************/

// ambil data yang dipost


```

```

$username = $_POST['username'];
$password1 = $_POST['password1'];
$password2 = $_POST['password2'];
$nama = filter_str($_POST['nama'], " \.,"); // spasi, titik, koma diperbolehkan
$kota = filter_str($_POST['kota'], " \.,");
$email = $_POST['email'];

$pesan_err = '';

// cek setiap field
if (!cek_field($_POST))
 $pesan_err = "Error: Masih ada field yang kosong.
";

// cek karakter dari username
if (ereg('^[a-zA-Z0-9_]', $username))
 $pesan_err .= "Error: Username hanya boleh terdiri dari karakter alpanumerik
 dan _.
";

// cek kecocokan password
if ($password1 != $password2)
 $pesan_err .= "Error: Password tidak sama.
";

// cek panjang password
if (strlen($password1) < 6)
 $pesan_err .= "Error: Password harus diantara 6 - 16 karakter.
";

// cek karakter password
if (ereg('^[a-zA-Z0-9_]', $password1))
 $pesan_err .= "Error: Password hanya boleh terdiri dari karakter alpanumerik
 dan _.
";

// cek email
if (!cek_email($email))
 $pesan_err .= "Error: Email tidak valid.
";

// cek username sudah terpakai atau belum
$q1 = mysql_query("SELECT * FROM tbl_pemesan WHERE username='$username'");
$q2 = mysql_query("SELECT * FROM tbl_member WHERE username='$username'");
if (mysql_num_rows($q1) > 0 || mysql_num_rows($q2) > 0)
 $pesan_err .= "Error: Username sudah terpakai, pilih yang lain.
";

if ($pesan_err != '')
 $utama = "$pesan_err$kembali";
else
{

// dapatkan nilai transfer
$data = SQL_array("SELECT * FROM tbl_transfer");
$transfer = $data[0];

// buat session untuk menyimpan data yang akan diproses
$_SESSION['reg_username'] = $username;
$_SESSION['reg_password'] = $password1;
$_SESSION['reg_nama'] = $nama;
$_SESSION['reg_kota'] = $kota;
$_SESSION['reg_email'] = $email;
$_SESSION['reg_transfer'] = $transfer;

// dapatkan profil dari reseller
$data_res = SQL_array("SELECT * FROM tbl_profil WHERE username='$res'");

// tampilkan nama dan no. rekening untuk transfer
$utama =
<table border='0' cellpadding='5' width='100%'>

```

```

<tr><td>

<p class='besar'>Segera Lakukan Pembayaran Hari Ini Juga dan Dapatkan
Berbagai keuntungan Diantaranya:</p>
<p>
 X Super Bonus dari Saya.
 Program Reseller, Anda dapat meraih komisi 50% dari setiap
 pembelian.
 Dan masih banyak lagi bonus-bonus lain yang akan saya berikan
 pada member area.
</p>

<table border='1' cellpadding='5'>
<tr><td bgcolor='#f4f4f4'>
 <p align='center'>Berikut ini adalah data REKENING
 untuk transfer.
Catat Dengan BENAR!

 Jumlah dana yang harus anda transfer adalah sebesar
 ".rupiah($transfer)." x2 masing-masing ke:</p>
 <p>ADMINISTRATOR: ".rupiah($transfer)."

 Rio Astamal - admin@ladangbisnis.localhost

 BCA WIYUNG Surabaya - Rek. 0809899999 - a.n Rio Astamal

 RESELLER: ".rupiah($transfer)."

 $data_res[nama] - $data_res[email]

 $data_res[bank]

 * Konfirmasi pendaftaran ini juga kami kirim ke alamat email anda.

 * Keanggotaan anda akan kami aktifkan maksimal 24 jam setelah anda
 transfer.
</td></tr>
</table>";

// panggil kelas deadline untuk membuat deadline baru(7 hari)
$dead = new deadline;
$dead->set_date();
$dead->set_mode(1); // penambahan tanggal
$dead->set_tgl(0, 7); // 7 hari deadline
$waktu_naik = $dead->show_deadline();
// buat session untuk data
$_SESSION['reg_waktunaik'] = $waktu_naik;

$utama .= "
<!-- ciptakan kondisi terburu-buru sebagai penutup -->

<table border='1' cellpadding='5'>
<tr><td bgcolor='#f4f4f4'>
 <p align='center'>
 Hanya Tinggal 11 Pemesan!</p>
 <p>Karena semakin meningkatnya respon pasar terhadap e-book ini. Maka
 harga Rp. 100.000,- akan saya naikkan menjadi Rp. 200.000,-.
 Harga akan saya naikkan kemungkinan pada $waktu_naik.</p>
 <p>Saat ini saya hanya melayani 11 pemesan saja dengan harga
 Rp. 100.000,-. Oleh karena itu jika saat ini anda sudah memesan segeralah
 mentransfer agar anda tetap memperoleh harga yang terendah.</p>
</td></tr>
</table>
<center>

<form action='thankyou.php?res=$res' method='post'>
 <input type='submit' value='KLIK UNTUK MELANJUTKAN PEMESANAN'>
</form></center>

</td></tr>
</table>";

} // akhir dari else $pesan_err != ''

```

```

break;
*****;

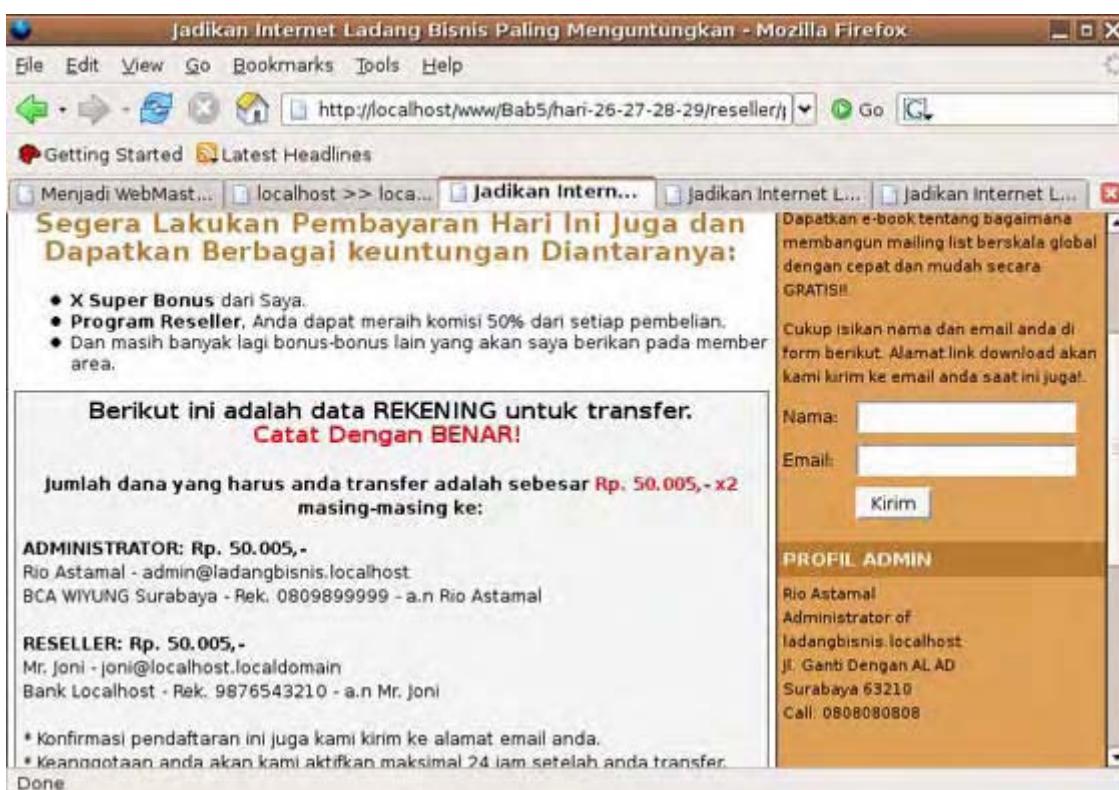
} // akhir dari switch()

$skin = new skin;
$skin->ganti_skin('template/index_skin.php');
$skin->ganti_tag('{TODAY}', "Hari ini: $hari_ini");
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KANAN}', $bonus.$me);
$skin->ganti_tag('{UTAMA}', $reseller.$utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

```

?>

Simpan pada direktori **reseller** dengan nama ***pesan.php***. Gambar 5.17 menunjukkan output dari file ***pesan.php***.



Gambar 5.17: Pembagian transfer saat pemesanan

## PENJELASAN SCRIPT

File *pesan.php* memiliki tiga case yaitu *case utama*, *case form*, dan *case pros\_form*. Pada *case utama* kita menampilkan peraturan dan perjanjian antara anda sebagai pemilik program dan pemesan. Istilah kerennya *End User Licence Agreement (EULA)*.

*Case form* pada case ini kita menampilkan prosedur pemesanan e-book dan menampilkan form registrasi.

*Case pros\_form* bertugas untuk memproses data yang dikirim case form. Karena data yang dikirim nantinya akan kita masukkan pada database. Maka sebelumnya kita melakukan berbagai pengecekan field yang kita simpan pada variabel \$pesan\_err. Jika tidak ada kesalahan maka kita masukkan data-data tersebut ke dalam session.

Mengapa tidak langsung menyimpannya di database? Pertanyaan bagus. Kita sebagai

programmer harus berfikir tentang efisiensi kerja. Jika kita langsung memasukkannya pada database saat pembagian transfer dan user tiba-tiba memutuskan untuk tidak memesan. Maka database kita akan dipenuhi oleh data sampah.

Selain itu jika user merefresh browsernya maka akan terjadi error karena duplikasi username. Oleh karena itu data-data tersebut kita simpan pada session terlebih dahulu. Dan jika user sudah benar-benar ingin memesan maka ia akan menekan tombol untuk lanjut. Nah, setelah itu data-data tersebut baru kita masukkan ke database dengan mengambilnya dari session.

Berikutnya file yang akan kita buat masih berhubungan dengan pemesanan. File ini akan memproses session yang dikirimkan oleh file *pesan.php* kemudian memasukkannya ke database. File ini juga bertugas mengirim email konfirmasi pendaftaran dan menghapus session pendaftaran yang telah digunakan.

Untuk membuatnya pada PHP DESIGNER klik menu **File > New > PHP** lalu salin script berikut.

#### Script dari file *thankyou.php*

---

```
<?php

/*
 * NamaFile: thankyou.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk memasukkan data ke database dan kirim email
 */

session_start();
// panggil file-file yang diperlukan
include ('include/class_skin.php');
include ('include/var_utama.php');

// ambil data pendaftaran dari session
$username = $_SESSION['reg_username'];
$userpass = $_SESSION['reg_password'];
$nama = $_SESSION['reg_nama'];
$kota = $_SESSION['reg_kota'];
$email = $_SESSION['reg_email'];
$transfer = $_SESSION['reg_transfer'];
$waktu_naik = $_SESSION['reg_waktunaik'];

// dapatkan profil reseller
$prof_res = SQL_array("SELECT * FROM tbl_profil WHERE username='$res'");

// masukkan ke tabel pemesan produk
$tgl = date('d-m-Y');
$hasil = mysql_query("INSERT INTO tbl_pemesan VALUES('$tgl', '$username',
 '$userpass', '$nama', '$email', '$kota', '$prof_res[username]',
 '$transfer')");
// cek status query
if (!$hasil)
 $utama = "

| |
|--|
| <p>Error: Gagal memasukkan data ke database. Mohon kontak admin dengan error berikut.

 MySQL Err: ".mysql_error()." </p> |
|--|

";
else
```

```

{
 // panggil file var_mail.php untuk template email
 include_once ('include/var_mail.php');

 // kirim email ke pada user yang baru saja mendaftar
 $to = $email;
 $subject = "Konfirmasi Pendaftaran Anda di ladangbisnis.localhost";
 $from = "From: admin@ladangbisnis.localhost";
 $isi = str_replace('{NAMA_U}', $nama, $mail_daftar);
 $isi = str_replace('{RUPIAH}', rupiah($transfer), $isi);
 $isi = str_replace('{NAMA_R}', $prof_res['nama'], $isi);
 $isi = str_replace('{BANK}', $prof_res['bank'], $isi);
 $isi = str_replace('{TELPON}', $prof_res['telpon'], $isi);
 $isi = str_replace('{EMAIL}', $prof_res['email'], $isi);
 $isi = str_replace('{WAKTU_NAIK}', $waktu_naik, $isi);

 // kirim email
 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail($to, $subject, $isi, $from);

 // update nilai pada tbl_transfer
 update(); // karena tanpa parameter maka yang diupdate adalah transfer

 // hapus session pendaftaran
 unset($_SESSION['reg_username']);
 unset($_SESSION['reg_password']);
 unset($_SESSION['reg_nama']);
 unset($_SESSION['kota']);
 unset($_SESSION['reg_email']);
 unset($_SESSION['reg_transfer']);
 unset($_SESSION['reg_waktunaik']);

 $utama = "
<table border='0' cellpadding='5'>
<tr><td bgcolor='#f4f4f4'>
<p>Terima Kasih

Saudara $nama, terima kasih karena saudara telah melakukan pemesanan
produk kami \"Internet Sebagai Ladang Bisnis\". Saya jamin dengan
e-book ini anda dapat merubah internet menjadi tempat kerja anda.</p>
</tr></td>
</table>";

} // akhir dari else !$hasil

$skin = new skin;
$skin->ganti_skin('template/index_skin.php');
$skin->ganti_tag('{TODAY}', "Hari ini: $hari_ini");
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KANAN}', $bonus.$me);
$skin->ganti_tag('{UTAMA}', $reseller.$utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **reseller** dengan nama ***thankyou.php***. Pada isi email kita juga menciptakan kondisi keterburu-buruan dengan mengatakan bahwa pemesan tinggal 11 saja. Dan harga akan naik 7 hari kemudian.

File yang akan kita buat berikutnya adalah file yang menampilkan daftar testimonial dari member. Buat file baru pada PHP DESIGNER 2006 kemudian ketik script berikut.

**Script dari file *testimonial.php***

```
<?php

/*
 * NamaFile: testimonial.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk menampilkan testimonial member
 */

// panggil file-file yang diperlukan
include ('include/class_skin.php');
include ('include/var_utama.php');
include ('include/class_halaman.php');

// ambil data dari URL
$page = filter_str($_GET['page']);
if ($page == '')
 $page = 0;

$utama = "
<table border='0' cellpadding='5'>
<tr><td>

Mereka Yang Sudah Menjadi Member!</p>
<table border='0' cellpadding='4'>";

// panggil class halaman
$hal = new halaman;
$hal->set_tabel('tbl_testimonial'); // nama tabel
$hal->set_page($page); // variabel yang dihandel
$hal->set_pph($u_tmph); // testimonial per halaman

$hal->query_SQL(1); // query SQL untuk mendapatkan jumlah data dsb.
$jml_data = $hal->get_jml_data(); // jumlah data
$jml_hal = $hal->get_jml_hal(); // jumlah halaman

$record = $hal->get_record(); // dapatkan jumlah record / baris
$hasil = $hal->query_SQL(3, 1, 1, 'tm_id'); // urut berdasar tm_id

// tampilkan semua testimonial dengan looping while
while ($data = mysql_fetch_array($hasil))
{
 $isi = nl2br(stripslashes($data['testimonial']));
 $utama .= "
<tr><td>$isi<hr></td></tr>";
}

$hal->set_hal();

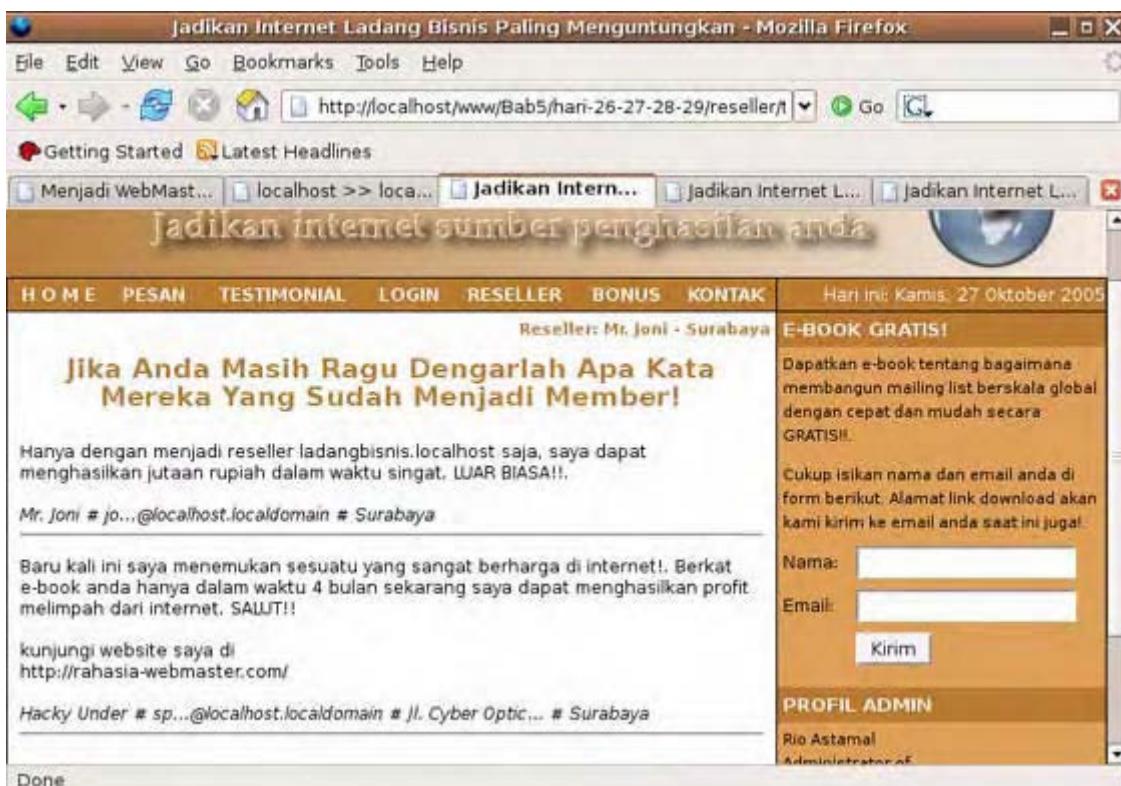
$utama .= "
</table>

.$hal->show_page("testimonial.php?res=$res")."
</td></tr>
</table>";

$skin = new skin;
$skin->ganti_skin('template/index_skin.php');
$skin->ganti_tag('{TODAY}', "Hari ini: $hari_ini");
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KANAN}', $bonus.$me);
$skin->ganti_tag('{UTAMA}', $reseller.$utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();
```

?&gt;

Simpan pada direktori **reseller** dengan nama ***testimonial.php***. Tidak ada yang baru pada file ini kita hanya memanggil class halaman untuk menampilkan data yang ada di database. Output dari file ini ditunjukkan oleh gambar 5.18 berikut.



Gambar 5.18: Daftar testimonial

File selanjutnya yaitu *login.php* bertugas untuk memproses login dan memproses proses lupa password. File ini akan mereset password user sesuai dengan email yang digunakan. Untuk membuatnya seperti biasa buat file PHP baru pada PHP DESIGNER 2006 kemudian salin script berikut.

#### Script dari file *login.php*

&lt;?php

```

/* NamaFile: login.php */
/* Penulis : Rio Astamal */
/* Tanggal : 25-10-2006 */
/* Fungsi : untuk memproses login member */

session_start(); // karena berhubungan dengan session

// panggil file-file yang diperlukan
include ('include/class_skin.php');
include ('include/var_utama.php');

// ambil data dari URL
$proses = filter_str($_GET['proses']);
if ($proses == '')
 $proses = 'utama';

// handel setiap proses dengan switch dan case
```

```

switch ($proses)
{
 /*****
 case 'utama':
 *****/
 $utama = "
<table border='0' cellpadding='5' width='100%'>
<tr><td><center>
<p class='besar'>Member Silahkan Login Di Sini</p>
<form action='login.php?proses=pros_login' method='post'>
<table border='1' cellpadding='4' bordercolor='#cgcgcg'>
<tr bgcolor='#ba7b31' height='30' align='center'>
 <td colspan='2' class='putih'>FORM LOGIN</td></tr>
<tr bgcolor='#f4f4f4'>
 <td>Username</td>
 <td>: <input type='text' name='username'></td></tr>
<tr bgcolor='#ffffff'>
 <td>Password</td>
 <td>: <input type='password' name='password'></td></tr>
<tr bgcolor='#ba7b31' align='center'><td colspan='2'>
 <input type='submit' value='Login'></td>
</tr>
</table>
</form>
<p>Lupa password?
 Klik di disini
</p>
</td></tr>
</table>";

break;
*****/

// *****
case 'pros_login':
// *****
// ambil data yang dipost
$username = filter_str($_POST['username']);
$password = filter_str($_POST['password']);

// enkripsi password
$password = balik_md5($password);

// cocokan username dan password menggunakan login()
if (!login('tbl_member', $username, $password))
 $utama = "<p align='center'>Username atau password salah
 .kembali</p>";
else
{
 // buatkan session karena user ini berhasil login
 $_SESSION['member'] = $username;
 header('Location: member/index.php');
} // akhir dari else !login()

break;

case 'lupa_pass':

$utama = "
<table border='0' cellpadding='5' width='100%'>

```

```

<tr><td><center>
<p class='besar'>Form Lupa Password</p>
<p>Masukkan email anda yang anda gunakan pada ladangbisnis.localhost</p>
<form action='login.php?proses=pros_lupapass' method='post'>
<table border='1' cellpadding='4' bordercolor='#cgcg'>
<tr bgcolor='#ba7b31' align='center'>
 <td colspan='2' class='putih'>FORM LUPA PASSWORD</td></tr></td>
<tr bgcolor='#f4f4f4'>
 <td>Email anda</td>
 <td><input type='text' name='email' size='30'></td></tr>
<tr>
 <td></td><td><input type='submit' value='Kirim'></td></tr>
</table>
</form></center>
</td></tr>
</table>";
break;
/*************/
/*proses_lupapass*/
/*************/

$email = $_POST['email'];
if (!cek_email($email))
 $utama = "<p class='error'>Error: Email tidak valid.$kembali</p>";
else
{
 // cek email dengan yang ada di database
 $hasil = mysql_query("SELECT * FROM tbl_member WHERE email='".$email'");
 if (mysql_num_rows($hasil) == 0)
 $utama = "<p class='error'>Maaf, email anda tidak terdaftar dalam
 database kami.$kembali</p>";
 else
 {
 $pass_baru = pass_acak(); // password baru (belum terenkrip)
 $enkrip_pass = balik_md5($pass_baru); // enkripsi password
 // update password yang ada di tabel member dan profil
 $update1 = mysql_query("UPDATE tbl_member SET password='".$enkrip_pass'
 WHERE email='".$email'");
 $update2 = mysql_query("UPDATE tbl_profil SET password='".$enkrip_pass'
 WHERE email='".$email'");
 // cek status
 if (!$update1 || !$update2)
 $utama = "<p class='error'>Error: Gagal mengupdate password.

 KET: ".mysql_error()."$.kembali</p>";
 else
 {
 // panggil file var_mail.php untuk template email
 include_once ('include/var_mail.php');
 // dapatkan data diri user
 $data = SQL_array("SELECT * FROM tbl_member WHERE email='".$email'");
 $tgl = date('d-m-Y');

 // kirim email
 $to = $email;
 $subject = "Password baru Anda di ladangbisnis.localhost";
 $header = "From: admin@ladangbisnis.localhost\n"
 ."Reply-To: no-reply@ladangbisnis.localhost\n";
 $isi = str_replace('{NAMA}', $data['nama'], $mail_lupa_pass);
 $isi = str_replace('{DATE}', $tgl, $isi);
 $isi = str_replace('{USERNAME}', $data['username'], $isi);
 $isi = str_replace('{PASSWORD}', $pass_baru, $isi);
 }
}

```

```

 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail ($to, $subject, $isi, $header);

 $utama = "<p align='center'>Password berhasil diupdate. Silahkan cek
 email anda untuk melihat password baru anda.</p>";
 } // akhir else !$update
} // akhir else mysql_num_rows()
} // akhir else !cek_email

break;
*****/
} // end of switch

$skin = new skin;
$skin->ganti_skin('template/index_skin.php');
$skin->ganti_tag('{TODAY}', "Hari ini: $hari_ini");
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KANAN}', $bonus);
$skin->ganti_tag('{UTAMA}', $reseller.$utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampil();

?>

```

Simpan pada direktori **reseller** dengan nama ***login.php***. Untuk login anda dapat masuk menggunakan username *joni*, *joko*, atau *jono* dengan password *pas123*. Tapi sabar itu nanti saja setelah semua file pada direktori *member* selesai dibuat.

## PENJELASAN SCRIPT

Sebenarnya prinsip kerja file ini sama dengan proyek-proyek sebelumnya. Di sini yang akan saya tekankan adalah pada case proses\_lupapass yang memproses form lupa password. Kita tidak menggunakan username melainkan email untuk meminimalisir aksi yang dilakukan oleh *"tangan-tangan jahil"*.

Jika kita menggunakan username seseorang yang usil akan dengan mudah memasukkan username yang ia inginkan pada form lupa password. Tanpa diketahui pemilik username, passwordnya telah berubah. Meskipun password tetap terkirim tetapi ini merugikan user yang bersangkutan.

Lain halnya jika kita menggunakan email. Sesorang yang berniat usil akan kesulitan karena ia tidak tahu alamat email orang yang ingin ia usili. Dengan demikian kita sudah melakukan upaya lebih untuk kenyamanan konsumen kita.

File berikutnya adalah file *reseller.php*. Tugas file ini hanya menampilkan keuntungan user jika menjadi reseller dari ladangbinis.localhost. Buat file PHP baru kemudian ketik kode dibawah ini.

### Script dari file ***reseller.php***

```

<?php
*****/*
/* NamaFile: reseller.php
/* Penulis : Rio Astamal
/* Tanggal : 25-10-2006
/* Fungsi : untuk menampilkan penjelasan tentang program reseller */
*****/

// panggil file-file yang diperlukan
include ('include/class_skin.php');
include ('include/var_utama.php');

```

```

$utama = "
<table border='0' cellpadding='5' width='100%'>
<tr><td>
 <p class='besar'>Dapatkan Komisi 50% atau Rp. 50.000,- Dari Setiap
 Pembelian Pada Website Anda</p>
 Dengan membeli e-book \"Internet Sebagai Ladang Bisnis\" maka secara
 otomatis anda menjadi reseller ladangbisnis.localhost. Dengan demikian
 Anda berkesempatan meraih komisi 50% atau sama dengan Rp.
 50.000,-.</p>
 Setelah menjadi member ladangbisnis.localhost anda secara otomatis
 akan memiliki duplikat website persis seperti ini. Namun dengan username
 anda sendiri contohnya,
http://ladangbisnis.locahost?res=
 username_anda</p>
 <p>Ketika user mengunjungi URL anda tersebut maka secara otomatis
 nama dan data bank anda yang akan muncul pada proses pembelian.
 DAN SETERUSNYA... DAN SETERUSNYA...</p>
</td></tr>
</table>";

$skin = new skin;
$skin->ganti_skin('template/index_skin.php');
$skin->ganti_tag('{TODAY}', "Hari ini: $hari_ini");
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KANAN}', $bonus.$me);
$skin->ganti_tag('{UTAMA}', $reseller.$utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **reseller** dengan nama **reseller.php**.

Sama dengan file berikutnya yaitu *bonus.php* bertugas menampilkan daftar bonus yang diperoleh user jika membeli produk dan juga memproses form bonus (E-book Gratis). Klik **File** > **New** > **PHP** kemudian ketik kode berikut.

#### Script dari file **bonus.php**

```

<?php

/*
 * NamaFile: bonus.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : menampilkan daftar bonus dan memproses form bonus
 */

// panggil file-file yang diperlukan
include ('include/class_skin.php');
include ('include/var_utama.php');

// dapatkan data dari URL
$proses = filter_str($_GET['proses']);
if ($proses == '')
 $proses = 'utama';

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*
 case 'utama':
 */
}
```

```

$utama = "
<table border='0' cellpadding='5' width='100%'>
<tr><td>
 <p class='besar'>Dapatkan X Super Bonus Senilai Ratusan Dollar Untuk
 Pemebelian Produk Utama Kami</p>
 <p>Dengan membeli produk utama yaitu e-book \"Internet Sebagai Ladang
 Bisnis\". Maka Anda dapat memiliki kesemua X super bonus tersebut.
 Hanya jika anda memesan sebelum $hari_deadline.</p>
 <p>Daftar dibawah ini adalah bonus-bonus yang akan anda dapatkan jika
 memesan sebelum $hari_deadline.</p>
 <p>DAN SETERUSNYA... DAN SETERUSNYA...</p>
</td></tr>
</table>";

break;
/*********/
case 'ebook':
/*********/

// ambil data yang dipost
// koma, titik, dan spasi perbolehkan
$nama = filter_str(ucwords($_POST['nama']), " \., ");
$email = $_POST['email'];

if (!cek_field($_POST))
 $utama = "<p align='center'>Error: Anda belum mengisi
 nama dan email Anda.$kembali</p>";
else
{
 // cek email
 if (!cek_email($email))
 $utama = "<p align='center'>Error: Email anda tidak
 valid.$kembali</p>";
 else
 {
 // masukkan nama dan emailnya ke tabel prospek
 $tgl = date('d-m-Y');
 $hasil = mysql_query("INSERT INTO tbl_prospek VALUES('$tgl', '$email',
 '$nama', '$res')");
 // cek status query
 if (!$hasil)
 $utama = "<p class='error'>Error: Gagal memasukkan data ke database.
 Kemungkinan email anda sudah ada dalam daftar kami. Gunakan alamat
 email lain.

Ket: ".mysql_error()." $kembali</p>";
 else
 {
 // dalam contoh ini kita mengirim email ke service autorespondor
 // yaitu freeautobot.com
 // jadi saya asumsikan anda sudah membuatnya di member area
 // freeautobot.com
 $to = 'ladangbisnis@freeautobot.com'; // account anda di freeautobot
 // email user (freeautobot akan mengirim email ke alamat ini)
 $from = "From: $nama <$email>";

 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail($to, '', '', $from);
 $utama = "<p align='center'>Alamat link download telah kami kirim ke
 email anda. Silahkan dicek!.</p>";
 } // akhir dari else !$hasil
 }
}

```

```

 } // akhir dari else !cek_email()
 } // akhir dari else !cek_field()

break;
***** */

} // akhir dari switch

$skin = new skin;
$skin->ganti_skin('template/index_skin.php');
$skin->ganti_tag('{TODAY}', "Hari ini: $hari_ini");
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KANAN}', $bonus.$me);
$skin->ganti_tag('{UTAMA}', $reseller.$utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>

```

---

Simpan pada direktori **reseller** dengan nama ***bonus.php***.

### PENJELASAN SCRIPT

Saya rasa case ebook perlu sedikit saya jelaskan. Pada case ini kita mengambil dua data yang dipost yaitu nama dan email. Data ini selain kita simpan pada database kita yaitu *tbl\_prospek*. Juga kita kirim ke servis autoresponder dalam hal ini saya asumsikan freeautbot.

Sebagaimana kita tahu jika kita mengirim email ke suatu alamat email autoresponder, misalnya [ladangbisnis@freeautobot.com](mailto:ladangbisnis@freeautobot.com). Maka secara otomatis kita akan masuk daftar follow up dari mesin autoresponder.

Jadi untuk isi email seperti link download bonus kita letakkan pada cycle pertama di autoresponder. Ketika user mensubmit form bonus maka ia secara otomatis akan dikirmi email oleh freeautbot.com yang sudah kita setting sebelumnya.

File yang terakhir pada direktori *reseller* adalah file *kontak.php*. Seperti namanya file ini bertugas untuk menyediakan form kontak bagi user agar dapat mengontak admin. Pada PHP DESIGNER 2006 buat file PHP baru kemudian salin kode berikut.

Script untuk file ***kontak.php***

```

<?php

/*
 * NamaFile: kontak.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : menampilkan form kontak admin dan memprosesnya
 */

// panggil file-file yang diperlukan
include ('include/class_skin.php');
include ('include/var_utama.php');

// dapatkan data dari URL
$proses = filter_str($_GET['proses']);
if ($proses == '')
 $proses = 'utama';

// handel setiap proses dengan switch dan case
switch ($proses)
{

}

```

---

```

case 'utama':
/******/
$utama = "
<table border='0' cellpadding='5' width='100%'>
<tr><td>
 <p class='besar'>Kontak Kami</p>
 <p>Jika anda merasa ada sesuatu yang kurang jelas atau anda ingin menanyakan sesuatu seputar ladangbisnis.localhost. Kami persilahkan anda untuk mengontak kami dengan mengisi form kontak berikut.</p>
<form action='kontak.php?proses=kirim' method='post'>
<table border='1' cellpadding='0'>
<tr bgcolor='#ba7b31' height='30'><td colspan='2' align='center' class='putih'>FORM KONTAK</td></tr>
<tr><td>
 <table border='0' cellpadding='4' bgcolor='#dea057'>
 <tr><td>Email Anda: </td>
 <td><input type='text' name='email' size='30'></td></tr>
 <tr><td>Subject: </td>
 <td><input type='text' name='subject' size='60'></td></tr>
 <tr><td>Pesan: </td>
 <td><textarea name='pesan' cols='60' rows='10'></textarea>
 <tr><td></td><td><input type='submit' value='Kirim'></td></td>
 </table>
 </td></tr>
</table>
</form>
</td></tr>
</table>";

break;
/******/
case 'kirim':
/******/

// ambil data yang dipost
$email = $_POST['email'];
$subject = $_POST['subject'];
$pesan = $_POST['pesan'];

// cek field
if (!cek_field($_POST) || !cek_email($email))
 $utama = "<p class='error'>Error: Masih ada field yang kosong atau email tidak valid.$kembali</p>";
else
{
 // kirim email ke admin
 $to = "admin@ladangbisnis.localhost"; // ganti dengan email anda
 $header = "From: $email\nReply-To: $email";
 // beri komentar pada mail() jika anda tidak berada pada server
 // sebenarnya atau anda tidak memiliki program mail server
 mail($to, $subject, $pesan, $header);

 $utama = "<p align='center'>Email berhasil dikirim.</p>";
}

break;
/******/
} // akhir dari switch

$skin = new skin;

```

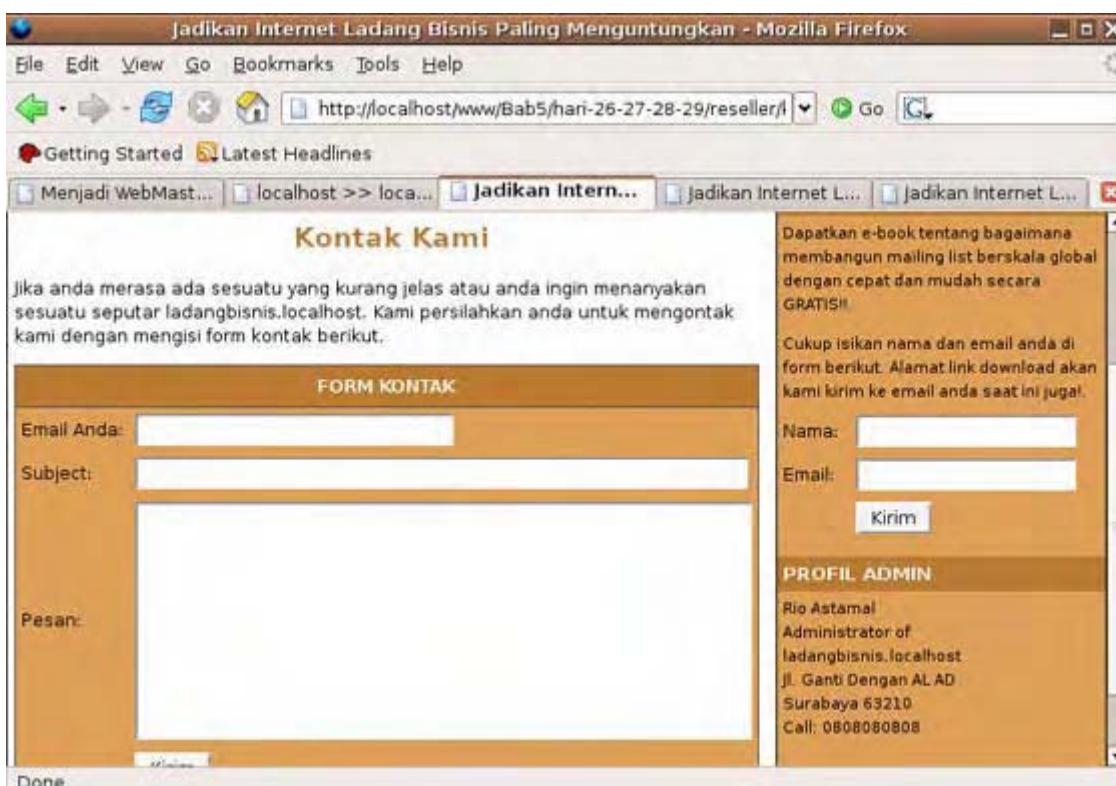
```

$skin->ganti_skin('template/index_skin.php');
$skin->ganti_tag('{TODAY}', "Hari ini: $hari_ini");
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KANAN}', $bonus.$me);
$skin->ganti_tag('{UTAMA}', $reseller.$utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **reseller** dengan nama ***kontak.php***. Output file ini ditunjukkan oleh gambar 5.19.



Gambar 5.19: Form kontak admin

Akhir dari file yang berada pada direktori **reseller**.

Berikutnya kita akan membuat file-file yang ada pada direktori *reseller/member*. File-file yang melayani semua aktivitas member. Untuk file yang pertama, masih seperti biasa kita akan membuat file *index.php* sebagai file wajib disetiap direktori.

Pada PHP DESIGNER 2006 klik menu **File > New > PHP** untuk membuat file PHP baru, kemudian salin script berikut ini.

Script dari file ***index.php***

```

<?php
/*
 * NamaFile: index.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk menampilkan halaman utama member area
 */
session_start();

```

```

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/member_var.php');

// cek apakah user sudah login atau belum
if (!cek_session('member'))
{
 header('Location: ../login.php');
 exit;
}
else
{

// dapatkan data dari URL
$proses = filter_str($_GET['proses']);
if ($proses == '')
 $proses = 'utama';

// handel setiap proses dengan switch dan case
switch ($proses)
{
 /*****
 case 'utama':
 ****/

 // dapatkan nama dari user
 $data = SQL_array("SELECT * FROM tbl_member WHERE username='$user'");

 $utama = "
<table border='0' width='100%' cellpadding='5'>
<tr><td>
 <p class='besar'>$data[nama], Selamat Datang Di Member Area</p>
 <p>Terima kasih karena anda telah membeli produk kami yaitu \"Internet Sebagai Ladang Bisnis\". Saya yakin anda akan memperoleh sesuatu yang sangat bermanfaat dari e-book tersebut.</p>
 <p>Karena anda sudah ada di member area, anda dapat mendownload produk kami beserta ke-x super bonus yang telah kami janjikan sebelumnya. DAN SETERUSNYA... DAN SETERUSNYA...</p>

 <!-- Yakinkan Member Agar Mengikuti Program Reseller -->
 <p class='besar'>Program Reseller</p>
 <p>Jika anda menginginkan penghasilan secara instan. Maka Saya sarankan kepada anda untuk mengikuti program reseller dari ladangbisnis.localhost. Karena anda cukup merefensikan situs ini melalui URL yang kami berikan. Setiap member yang masuk lewat URL anda maka anda secara otomatis akan menerima komisi 50% atau sebesar Rp. 50.000,-.</p>
 <p>Alamat URL yang anda promosikan adalah:

 http://ladangbisnis.localhost?res=$user</p>
 <p>DAN SETERUNYA... DAN SETERUSNYA...</p>

 <!-- Yakinkan Member Untuk Memberikan Testimonialnya -->
 <p class='besar'>Testimonial</p>
 <p>Setelah anda membaca dan menerapkan teori yang ada pada e-book \"Internet Sebagai Ladang Bisnis\". Atau anda mempunyai komentar tentang ladangbisnis.localhost entah itu berupa kritik atau saran. Saya persilahkan anda untuk memberikan testimonial anda. Silahkan klik link Testimonial. Akan ada bonus spesial untuk anda.
</td></tr>
</table>";

```

```

break;
***** */

***** */
case 'logout':
***** */
if (!logout('member'))
 header('Location: ../login.php');
else
{
 $utama = "<p align='center'>Anda telah logout. Klik di sini untuk login kembali.</p>";
}

break;
***** */

} // akhir dari switch

} // akhir dari else !cek_session

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $mem_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **reseller\member** dengan nama ***index.php***.



Gambar 5.20: Halaman utama member area

Tidak ada yang istimewa pada script ini. Karena script ini hanya bertugas untuk menampilkan beberapa kata "mutiara" untuk member dan memproses logout.

File selanjutnya yang akan kita buat adalah file yang bertugas untuk mengupdate profil dari user. File ini kita namakan *profil.php*. Buat file PHP baru pada PHP DESIGENR 2006 kemudian salin script berikut.

#### Script dari file *profil.php*

```
<?php

/*
 * NamaFile: profil.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk mengupdate profil member
 */

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/member_var.php');

// cek apakah user sudah login atau belum
if (!cek_session('member'))
{
 header('Location: ../login.php');
 exit;
}
else
{
 // dapatkan data dari URL
 $proses = filter_str($_GET['proses']);
 if ($proses == '')
 $proses = 'utama';

 switch ($proses)
 {

 /*
 case 'utama':
 */

 // dapatkan profil dari user
 $data = SQL_array("SELECT * FROM tbl_profil WHERE username='$user'");

 $utama = "
<table border='0' cellpadding='5' width='100%>
<tr><td>
 <p class='besar'>Update Profil</p>
 <center>
 <form action='profil.php?proses=update' method='post'>
 <table border='1' cellpadding='5' bordercolor='#cgcgcf' bgcolor='#ffffff'>
 <tr bgcolor='#ba7b31' align='center'>
 <td class='putih' colspan='2'>FORM UPDATE PROFIL</td></tr>
 <tr bgcolor='#f4f4f4'><td>Username</td><td>$user</td></tr>
 <tr><td>Password</td>
 <td><input type='password' name='password1' maxlength='16'></td></tr>
 <tr bgcolor='#f4f4f4'><td>Ulangi</td>
 <td><input type='password' name='password2' maxlength='16'></td></tr>
 <tr><td>Nama Lengkap</td>
 <td><input type='text' name='nama' value='".$data[nama]."'></td></tr>
 <tr bgcolor='#f4f4f4'><td>Email</td>
```

```

<td><input type='text' name='email' value='$data[email]'></td></tr>
<tr><td>Alamat</td>
 <td><input type='text' name='alamat' value='$data[alamat]' size='50'></td></tr>
<tr bgcolor='#f4f4f4'><td>Kota</td>
 <td><input type='text' name='kota' value='$data[kota]'></td></tr>
<tr><td>Telp./HP</td>
 <td><input type='text' name='telpon' value='$data[telpon]'></td></tr>
<tr bgcolor='#f4f4f4'><td>Bank</td>
 <td><input type='text' name='bank' value='$data[bank]' size='50'></td></tr>
<tr><td>Contoh</td>
 <td>Bank Virtual - Rek. 0101010101 - a.n Rio Astamal</td></tr>
<tr bgcolor='#f4f4f4'><td><input type='submit' value='Update Profil'></td></tr>
</td></tr>
</table>
</center></form>
</td></tr>
</table>";

break;
/************/
/*update*/
/************/

// ambil data-data yang dipost
$password1 = filter_str($_POST['password1']);
$password2 = filter_str($_POST['password2']);
$nama = filter_str($_POST['nama'], " \.,"); // spasi, titik, koma boleh
$email = $_POST['email'];
$alamat = filter_str($_POST['alamat'], " \.,");
$kota = filter_str($_POST['kota'], " \.,");
$telpon = filter_str($_POST['telpon'], " -"); // spasi, - boleh
$bank = filter_str($_POST['bank'], " -\.,\/"); // -, titik, / boleh

// cek untuk error
$pesan_err = '';

if (!cek_field($_POST))
 $pesan_err = "Error: Masih ada field yang kosong.
";

if (!cek_email($email))
 $pesan_err .= "Error: Email tidak valid.
";

if (strlen($password1) < 6)
 $pesan_err .= "Error: Password harus diantara 6-16 karakter.
";

if ($password1 != $password2)
 $pesan_err .= "Error: Password tidak sama.
";

// cek status
if ($pesan_err != '')
 $utama = "<p class='error'>$pesan_err$kebalikan</p>";
else
{
 // enkripsi password
 $password1 = balik_md5($password1);

 // update database
 // update tabel profil
 $hasil = mysql_query("UPDATE tbl_profil SET password='$password1',

```

```

nama='$nama', email='$email', kota='$kota', alamat='$alamat',
telpon='$telpon', bank='$bank' WHERE username='$user'");

// update password pada tabel member
mysql_query("UPDATE tbl_member SET password='$password1' WHERE
username='$user'");

// cek status
if (!$hasil)
$utama = "<p class='error'>Gagal mengupdate database. Kontak
Admin. $kembali</p>";
else
$utama = "<p align='center'>Profil berhasil diupdate.$kembali</p>";

} // akhir dari $pesan_err != ''
break;
***** */

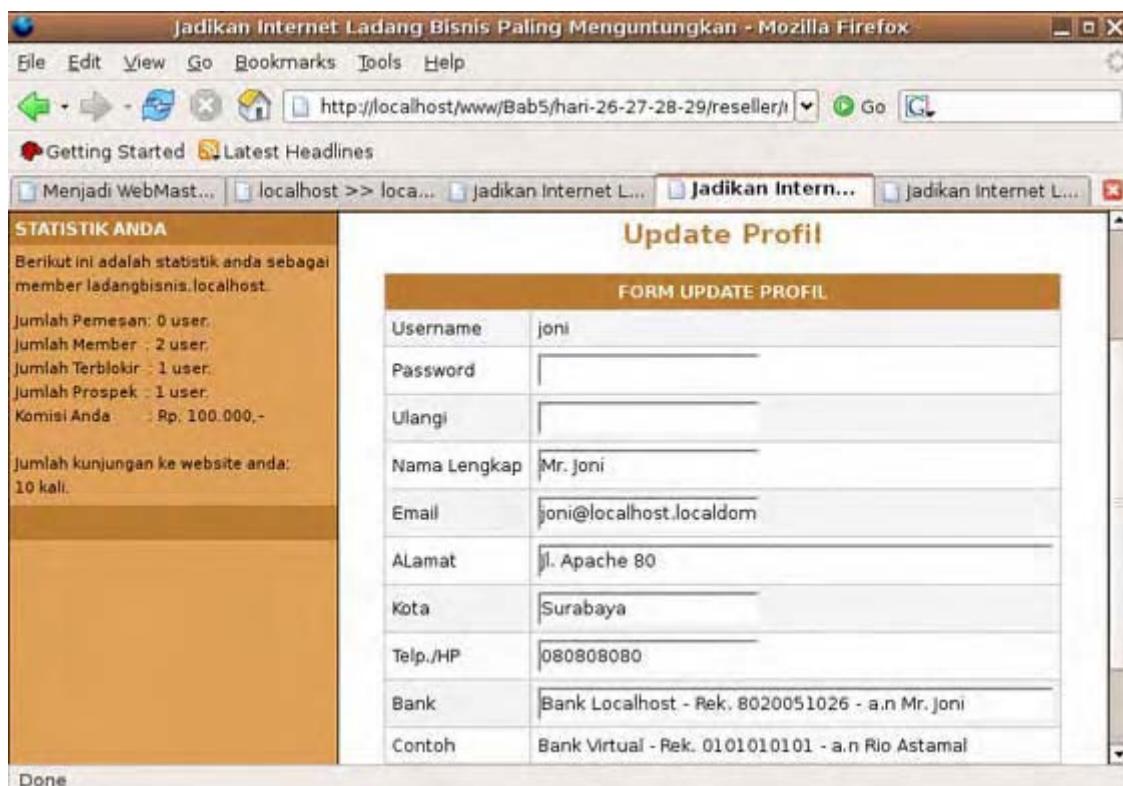
} // akhir dari switch

} // akhir dari else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $mem_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();
?>

```

Simpan pada direktori **reseller\member** dengan nama ***profil.php***.



Gambar 5.21: Output file **profil.php**

Pada file ini kita hanya melakukan mendapatkan data dari database kemudian mengupdatenya

sesuai dengan yang diisikan user. Untuk password kita mengupdate pada dua tabel yaitu tabel tbl\_profil dan tbl\_member.

Selanjutnya kita akan membuat file yang menampilkan halaman download e-book utama yaitu "Internet Sebagai Ladang Bisnis\". Langsung saja buat file PHP baru kemudian ketik kode berikut.

Script dari file ***download.php***

---

```
<?php

/*
 * NamaFile: download.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk menampilkan halaman download
 */

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/member_var.php');

// cek apakah user sudah login atau belum
if (!cek_session('member'))
{
 header('Location: ../login.php');
 exit;
}
else
{

$utama = "
<table border='0' width='100% cellpadding='5'>
<tr><td>
 <p class='besar'>Download Area</p>
 <p>"Internet Sebagai Ladang Bisnis\" saya kemas dalam bentuk file PDF.

 Jadi anda memerlukan Software yang dapat membaca file PDF untuk membaca

 e-book ini. Adobe(R) Acrobat(R) Reader merupakan software yang paling

 banyak digunakan dilingkungan Windows(R).</p>
 <p>Versi minimal dari Acrobat Reader yang digunakan untuk dapat membaca

 e-book ini adalah versi 5.x ke atas. Untuk mendownload Acrobat Reader

 klik di sini.</p>
 <p>

 Download \"Internet Sebagai Ladang Bisnis\"</p>
 <p>DAN SETERUSNYA... DAN SETERUSNYA...</p>
</td></tr>
</table>";

} // akhir dari else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $mem_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();
?>
```

---

Simpan pada direktori **reseller\member** dengan nama ***download.php***.

Pada file download.php kita hanya menampilkan halaman untuk link download e-book. Karena produknya belum ada maka linknya kita isi sembarang saja.

File selanjutnya yang akan kita buat adalah file yang paling rumit pada direktori *reseller/member*. Prinsip kerja file ini sama dengan file statistik.php yang ada pada direktori *reseller/\_my\_cp\_*. File ini bertugas untuk menampilkan statistik member, pemesan, prospek atau terblokir yang bergabung lewat rekomendasi reseller itu sendiri.

Untuk membuatnya klik menu **File > New > PHP** pada PHP DESIGNR 2006 kemudian ketik script berikut ini.

#### Script dari file **statistik.php**.

---

```
<?php

/***
/* NamaFile: statistik.php
/* Penulis : Rio Astamal
/* Tanggal : 25-10-2006
/* Fungsi : untuk statistik dari member
/**

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/member_var.php');
include ('../include/class_halaman.php');

// cek apakah user sudah login atau belum
if (!cek_session('member'))
{
 header('Location: ../login.php');
 exit;
}
else
{

// dapatkan data dari URL
$proses = filter_str($_GET['proses']);
if ($proses == '')
 $proses = 'utama';

$page = filter_str($_GET['page']);
if ($page == '')
 $page = 0;

$back_to = "<< Kembali ke Statistik";

switch ($proses)
{

/***
case 'utama':
/***

$utama = "
<table border='0' cellpadding='5' width='100%'>
<tr><td>
 <p class='besar'>Statistik Anda</p>
 <p>Pada halaman ini anda dapat melihat statistik dari website anda.
 Berapa kali website anda dikunjungi, berapa member yang berhasil anda
 rekrut dan lain-lain.</p>
 <p>Jumlah kunjungan ke website anda http://ladangbisnis.localhost?res=$user

```

```

adalah sebanyak <u>$traj[1]</u> kali.</p>
<table border='1' cellpadding='0' width='50%'>
<tr><td>
 <table border='0' cellpadding='5' width='100%'>
 <tr bgcolor='#ba7b31' height='30' class='putih'>
 <td>STATISTIK</td>
 <td>JUMLAH</td></tr>
 <tr bgcolor='#f4f4f4'><td>Jumlah Pemesan</td><td>
 $jml_pemesan user
 </td></tr>
 <tr bgcolor='#ffffff'><td>Jumlah Member</td><td>
 $jml_member user
 </td></tr>
 <tr bgcolor='#f4f4f4'><td>Jumlah Terblokir</td><td>
 $jml_blokir user
 <tr bgcolor='#ffffff'><td>Jumlah Prospek</td><td>
 $jml_prospek user
 <tr bgcolor='#f4f4f4'><td>Jumlah Komisi</td>
 <td>" .rupiah($komisi). "
 </td></tr>
 </table>
</td></tr>
</table>
<p>Untuk melakukan pencarian data klik di sini
</p>
</td></tr>
</table>";
break;
/*********/
/*********/
case 'lihat':
/*********/
// kosongkan hari dan statistik agar tabel lebih lebar
$hari_ini = ''; $mem_stat = '';

// ambil data dari URL
// nama tabel disamarkan untuk alasan keamanan
$tabel = filter_str($_GET['data']); // nama tabel belum lengkap
if ($tabel == '')
 $tabel = 'pemesan';

$tabel = "tbl_.$tabel"; // nama tabel lengkap

// buat array untuk header kolom (pemesan dan terblokir)
$header_arr = array("Tgl. Join", "Username", "Nama", "Email", "Kota",
 "Reseller", "Transfer");
$order_by = 'transfer'; // kolom yg diurutkan pada query

if ($tabel == 'tbl_member')
{
 $jdl = "
<p class='besar'>Daftar Member</p>
<p>Statistik berikut menunjukkan member yang masuk lewat rekomendasi
anda.</p>";
$header_arr[] = "Blokir?"; // tambah satu kolom
$java = $java_konfirm; // javascript dialog box
$bawah =
<p>Jika ada member dari daftar diatas yang belum menyelesaikan pembayaran
kepada anda. Maka anda berhak untuk memblokir keanggotaan mereka. Tekan
Tombol Blokir untuk melakukannya.</p>";

```

```

}

else if ($tabel == 'tbl_pemesan')
$jd1 = "
<p class='besar'>Daftar Pemesan</p>
<p>Statistik berikut menunjukkan pemesan e-book \"Internet Sebagai
Ladang Bisnis\" yang masuk lewat rekomendasi anda.</p>";
else if ($tabel == 'tbl_blokir')
{
$jd1 = "
<p class='besar'>Daftar Terblokir</p>
<p>Statistik berikut menunjukkan member yang keanggotaannya anda
blokir.</p>";
$bawah = "
<p>Jika member yang anda blokir telah menyelesaikan pembayarannya.
Anda dapat mengirim email ke admin@ladangbisnis.localhost. Dengan
subject \"Re-Aktivasi Member\". Pada isi email tuliskan username
dari member yang ingin anda aktifkan.</p>
<p>Anda dapat menggunakan
Form Kontak untuk mengirim email.</p>";
}
else
{
$jd1 = "
<p class='besar'>Daftar Prospek</p>
<p>Statistik berikut menunjukkan prospek yang masuk rekomendasi anda.</p>;
// tindih/overwrite $header_arr dengan yang baru
$header_arr = array("Tgl. Masuk", "Email", "Nama", "Reseller");
$order_by = "tgl_masuk"; // kolom yang akan diurutkan
}

// panggil class halaman
$hal = new halaman;
$hal->set_tabel($tabel);
$hal->set_page($page);
$pph = $hal->set_pph($u_pph); // ada di konfig.php

// lakukan query untuk mendapatkan banyaknya data dan jumlah halaman
$hal->query_SQL(2, 'reseller', $user);
$jml_data = $hal->get_jml_data();
$jml_hal = $hal->get_jml_hal();

$utama = "
$java
<table border='0' cellpadding='5' width='100%'>
<tr><td>
$jd1
<table border='1' cellpadding='4' width='785' bordercolor='#cgcgcg'>
<tr bgcolor='#ba7b31' class='putih' height='30'>
.buat_kolom($header_arr, 'th')."</tr>";
.buat_kolom($header_arr, 'td');

// lakukan query ulang
$hal->get_record();
$hasil = $hal->query_SQL(3, 'reseller', $user, $order_by);

// looping untuk mendaftar semua prospek
while ($data = mysql_fetch_array($hasil))
{
 if ($color == '#f4f4f4')
 $color = '#ffffff';
 else
 $color = '#f4f4f4';

 // buat array untuk kolom
 $kolom_arr = array($data[0], $data[1], $data[3], $data[4], $data[5],

```

```

 $data[6], rupiah($data[7]));
// jika tabelnya member tambahkan tombol blokir
// jika prospek tindih dengan array yang baru
if ($tabel == 'tbl_member')
{
 $pesan = "Apakah anda yakin akan memblokir keanggotaan dari\n"
 . "Saudara $data[3]?" ;
 $form =
 <form action='statistik.php?proses=blokir' method='post'>
 <input type='hidden' name='username' value='$data[1]'>
 <input type='submit' value='Blokir' onClick=\"return
konfirmasi('$pesan')\">
 </form>;
 $kolom_arr[] = $form;
}
else if ($tabel == 'tbl_prospek')
 $kolom_arr = array($data[0], $data[1], $data[2], $data[3]);

$utama .= "
<tr bgcolor='$color' height='25'>
.buat_kolom($kolom_arr, 'td', 'align=center' class='sf') .</tr>";
} // akhir while

$hal->set_hal(); // beri nilai untuk navigasi halaman
$tabel = explode("_", $tabel); // pecah nama tabel (menjadi array)

$utama .= "
</table>

.$hal->show_page("statistik.php?proses=lihat&data=$tabel[1]")."
$bawah
$kembali
</td></tr>
</table>";

break;
/************/
/* **** */
case 'blokir':
/* **** */

$username = $_POST['username']; // ambil data yang dipost
// panggilkan fungsi pindah_data() untuk memindahkan data dari
// tabel member ke tabel blokir
$hasil = pindah_data('tbl_member', 'tbl_blokir', 'username', $username);
if (!$hasil)
 $utama = "<p class='error'>Gagal melakukan blokir member. Hubungi "
 . " Admin.$kembali</p>";
else
{
 // panggil file var_mail.php untuk template email
 // pastikan hanya filenya dipanggil satu kali dengan include_once()
 include_once ('../include/var_mail.php');

 // array profil dari reseller dan member yang diblokir
 $prof_res = SQL_array("SELECT * FROM tbl_profil WHERE username=' $user '");
 $prof_mem = SQL_array("SELECT * FROM tbl_blokir WHERE username=' $username '");

 // kirim email
 $to = $prof_mem['email'];
 $subject = "Keanggotaan anda di ladangbinis.localhost kami BLOKIR";
 $header = "From: admin@ladangbinis.localhost";
 // Ganti tag {XXX} dengan data yang sesuai
 $isi = str_replace('{NAMA_U}', $prof_mem['nama'], $mail_res_blokir);
}

```

```

$isi = str_replace('{NAMA_R}', $prof_res['nama'], $isi);
$isi = str_replace('{USERNAME}', $prof_res['username'], $isi);
$isi = str_replace('{EMAIL}', $prof_res['email'], $isi);
$isi = str_replace('{TELPON}', $prof_res['telpon'], $isi);
$isi = str_replace('{BANK}', $prof_res['bank'], $isi);
// beri komentar pada mail() jika anda tidak berada pada server
// sebenarnya atau anda tidak memiliki program mail server
mail ($to, $subject, $isi, $header);

$utama = "<p align='center'>Username $username berhasil diblokir.
$kembali</p>";
} // akhir dari else !$hasil

break;

} // akhir dari switch

} // akhir dari else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $mem_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>

```

Simpan pada direktori **reseller\member** dengan nama **statistik.php**. Output dari file ini ditunjukkan oleh gambar 5.22.

The screenshot shows a Mozilla Firefox window with the title "Jadikan Internet Ladang Bisnis Paling Menguntungkan - Mozilla Firefox". The address bar shows the URL "http://localhost/www/Bab5/hari-26-27-28-29/reseller/". The page content is titled "Daftar Member" and contains the following text:

Statistik berikut menunjukkan member yang masuk lewat rekomendasi anda.

Tgl. Join	Username	Nama	Email	Kota	Reseller	Transfer	Blokir?
27-10-2005	hacky	Hacky Under	spy@localhost.localdomain	Surabaya	joni	Rp. 50.005,-	<input type="button" value="Blokir"/>
26-10-2005	joko	Mr. Joko.	joko@localhost.localdomain	Surabaya	joni	Rp. 50.002,-	<input type="button" value="Blokir"/>

Jika ada member dari daftar diatas yang belum menyelesaikan pembayaran kepada anda. Maka anda berhak untuk memblokir keanggotaan mereka. Tekan Tombol Blokir untuk melakukannya.

<< Kembali

Gambar 5.22: Output statistik.php saat menampilkan daftar member

## PENJELASAN SCRIPT

Sama dengan file *statistik.php* yang ada pada direktori *\_my\_cp\_*. File ini juga melakukan aksi

berdasarkan tabel yang diinginkan. Namun untuk alasan keamanan kita tidak memunculkan nama tabel pada URL. Karena setiap nama tabel kita berawalan "tbl\_", maka kita menambahkannya setelah data difilter.

Untuk statistik member kolomnya sedikit berbeda karena kita menyediakan satu kolom tambahan untuk tombol blokir. Jadi pada array \$header\_arr kita menambahkan "Blokir?" didalamnya. Jadi pada looping while kita juga menambahkan sebuah tombol untuk blokir.

```
if ($tabel == 'tbl_member')
{
 $pesan = "Apakah anda yakin akan memblokir keanggotaan dari\n"
 . "Saudara $data[3]?";
 $form = "
<form action='statistik.php?proses=blokir' method='post'>
<input type='hidden' name='username' value='$data[1]'>
<input type='submit' value='Blokir' onClick=\"return
konfirmasi('$pesan')\">
</form>";
 $kolom_arr[] = $form;
}
```

Pada input submit kita menambahkan attribut onclick yang akan menghandel even klik dari user. Jika tombol diklik maka akan muncul Box dialog seperti ditunjukkan oleh gambar 5.x

Jika kita mengklik Cancel maka aksi dibatalkan, namun jika kita mengklik OK maka proses pemblokiran akan dilakukan. Maka selanjutnya blok kode pada *case blokir* akan dijalankan. Kita memindahkan member menggunakan fungsi pindah\_data().



Gambar 5.23: Dialog box konfirmasi saat tombol diklik

Email yang berisi tagihan akan langsung dikirim ke alamat email member yang anda blokir tersebut. Jika ia sudah melunasi kewajibannya maka member dapat mengirim email kepada anda untuk aktivasi kembali user tersebut.

File berikutnya yang akan kita buat adalah file yang bertugas untuk mengirimkan testimonial member ke email anda sebagai administrator. Untuk mengaplikasikannya buat file PHP baru kemudian salin script berikut.

Script dari file ***testimonial.php***

```
<?php
/*
 * NamaFile: testimonial.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : memproses testimonial member
*/
session_start();
// panggil file-file yang diperlukan
```

```

include ('../include/class_skin.php');
include ('../include/member_var.php');

// cek apakah user sudah login atau belum
if (!cek_session('member'))
{
 header('Location: ../login.php');
 exit;
}
else
{
 // dapatkan data dari URL
 $proses = filter_str($_GET['proses']);
 if ($proses == '')
 $proses = 'utama';

 switch ($proses)
 {

 //*****
 case 'utama':
 //*****

 // dapatkan profil dari member
 $data = SQL_array("SELECT * FROM tbl_profil WHERE username='$user'");

 $utama = "
<table border='0' width='100%' cellpadding='5'>
<tr><td>
 <p class='besar'>Testimonial</p>
 <p>Setelah anda membaca e-book \"Internet Sebagai Ladang Bisnis\" dan menerapkannya. Atau anda mempunyai komentar, kritik dan saran tentang ladangbisnis.localhost. Saya persilahkan anda untuk mengisi testimonial berikut.</p>
 <p>Data-data yang menyangkut kepentingan pribadi seperti email dan alamat akan kami tampilkan secara samar.</p>
 <center>
 <form action='testimonial.php?proses=kirim' method='post'>
 <table border='1' cellpadding='5' bordercolor='#cgcgcg' bgcolor='#ffffff'>
 <tr bgcolor='#ba7b31' align='center'>
 <td colspan='2' class='putih'>FORM TESTIMONIAL</td></tr>
 <tr bgcolor='#f4f4f4'><td>Subject</td>
 <td><input type='text' name='subject' value='TESTIMONIAL - $user' readonly='yes'></td></tr>
 <tr><td>Nama *</td>
 <td><input type='text' name='nama' value='".$data[nama].'"></td></tr>
 <tr bgcolor='#f4f4f4'><td>Email *</td>
 <td><input type='text' name='email' value='".$data[email].'"></td></tr>
 <tr><td>Alamat *</td>
 <td><input type='text' name='alamat' size='50' value='".$data[alamat]."'></td></tr>
 <tr bgcolor='#f4f4f4'><td>Kota *</td>
 <td><input type='text' name='kota' value='".$data[kota]."'></td></tr>
 <tr><td>Website (Jika ada)</td>
 <td><input type='text' name='website' value='http://' size='50'></td></tr>
 <tr bgcolor='#f4f4f4'><td>Testimonial *</td>
 <td><textarea name='testimonial' rows='10' cols='55'></td></tr>
 </td></tr>
 <tr><td></td><td class='sf'>Field yang bertanda * harus diisi.</td></tr>
 <tr bgcolor='#f4f4f4'><td></td><td class='sf'>
 Data berikut akan ditampilkan atau tidak ketika testimonial anda ditampilkan.</td></tr>
 <tr><td>Tampilkan Email?</td>

```

```

<td><input type='radio' name='r_email' value='Ya'>Ya
<input type='radio' name='r_email' value='No'>Tidak</td></tr>
<tr bgcolor='#f4f4f4'>
 <td>Tampilkan Alamat?</td>
 <td><input type='radio' name='r_alamat' value='Ya'>Ya
 <input type='radio' name='r_alamat' value='No'>Tidak</td></tr>
<tr><td>Tampilkan Kota?</td>
 <td><input type='radio' name='r_kota' value='Ya'>Ya
 <input type='radio' name='r_kota' value='No'>Tidak</td></tr>
<tr bgcolor='#f4f4f4'><td>Tampilkan Website</td>
 <td><input type='radio' name='r_website' value='Ya'>Ya
 <input type='radio' name='r_website' value='No'>Tidak</td></tr>
<tr><td></td><td><input type='submit' value='Kirim'></td></tr>
</table>
</form></center>
</td></tr>
</table>";
break;
***** */

***** */
case 'kirim':
***** */
// ambil data-data yang dipost
$_subject = $_POST['subject'];
$_nama = $_POST['nama'];
$_email = $_POST['email'];
$_alamat = $_POST['alamat'];
$_kota = $_POST['kota'];
$_website = $_POST['website'];
$_testimonial = $_POST['testimonial'];
$_r_email = $_POST['r_email'];
$_r_alamat = $_POST['r_alamat'];
$_r_kota = $_POST['r_kota'];
$_r_website = $_POST['r_website'];

// cek setiap field
$pesan_err = '';

if (!cek_field($_POST))
 $pesan_err = "Error: Masih ada field yang kosong.
";

if (!cek_email($_email))
 $pesan_err .= "Error: Email tidak valid.
";

// jika tidak ada string http:// maka false
if (!strstr($_website, 'http://'))
 $pesan_err .= "Error: URL website tidak valid.
";

if ($pesan_err != '')
 $utama = "<p align='center'>$pesan_err$kembali</p>";
else
{
 $to = "admin@ladangbisnis.localhost"; // ganti dg email anda
 $header = "From: $_nama <$_email>";
 $isi = $_testimonial."\n\n"
 ."===== PILIHAN USER =====\n"
 ."Tampilkan Email : $_email => $_r_email\n"
 ."Tampilkan Alamat : $_alamat => $_r_alamat\n"
 ."Tampilkan Kota : $_kota => $_r_kota\n"
 ."Tampilkan Website: $_website => $_r_website\n"
 ."=====";
 // kirim email
 // beri komentar pada mail() jika anda tidak berada pada server
}

```

```

// sebenarnya atau anda tidak memiliki program mail server
mail($to, $subject, $isi, $header);

$utama = "<p align='center'>Testimonial berhasil dikirim.</p>" ;
} // akhir dari $pesan_err != ''

break;

} // akhir dari switch

} // akhir dari else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $mem_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();
?>

```

Simpan pada direktori **reseller\member** dengan nama ***testimonial.php***. Output dari file ini dapat anda lihat pada gambar 5.24.

Setelah menerima email testimonial dari user anda sebagai admin dapat memasukkan testimonial member tersebut melalui form tambah testimonial pada admin area. Data-data seperti email, alamat, kota, website anda tampilkan atau tidak tergantung dari pilihan user. Dimana pada bagian bawah email terdapat pilihan-pilihan user.



Gambar 5.24: Tampilan form testimonial member

Dan yang terakhir untuk direktori ini dan juga proyek ladangbinis.localhost adalah file ***strategi.php***. Fungsi file ini hanya menampilkan strategi-strategi yang anda berikan kepada reseller agar promosi mereka efektif.

Pada PHP DESIGNER 2006 klik menu **File > New > PHP** lalu salin kode berikut.

---

Script dari file ***strategi.php***

---

```
<?php

/*
 * NamaFile: strategi.php
 * Penulis : Rio Astamal
 * Tanggal : 25-10-2006
 * Fungsi : untuk menampilkan halaman strategi promosi
 */

session_start();

// panggil file-file yang diperlukan
include ('../include/class_skin.php');
include ('../include/member_var.php');

// cek apakah user sudah login atau belum
if (!cek_session('member'))
{
 header('Location: ../login.php');
 exit;
}
else
{

// tunjukkan pada member anda bagaimana strategi promosi yang
// efektif dan menguntungkan
$utama = "
<table border='0' cellpadding='5' width='100%'>
<tr><td>
 <p class='besar'>Strategi Promosi Program Reseller
 ladangbisnis.localhost</p>
 <p>Setiap member secara otomatis juga menjadi reseller program
 ladangbisnis.localhost. Jadi berkesempatan untuk mendapatkan komisi
 50% atau Rp. 50.000,- dari setiap member yang masuk lewat rekomendasi
 anda.</p>
 <p>Sekedar mengingatkan URL yang anda promosikan adalah:

 http://ladangbisnis.localhost?res=$user</p>
 <p>Semakin banyak user yang mengunjungi URL anda semakin besar pula
 kesempatan anda untuk memperoleh komisi. Namun untuk dapat menghasilkan
 banyak pengunjung. Diperlukan cara-cara yang efektif dalam promosi. Pada
 halaman inilah anda akan menemukan startegi promosi yang efektif.</p>
 <p>DAN SETERUSNYA... DAN SETERUSNYA...</p>
</td></tr>
</table>";

} // akhir dari else !cek_session()

$skin = new skin;
$skin->ganti_skin('../template/member_skin.php');
$skin->ganti_tag('{TODAY}', $hari_ini);
$skin->ganti_tag('{MENU}', $menu);
$skin->ganti_tag('{KIRI}', $mem_stat);
$skin->ganti_tag('{UTAMA}', $utama);
$skin->ganti_tag('{FOOTER}', $footer);
$skin->ganti_tampilan();

?>
```

---

Simpan pada direktori **reseller\member** dengan nama ***strategi.php***.

Kamis, 27 Oktober 2005 23:15

**STATISTIK ANDA**

Berikut ini adalah statistik anda sebagai member jadangbisnis.localhost.

Jumlah Pemesan:	1 user
Jumlah Member:	3 user
Jumlah Terblokir:	0 user
Jumlah Prospek:	2 user
Komisi Anda:	Rp. 150.000,-

Jumlah kunjungan ke website anda:  
10 kali.

**Strategi Promosi Program Reseller  
jadangbisnis.localhost**

Setiap member secara otomatis juga menjadi reseller program jadangbisnis.localhost, jadi berkesempatan untuk mendapatkan komisi 50% atau Rp. 50.000,- dari setiap member yang masuk lewat rekomendasi anda.

Sekedar mengingatkan URL yang anda promosikan adalah:  
<http://jadangbisnis.localhost/res=joni>

Semakin banyak user yang mengunjungi URL anda semakin besar pula kesempatan anda untuk memperoleh komisi. Namun untuk dapat menghasilkan banyak pengunjung. Diperlukan cara-cara yang efektif dalam promosi. Pada halaman inilah anda akan menemukan strategi promosi yang efektif.

DAN SETERUSNYA... DAN SETERUSNYA...

Gambar 5.25: Halaman strategi promosi untuk member

Sekarang semua file untuk member telah jadi, untuk itu anda dapat login ke member area. Untuk username gunakan saja *joni* dan untuk password *pas123*.

Akhir dari file yang berada pada direktori **reseller\member**.

## Kalender



## Berikutnya

Pada bab yang terakhir yaitu bab 6, kita akan membahas seputar mengupload file ke server dan konfigurasi website pada control panel.

---

# Bab 6

- **Upload File**
- **Konfigurasi Website**
- **Domain dan Web Hosting**

Sebelum melanjutkan ke materi bab 6 ikuti langkah-langkah berikut:

- Masuk pada direktori C:\Apache2\htdocs\webmaster
- Buat folder baru, beri nama folder tersebut **Bab6**

\*\*\*\*\*

## Hari Ke-30

Lakukan konfigurasi direktori berikut:

- Masuk pada direktori C:\Apache2\htdocs\webmaster\Bab6
- Buat folder baru, beri nama folder tersebut **hari-30**

# Upload File

Tidak terasa kita sudah berada pada hari yang ketiga puluh. Pada hari yang terakhir ini kita akan membahas bagaimana mengupload file ke server dan mengkonfigurasi website kita pada server. Diharapkan setelah membaca materi ini anda akan paham tentang:

- Pengertian upload
- Software FTP
- Memilih web hosting gratisan
- Trik menyiasati nama domain

File-file yang akan kita upload adalah file proyek terakhir kita yaitu website reseller yang telah kita buat pada bab 5.

## Pengertian Upload

Mungkin sebagian besar dari anda belum pernah mendengar kata upload. Karena yang sering kita dengar adalah *download* bukan *upload*. Jika download merupakan proses pengiriman file dari server ke komputer kita. Maka upload merupakan kebalikan dari proses tersebut.

Secara sederhana *upload merupakan proses pengiriman file dari komputer kita ke komputer server*. Server penyedia ruang atau hardisk di internet disebut **web hosting**. Agar website kita dapat diakses oleh semua orang di seluruh dunia. Maka kita perlu mengupload file kita ke server web hosting.

Untuk mengupload file-file kita ke komputer server dapat dilakukan dengan berbagai cara. Salah satu diantaranya adalah menggunakan FTP yaitu **File Transfer Protocol**. Koneksi FTP menjadi standar di internet untuk proses pengiriman file. Selain FTP, kita juga dapat menggunakan koneksi HTTP (Hyper Text Transfer Protocol) yang berbasis web.

Ada beberapa istilah yang perlu anda ketahui dalam proses upload file. Diantaranya *Local Computer* dan *Remote Computer*.

Local Computer merupakan istilah untuk komputer anda. Sedangkan Remote Computer merupakan komputer server. Jadi jika ada istilah local file berarti file-file tersebut adalah file pada komputer anda, dan remote file merupakan file-file yang ada pada komputer server.

## Software FTP

Jika anda mengupload file menggunakan HTTP anda tidak memerlukan software FTP untuk mengupload file. Ini dikarenakan hampir semua web hosting menyediakan form upload file di member area. Yang perlu anda lakukan hanyalah mengklik tombol browse... untuk memilih file yang akan diupload.

Dalam bahasan kali ini kita hanya membahas bagaimana mengupload file menggunakan koneksi FTP. Untuk itu kita memerlukan Software FTP.

Di internet banyak sekali terdapat software FTP baik yang gratis atau yang berbayar. FTP software yang saya gunakan untuk contoh ini adalah **AceFTP3 Freeware**. Dari hasil pengujian yang saya lakukan kecepatan upload AceFTP lebih baik dari software lainnya.

Anda dapat mendownload AceFTP3 di alamat  
<http://freeware.aceftp.com/>

Meskipun anda menggunakan software FTP lain itu tidak masalah. Karena sebagian besar user interface software-software FTP hampir sama. Begitu juga dengan cara penggunaannya saya rasa tidak akan jauh berbeda satu sama lain.

Untuk software FTP lainnya anda dapat mencarinya lewat google. Ketikkan saja *FTP Software* atau *Free FTP Software*. Mungkin anda akan bingung memilih yang mana karena hasil pencarian mencapai ribuan link.

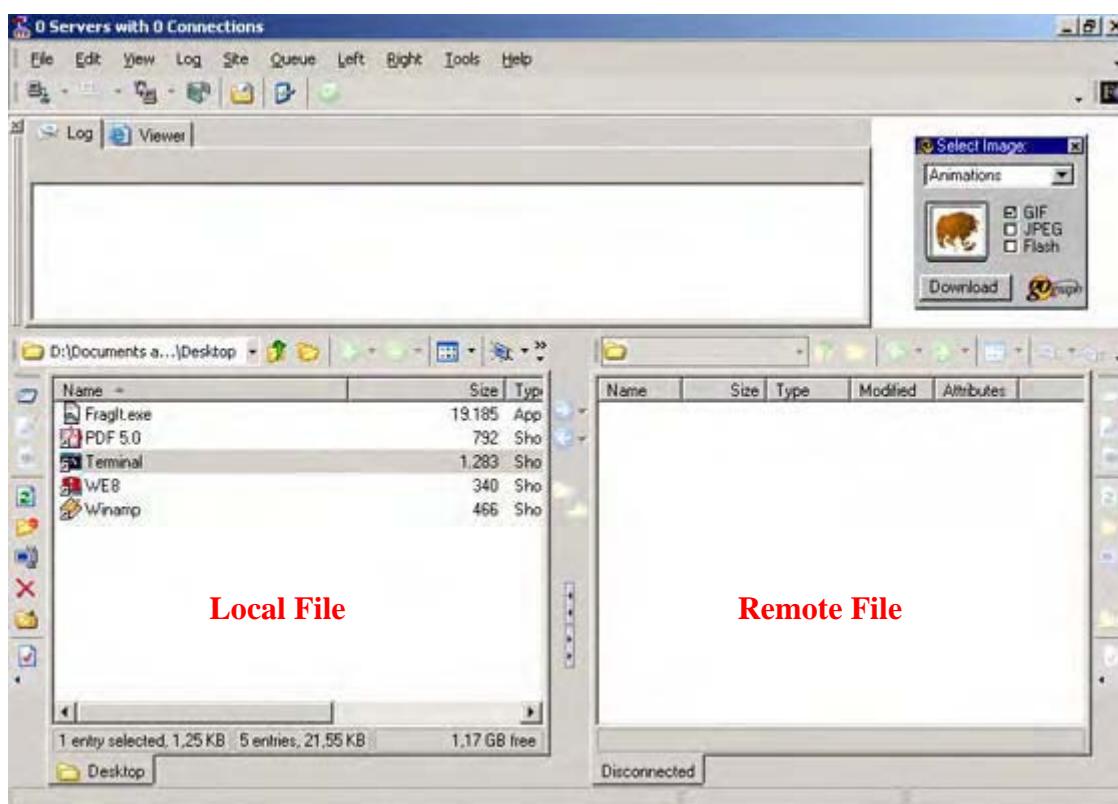
### **Menggunakan AceFTP3Freeware**

Setelah anda mendownload file instalasinya klik dua kali untuk memulai proses instalasi. Untuk menjalankannya anda dapat mengklik menu Start > Program > AceFTP Freeware > AceFTP Freeware.

Karena kita baru satu kali menjalankan AceFTP maka muncul window Registration. Klik saja tombol **Skip** untuk melanjutkan.

AceFTPFreeware hanya dapat digunakan selama 30 hari saja jika anda tidak melakukan registrasi ke <http://freeware.aceftp.com/register/>. Proses registrasi ini tidak memerlukan biaya. Setelah anda melakukan registrasi anda akan dikirim email yang berisi Kode registrasi untuk AceFTP3Freeware.

Untuk menghilangkan masa trial, klik menu Help > Register ... lalu masukkan alamat email yang anda gunakan untuk registrasi dan masukkan juga kode registrasinya. Setelah itu klik OK.



Gambar 6.0: Tampilan AceFTP3Freeware

Seperti software FTP lainnya, untuk mengupload atau mendownload file kita dapat melakukannya dengan ***drag n drop*** dari jendela local file ke remote file atau sebaliknya. Kita juga dapat melakukannya dengan metode copy-paste.

## Memilih Web Hosting Gratisan

Karena tujuan kita hanya untuk pembelajaran. Kita tidak perlu menyewa web hosting yang berbayar. Cukup dengan web hosting gratisan anda dapat merasakan bagaimana rasanya memiliki website yang dapat dilihat oleh user lain di seluruh dunia.

Dalam memilih web hosting gratisan dapat dikatakan *susah-susah gampang*. Karena banyak banyak sekali web hosting gratisan yang ditawarkan di internet. Namun hanya segelintir yang mempunyai kualitas setara dengan web hosting komersial. Coba pergi ke google ketikkan keyword *Free Web Hosting*, dijamin hasilnya bakal membuat pusing kepala anda.

Kebanyakan web hosting gratisan tidak menyertakan PHP dan MySQL dalam layanannya. Misalnya web hosting gratisan milik yahoo yaitu *Geocities*. Geocities hanya dapat memproses halaman HTML saja dan tidak dapat memproses PHP.

Setelah surfing berkali-kali mencari web hosting gratisan yang bagus dan support PHP dan MySQL. Akhirnya saya menemukan web hosting gratisan yang cukup bagus yaitu <http://100webspace.com>.

Meskipun gratis 100webspace menawarkan kapasitas penyimpanan yang cukup besar yaitu **100MB**. Kapasitas sebesar itu sudah sangat mencukupi untuk website skala menengah. Disamping pelayanan yang diberikan sangat bagus, walau account kita gratis namun mereka masih menyediakan fasilitas help dan kontak 24/7 ke bagian support mereka.

## Registrasi 100webspace.com

Untuk mencicipi layanan ini konek ke internet arahkan browser anda ke <http://100webspace.com/>.

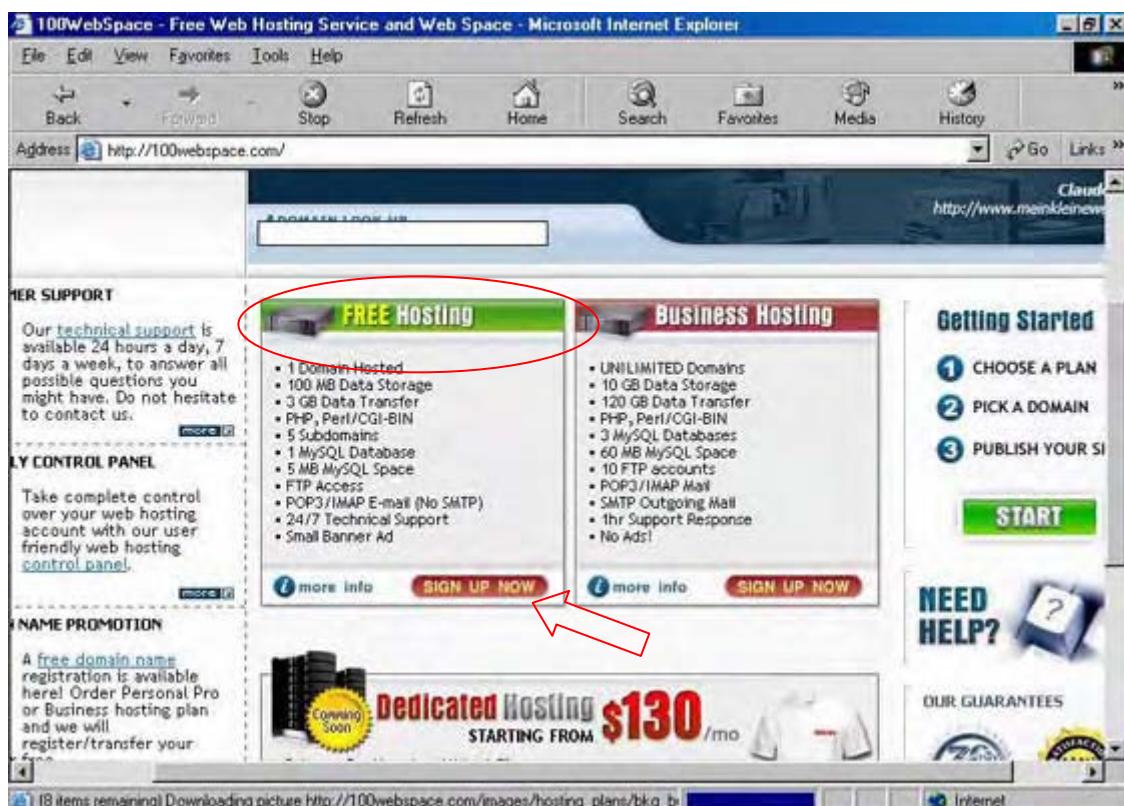
1. Pilih **Free Hosting** lalu klik **Sign Up Now**. (Gambar 6.1)
2. Selanjutnya muncul beberapa pilihan domain. Pilih **Use A FREE SUB-DOMAIN** lalu klik Next Step. (Gambar 6.2)
3. Setelah anda mengklik Next Step akan muncul form isian. Isi form tersebut dan pada bagian bawah form pilih **Free Hosting** dan **Use sub domain**. (Gambar 6.3)
4. Akan muncuk iklan penawaran untuk menghilangkan banner dan pop-up. Klik saja No, thanks, continue with my ad-supported account. (Gambar 6.4)
5. Selanjutnya data anda akan diproses. Jika berhasil maka anda akan dibawa ke halaman konfirmasi. (Gambar 6.4)
6. Password anda untuk masuk ke member area dan FTP dikirim ke email anda. Cek email tersebut.

Saat tulisan ini dibuat 100webspace menggunakan sub domain hollosite.com. Sebelumnya 100webspace menggunakan farvista.net. Jadi mungkin saat anda mendaftar, sub domian yang diberikan bukan hollosite.com melainkan lainnya. Oleh karena itu pada bahasan selanjutnya sesuaikan dengan nama subdomain anda sendiri.

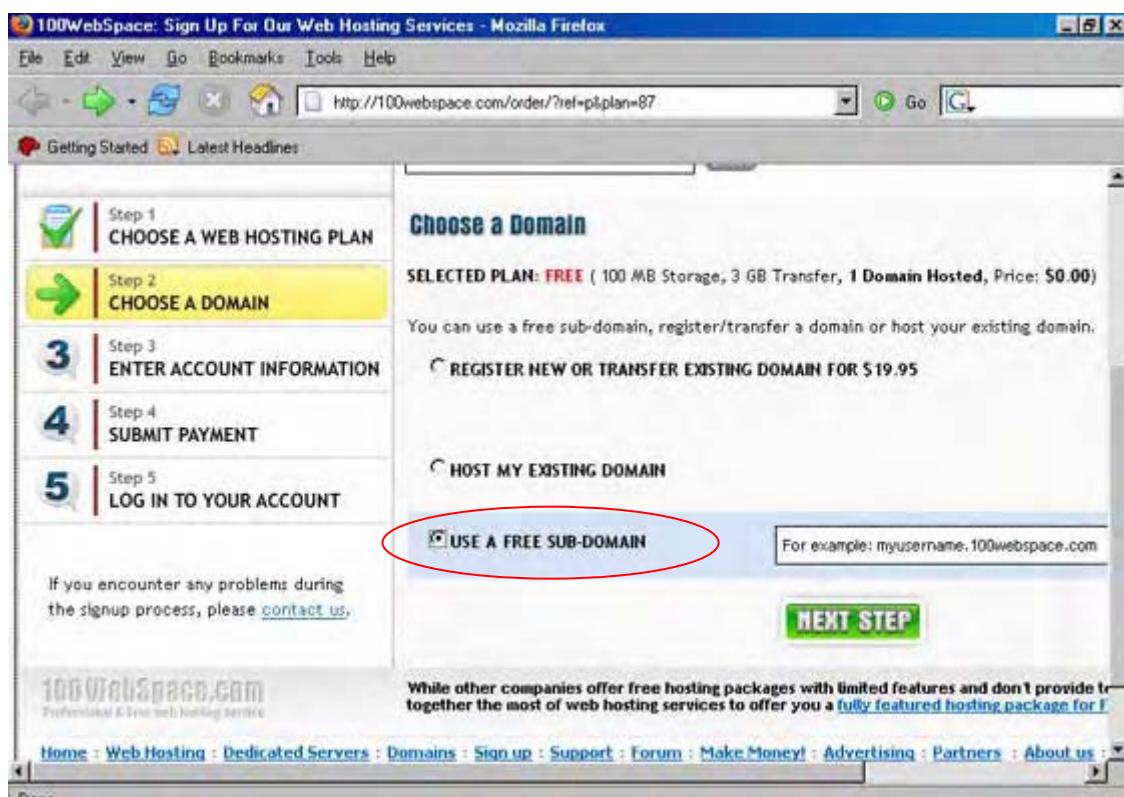
Sebagai contoh nama subdomain yang diberikan kepada saya adalah <http://rioast8.hollosite.com>.

rioast8 merupakan username saya untuk masuk ke member area 100webspace. Username dibuat sendiri oleh 100webspace secara otomatis dan bukan keinginan kita.

Karena kita memilih paket free hosting sudah tentu nantinya website kita akan dipasangi iklan oleh 100webspace.com. Tapi tidak apa-apa, yang penting website kita bisa jalan.



Gambar 6.1: Pilihan Free Hosting di 100webspace.com



Gambar 6.2: Memilih sub domain

Fax (optional):

**CHOOSE YOUR WEB HOSTING PLAN:**

<input checked="" type="radio"/> FREE Hosting	100 MB Disk Space, 3,000 MB Traffic, 1 Domain
<input type="radio"/> FREE Hosting without banners	100 MB Disk Space, 3,000 MB Traffic, 1 Domain
<input type="radio"/> Personal package	500 MB Disk Space, 10,000 MB Traffic, 1 Domain
<input type="radio"/> Personal Pro	5,000 MB Disk Space, 80,000 MB Traffic, 10 Domains
<input type="radio"/> Business package	10,000 MB Disk Space, 120,000 MB Traffic, 9999 Domains

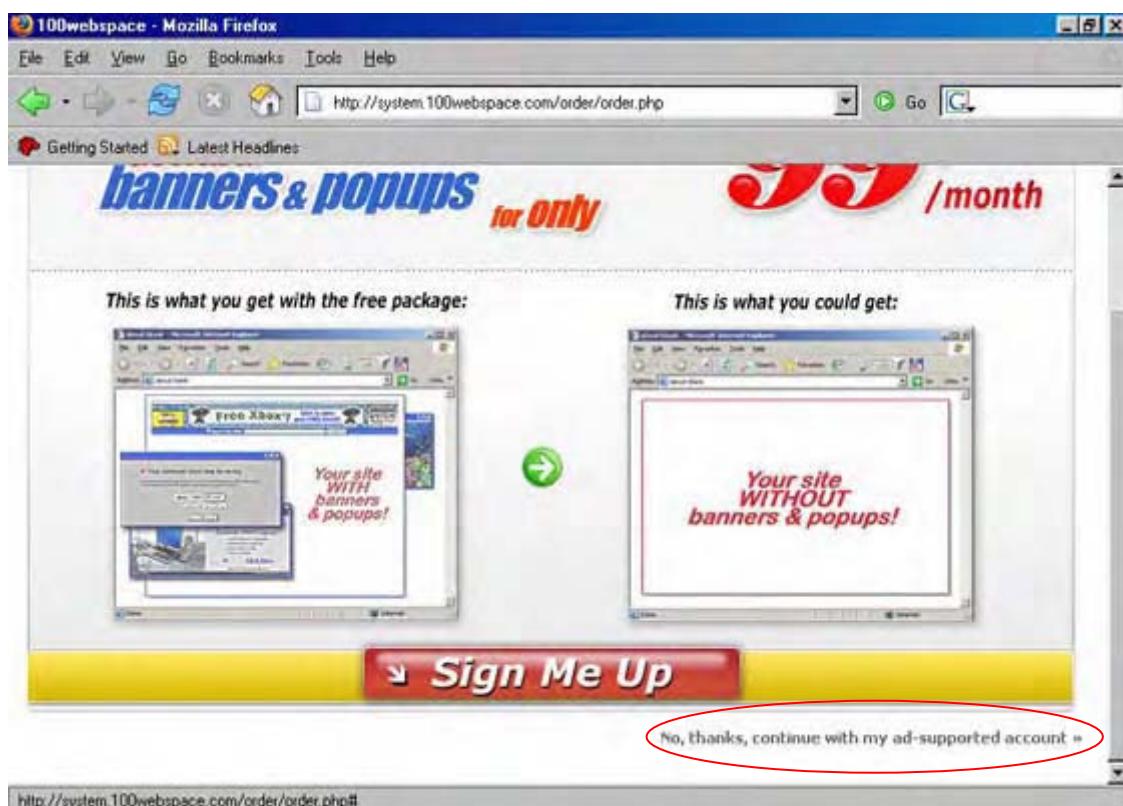
Payment Cycle:  Annually \$ 0.00 (\$ 0.00 per month)

**PICK UP A DOMAIN NAME:**

<input type="radio"/> Register new domain	For example: myusername.100webspace.com
<input type="radio"/> Transfer domain name	
<input type="radio"/> Use my existing domain	
<input checked="" type="radio"/> Use a subdomain	

Total amount to be charged: **\$ 0.00 (USD)**

Gambar 6.3: Form Registrasi dan pilihan Hosting



Gambar 6.4: Penawaran untuk menghilangkan banner dan pop-up



Gambar 6.5: Proses registrasi selesai

## Trik Menyiasati nama domain

Di internet semua penyedia layanan gratis baik itu email atau web hosting pasti menambahkan subdomain pada account setiap membernya. Hal itu wajar karena mereka memberikan suatu layanan gratis dan sebagai timbal baliknya sub domain mereka ditambahkan pada account anda.

Contoh yang sering kita lihat adalah layanan email dari yahoo!. Ketika anda mendaftar untuk menjadi member dari yahoo! dengan pilihan paket gratis. Maka email anda akan berformat usernameanda@yahoo.com.

Begitu juga dengan web hosting. Pada bahasan sebelumnya kita menggunakan web hosting gratisan milik 100webspace.com. Setelah kita melakukan registrasi maka alamat website kita nantinya adalah username\_anda.hollosite.com.

Alamat domain seperti diatas terasa kurang ***bonafid***. Ada beberapa cara agar domain anda kelihatan lebih ***profesional*** meskipun *gratisan*.

Agar domain anda yang semula id\_anda.hollosite.com dapat dipendekkan maka kita dapat menggunakan layanan URL Redirection Service. URL Redirection service merupakan layanan yang bertujuan untuk memendekkan URL. Di internet banyak tersedia layanan URL redirection service dan layannya bersifat gratis.

Prinsip kerja dari layanan URL Redirection Service adalah melakukan redirect halaman. Ketika seseorang mengunjungi website URL Redirection anda maka secara otomatis ia akan diarahkan ke website asli anda.

Berikut ini adalah daftar penyedia layanan URL Redirection Service yang sering saya gunakan sewaktu belum memiliki domain berbayar.

### DOT TK ([www.dot.tk](http://www.dot.tk))

---

Situs ini menyediakan layanan URL redirection service dan juga email. Format website anda nantinya adalah www.domain\_anda.tk. Ketika seseorang mengunjung www.domain\_anda.tk maka secara otomatis mereka akan diarahkan ke website asli anda misalnya id\_anda.hollosite.com.

### .CO.NR ([www.freedomain.co.nr](http://www.freedomain.co.nr))

Menurut saya daripada URL www.domain\_anda.tk alamat URL anda akan tampak lebih professional dengan nama www.domain\_anda.co.nr. Karena banyak domain komersial memiliki akhiran .co.xx misalnya .co.id, .co.uk, .co.au dan sebagainya. Itu menurut saya entah menurut anda.

NB: Pada saat registrasi di .co.nr pastikan anda pada pilihan **Kill Frame** anda memilih **No**. Pada pilihan **Cloaking** pilih **On**. Anda juga dapat merubahnya di member area .co.nr.

### CJB.NET ([www.cjb.net](http://www.cjb.net))

Layanan ini merupakan layanan yang saya gunakan pertama kali. Namun setelah mengetahui dot.tk dan co.nr saya tinggalkan layanan dari cjb.net.

Untuk layanan URL Redirection Service lainnya anda dapat mencarinya lewat google. Ketik saja *URL Redirection Service*.

Untuk tutorial bagaimana membuat URL redirection service anda dapat membacanya di member area rahasia-webmaster.com.

### Kekurangan URL Redirection Service

Sebenarnya layanan ini hanya cocok untuk digunakan untuk promosi saja. Seperti mereka yang mengikuti program reseller dan sejenisnya. Karena pada program reseller mereka hanya menuju satu halaman saja.

Mengapa hanya cocok untuk promosi? Karena setelah user mengunjung website Redirection anda misalnya www.domain\_anda.tk. Mereka akan diarahkan ke website asli anda misalnya id\_anda.hollosite.com. Kalau website anda hanya terdiri dari satu halaman itu tidak masalah. Tapi jika website anda memiliki banyak halaman, maka alamat website "gadungan" anda akan kembali muncul di browser.

Itu dikarenakan setelah user mengklik link menuju halaman lainnya. Domain yang ada pada address bar di browser sudah tidak lagi www.domain\_anda.tk. Melainkan [www.id\\_anda.hollosite.com/halaman.php](http://www.id_anda.hollosite.com/halaman.php). Nah..., ketahuan deh alamat asli website anda.

### Menyembunyikan URL Asli dengan Frame

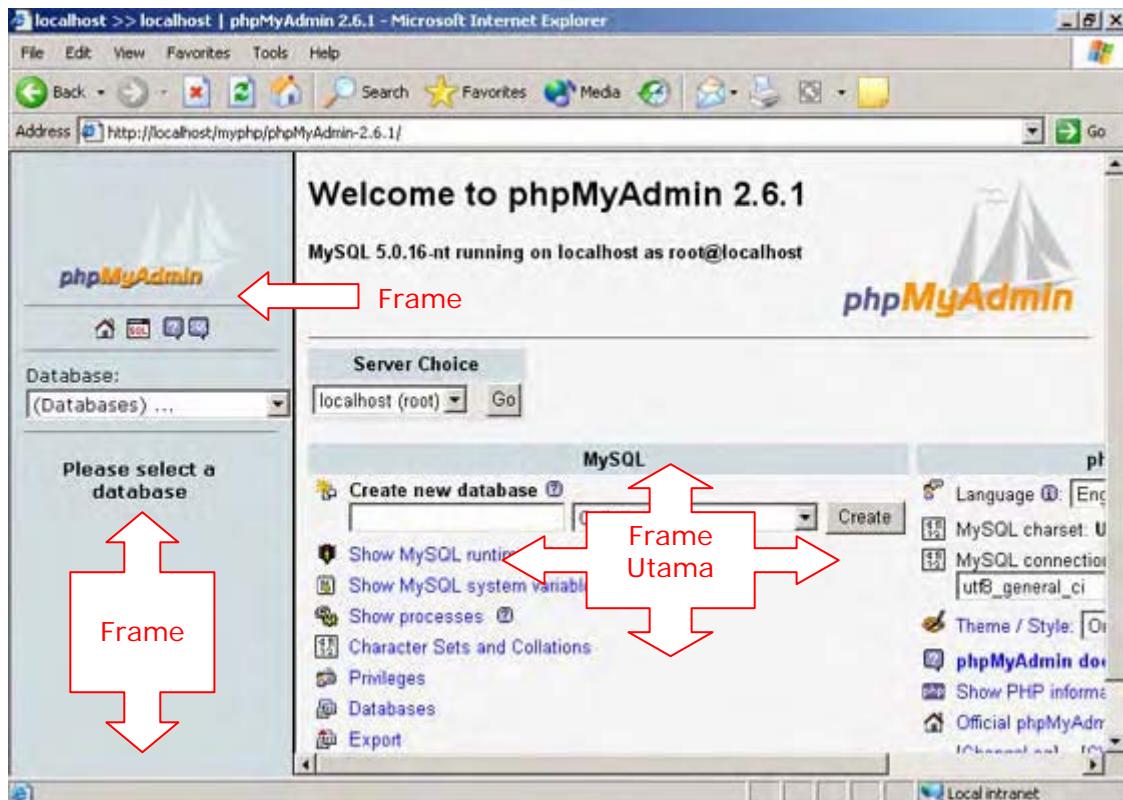
Cara yang umum digunakan agar URL di address bar browser anda tetap tidak berubah adalah menggunakan **frame**(bingkai). Frame bukan termasuk bagian dari PHP tetapi hanya berupa tag HTML saja.

Salah satu contoh penggunaan frame adalah pada phpMyAdmin. Coba arahkan browser anda ke <http://localhost/phpMyAdmin-2.6.x/> (sesuaikan dengan versi anda). Coba klik link menuju halaman lainnya misalnya klik link SQL.

Lihat pada address bar di browser, bagaimana URLnya Selalu tetap bukan?. Itulah yang akan kita terapkan ketika menggunakan web hosting gratisan. Bedanya pada phpMyAdmin menggunakan tiga frame sedangkan kita nanti hanya akan menggunakan satu frame saja.

Pada bagian kiri terdapat dua frame yaitu bagian frame untuk logo dan icon dan bagian bawah adalah nama database dan nama-nama tabel. Pada sebelah kanan hanya terdapat satu frame dan frame ini merupakan frame utama. Lihat gambar 6.6

Selain itu frame juga dapat meminimalisir terjadinya Cross Site Scripting karena variabel tidak tampak di address bar browser.



Gambar 6.6: Penggunaan frame pada phpMyAdmin

Prinsip kerja dari frame mirip dengan tag table. Dimana di tag tersebut terdapat baris dan kolom.

Karena orientasi kita pada website ladangbisnis.localhost. Maka frame yang kita buat hanya satu frame saja.

### Penerapan Teori Frame

Untuk penerapan teori frame ini kita akan mengedit website ladangbisnis.localhost. Pengeditan yang kita lakukan tidak banyak. Karena kita hanya membuat file baru dan mengedit file utama index.php saja.

- Buka PHP DESIGNER 2006 klik **File > Open**
- arahkan ke C:\Apache2\htdocs\webmaster\Bab5\hari-26-27-28-29\reseller\
- Pilih file *index.php*.
- Simpan dengan nama lain klik **File > Save As** isikan ***utama.php*** lalu klik Save

Sekarang kita akan mengedit kembali file **index.php** klik **File > Open** pilih file index.php. Edit file tersebut sehingga menjadi seperti berikut,

Script dari file **index.php** setelah diedit

```
<?php

/*****
/* NamaFile: index.php
/* Penulis : Rio Astamal */
**/
**/
```

---

```

/* Tanggal : 25-10-2006 */

/* Fungsi : untuk menampilkan Frame utama ladangbisnis.localhost */

// panggil file var_utama.php

include ('include/var_utama.php');

?

<html>

<head>

<title>Jadikan Internet Ladang Bisnis Paling Menguntungkan</title>

<!-- OPTIMASI META TAG -->

<meta name="keywords" content="peluang bisnis, peluang usaha, lowongan kerja,

internet marketing, mlm, bisnis, web programmer, programmer">

<meta name="description" content="Temukan cara paling ampuh untuk mengubah

internet menjadi ladang bisnis paling menguntungkan">

<meta name="title" content="ladangbisnis.localhost">

<meta name="author" content="Rio Astamal">

<meta name="publisher" content="ladangbisnis.localhost">

<meta name="owner" content="ladangbisnis.localhost">

<meta name="distribution" content="global">

<meta name="robots" content="index, follow">

<meta name="revisit-after" content="30 days">

<!-- AKHIR META TAG -->

<!-- PANGGIL FILE style.css -->

<link rel="stylesheet" type="text/css" href="template/style.css">

</head>

<!-- FRAME -->

<frameset framespacing="0" frameborder="0" border="0" rows="*" cols="*">

<frame name="utama" scrolling="auto" src="utama.php?res=<?php echo $res; ?>">

<noframes>

<body>

Website ini menggunakan Frame tetapi browser anda tidak

men-supportnya.</p>

</body>

</noframes>

</frame>

</frameset>

</html>


```

---

Simpan kembali dengan mengklik **File > Save**. Sekarang coba klik halaman lainnya, lihat URL pada address bar di browser. URL tidak berubah bukan? Meskipun anda mengklik halaman lainnya URL akan tetap <http://localhost/webmaster/hari-26-27-28-29/reseller/>.

### PENJELASAN SCRIPT

Pada script diatas kita kita memanggil file var\_utama.php. Tujuan utama pemanggilan file ini hanya untuk mendapatkan variabel \$res yang ada pada file tersebut.

Untuk membuat frame seperti halnya table kita harus membuat bingkai utamanya terlebih dahulu. Untuk itu kita gunakan tag frameset. Tag frame frameset memiliki beberapa attribut diantaranya border, framespacing, frameborder, noresize, cols, dan rows.

Attribut framespacing menentukan digunakan untuk menentukan jarak spasi antar frame. Frame border seperti namanya menentukan tebal bingkai dari frame. Jika anda ingin frame

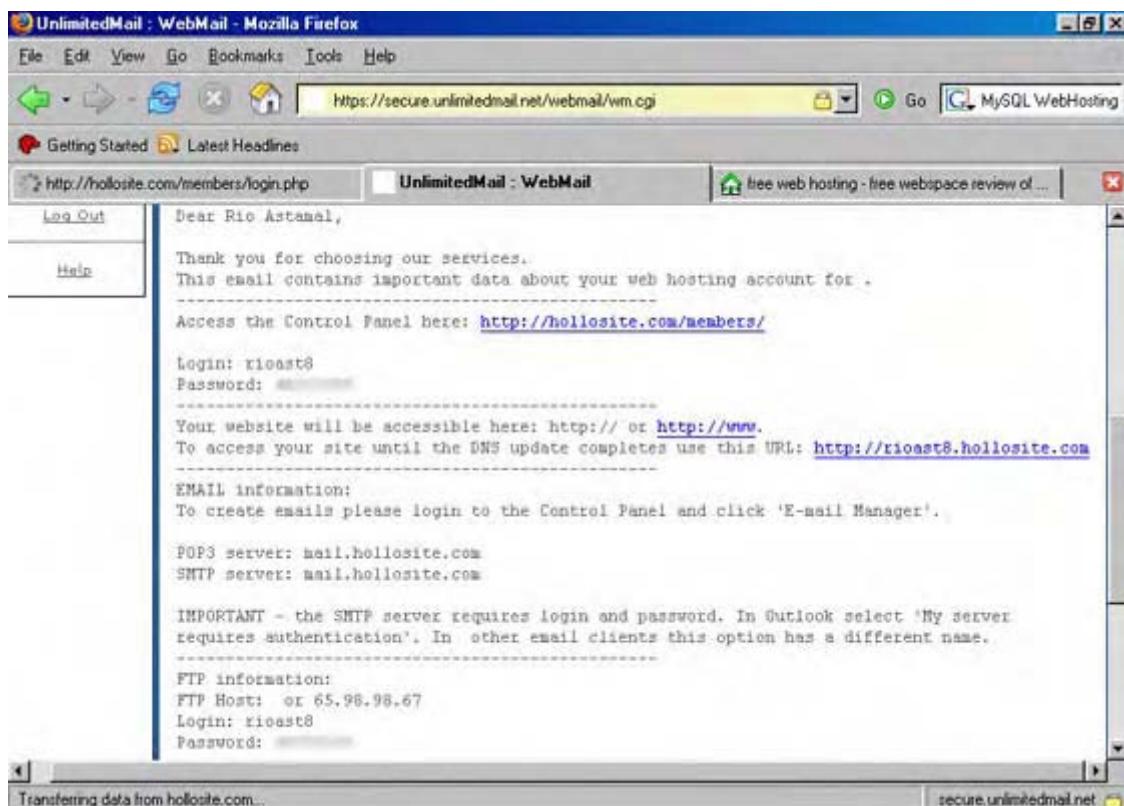
---

tersebut tidak bisa diubah ukurannya maka masukkan juga attribut noresize. Cols dan rows masing-masing untuk menentukan jumlah kolom dan baris dari frame.

Isi dari attribut rows dan cols kita isi dengan "\*". Ini menunjukkan bahwa lebar kolom dan baris yang ada diframe mengikuti lebar dari halaman yang di dalamnya.

# Konfigurasi Website

Setelah proses registrasi selesai, saatnya kita untuk mengupload file-file kita. Namun sebelumnya kita perlu melakukan setting pada account kita di 100webspace.com. Email yang anda terima berisi informasi penting seperti password untuk member area dan FTP account. Lihat gambar 6.7



Gambar 6.7: Email konfirmasi dari 100webspace.com

Pada email diatas alamat untuk mengakses control panel adalah di <http://hollosite.com/member/>

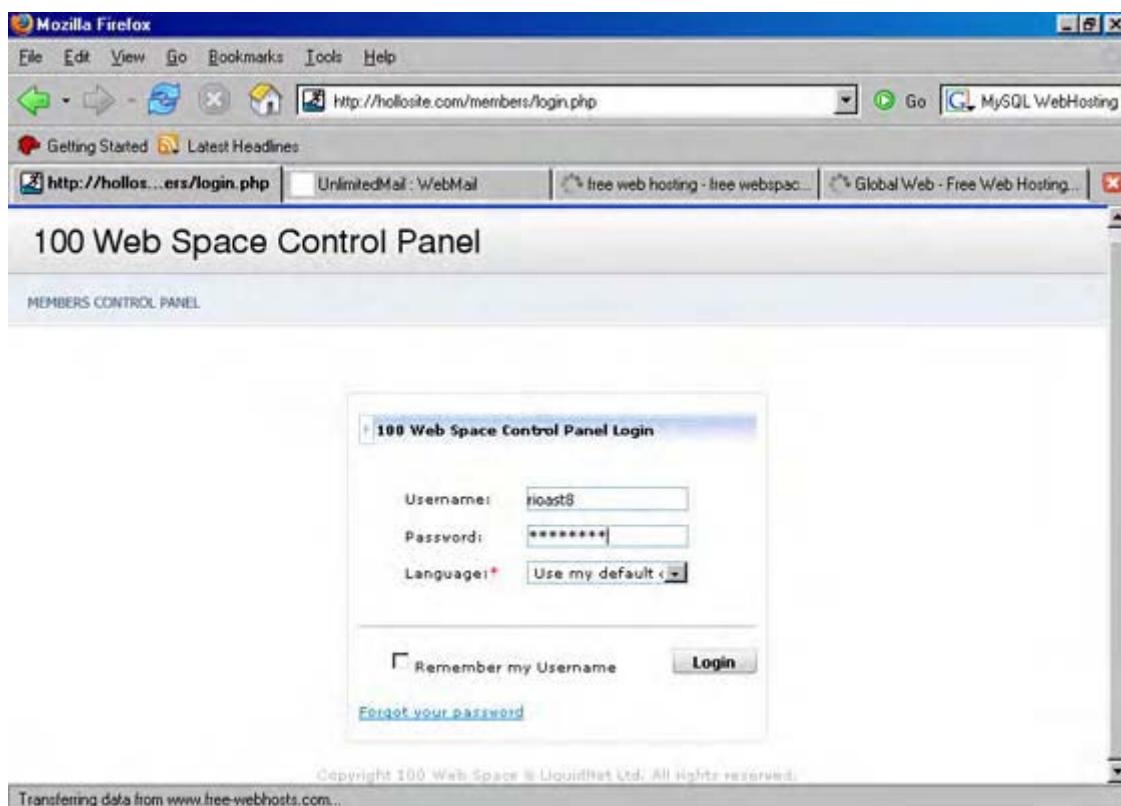
Dan untuk website dapat diakses  
Di <http://rioast8.hollosite.com/>

Nama domainnya kok kurang keren gitu?!, namanya juga gratisan pasti dibubuh sub domain. Tapi tenang nanti saya tunjukkan **trik** bagaimana agar URL yang "jelek" itu dapat anda ganti sesuai dengan keinginan anda.

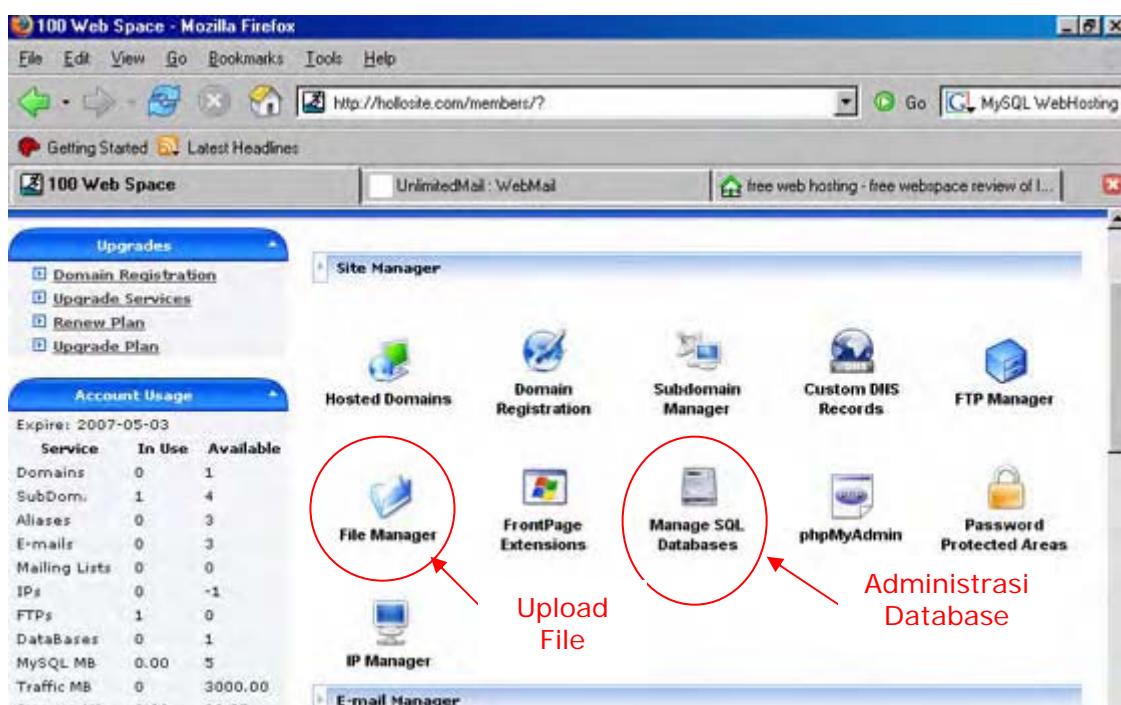
Untuk selanjutnya login ke control panel arahkan browser anda ke <http://hollosite.com/member/>. Isilah username dan password sesuai dengan yang anda terima pada email konfirmasi.

Halaman login ditunjukkan oleh gambar 6.8. Setelah login saya sarankan agar anda mengganti password default yang diberikan. Karena password tersebut susah diingat. Tapi ingat juga prinsip membuat password. Gunakan kolaborasi antara angka dan huruf agar password anda sulit untuk ditebak.

Setelah anda login anda dapat melakukan konfigurasi untuk website anda. 100webspace.com menyediakan control panel yang lengkap layaknya web hosting komersial. Lihat gambar 6.9.



Gambar 6.8 Halaman login untuk masuk control panel 100webspace.com



Gambar 6.9: Control panel website pada 100webspace.com

Ada beberapa menu yang penting untuk anda ketahui. Diantaranya SubDomain Manager, File Manager dan Manage SQL Database.

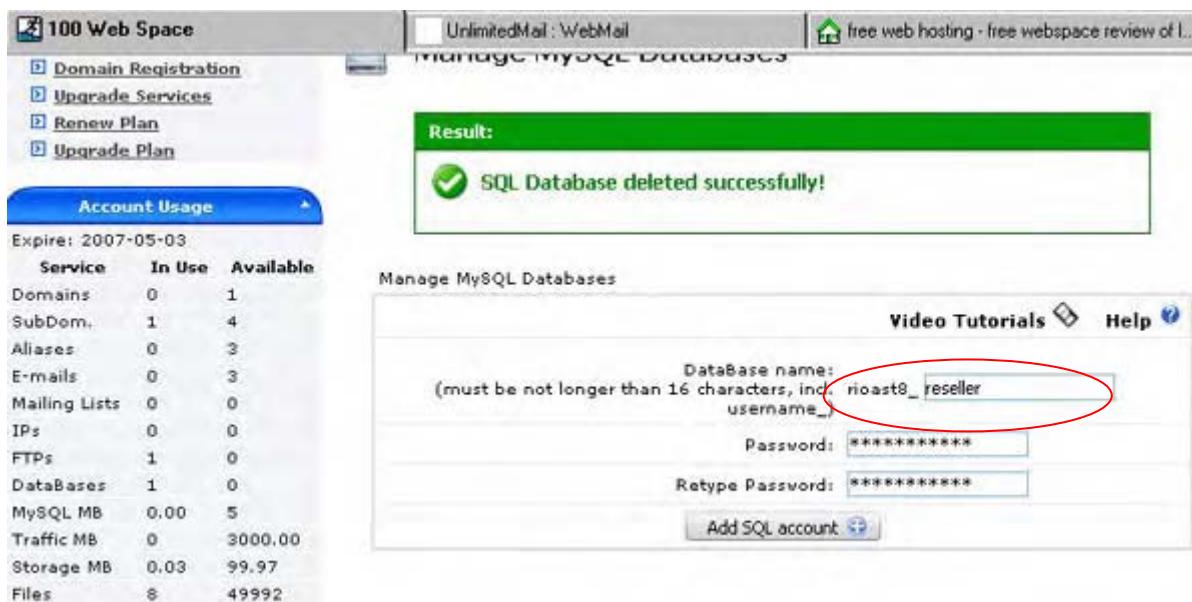
Pada menu SubDomain Manager anda dapat menambah domain baru. Maksimal domain yang dapat anda buat adalah 5 domain termasuk domain awal yaitu id\_anda.hollosite.com.

Halaman File manager berfungsi untuk menyediakan form upload. Anda dapat menggunakan fasilitas ini bilamana anda tidak mempunyai Software FTP.

## Membuat Database

Setelah login kita dapat membuat database untuk file-file ladangbisnis.localhost yang akan kita upload. Untuk itu klik **Manage SQL Databases**. Untuk nama database akan berbeda dengan yang kita buat pada komputer lokal.

Pada web hosting baik yang gratisan ataupun yang komersial. Nama database kita biasanya akan ditambahkan username kita. Hal ini bertujuan agar tidak ada database yang sama dalam satu web hosting.



Gambar 6.9: Membuat database baru

Untuk nama database isiakan **reseller**. Dan untuk password isiakan **reseller123**. Setelah itu klik tombol **Add SQL account**. Nantinya nama database anda bukan reseller melainkan **usernameanda\_reseller**. Jadi kita perlu mengedit file SQL yang ada pada proyek sebelumnya. Kapasitas maksimal yang disediakan untuk penyimpanan database adalah sebesar 5 MB.

Untuk username dari database sama dengan nama database yaitu `usernameanda_reseller`. Maksimal panjang nama database adalah 16 karakter.

100webspace.com secara otomatis akan mengirimkan email setelah database baru dibuat. Email tersebut berisi username, password, nama database, nama host, dan port yang digunakan. Jadi cek email anda untuk memastikan konfigurasi yang tepat.

Untuk menuju phpMyAdmin anda dapat mengklik nama database yang baru anda buat. Atau anda dapat mengklik link phpMyAdmin pada halaman utama Control Panel.

Namun sebelum itu ada hal-hal yang perlu anda perhatikan. Versi MySQL yang digunakan 100webspace.com adalah MySQL 4.0.24, sedangkan milik kita adalah MySQL 5.0.16. Dari hasil percobaan yang saya lakukan ada beberapa query yang tidak bisa jalan.

Query yang tidak bisa dijalankan adalah query yang melibatkan penggunaan jenis tabel yang digunakan (MyISAM atau InnoDB). Query berikut menyebabkan error pada phpMyAdmin di 100webspace.com

```
CREATE TABLE tbl_admin
```

```
(
 ...
 ...
) TYPE='MyISAM';
```

Ketika query yang diatas saya jalankan ternyata error. Saya sempat bingung mengapa terjadi error. Setelah saya hilangkan query TYPE='MyISAM' ternyata query berhasil dilakukan.

Satu hal lagi yang perlu anda ingat pada phpMyAdmin di web hosting. Anda tidak dapat melakukan query DROP DATABASE dan CREATE DATABASE karena privilege kita dibatasi oleh administrator. Jadi pada file script kita nantinya baris query untuk DROP DATABASE dan CREATE DATABASE akan kita hapus.

Untuk selanjutnya kita akan mengedit file db\_reseller.sql dan fungsi.php karena ke dua file ini berhubungan dengan database. Untuk yang pertama kita akan mengedit db\_reseller.sql. Jalankan PHP DESIGNER 2006 atau Notepad. Buka file db\_reseller.sql yang ada pada C:\Apache2\htdocs\webmaster\Bab5\hari-26-27-28-29\reseller

Script dari file db\_reseller.sql setelah diedit

---

```
-- hilangkan query DROP DATABASE dan CREATE DATABASE
-- hilangkan semua query TYPE='MyISAM'
-- hilangkan query GRANT

-- aktifkan database reseller
-- sesuaikan dengan nama database anda
-- usernameanda_reseller
USE rioast8_reseller;

-- buat tabel tbl_admin
CREATE TABLE tbl_admin
(
 username varchar(16) PRIMARY KEY,
 password varchar(32)
) ;

-- buat tabel tbl_blokir
CREATE TABLE tbl_blokir
(
 tgl varchar(10),
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 nama varchar(50),
 email varchar(30),
 kota varchar(20),
 reseller varchar(16),
 transfer int(10)
) ;

-- buat tabel tbl_member
CREATE TABLE tbl_member
(
 tgl varchar(10),
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 nama varchar(50),
 email varchar(30),
 kota varchar(20),
 reseller varchar(16),
 transfer int(10)
) ;

-- buat tabel tbl_pemesan
```

```

CREATE TABLE tbl_pemesan
(
 tgl varchar(10),
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 nama varchar(50),
 email varchar(30),
 kota varchar(20),
 reseller varchar(16),
 transfer int(10)
);

-- buat tabel tbl_profil
CREATE TABLE tbl_profil
(
 username varchar(16) PRIMARY KEY,
 password varchar(32),
 nama varchar(50),
 email varchar(30),
 kota varchar(20),
 reseller varchar(16),
 alamat varchar(100),
 telpon varchar(20),
 bank varchar(100)
);

-- buat tabel tbl_prospek
CREATE TABLE tbl_prospek
(
 tgl_masuk varchar(10),
 email varchar(30) PRIMARY KEY,
 nama varchar(50),
 reseller varchar(16)
);

-- buat tabel tbl_testimonial
CREATE TABLE tbl_testimonial
(
 tm_id int(4) PRIMARY KEY auto_increment,
 username varchar(16),
 testimonial text
);

-- buat tabel tbl_traffic
CREATE TABLE tbl_traffic
(
 username varchar(16) PRIMARY KEY,
 traffic int(6)
);

CREATE TABLE tbl_transfer
(
 nilai int(6) PRIMARY KEY
);

-- masukkan account admin ke tbl_admin
-- password => _super_pass_ = 1835a30bcdef3c4fa4fca7b6d9092113
-- setelah dienkrip dengan balik_md5()
INSERT INTO tbl_admin VALUES
('`_super_user_`', '1835a30bcdef3c4fa4fca7b6d9092113');

-- masukkan beberapa data ke tbl_member, tbl_pemesan, dan tbl_profil
-- masukkan member (kita asumsikan passwordnya pas123)
-- pass123 = 0da656c1352c7ec167b91122dd6d1aff (setelah dienkrip balik_md5())

```

```

INSERT INTO tbl_member VALUES
('26-10-2005', 'joni', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Joni',
'joni@localhost.localdomain', 'Surabaya', 'admin', 50000),
('26-10-2005', 'jono', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Jono',
'jono@localhost.localdomain', 'Surabaya', 'joni', 50001),
('26-10-2005', 'joko', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Joko',
'joko@localhost.localdomain', 'Surabaya', 'joni', 50002);

-- masukkan beberapa data ke tbl_profil
INSERT INTO tbl_profil VALUES
('joni', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Joni',
'joni@localhost.localdomain', 'Surabaya', 'admin', 'Jl. Apache 80',
'080808080', 'Bank Localhost - Rek. 9876543210 - a.n Mr. Joni'),
('jono', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Jono',
'jono@localhost.localdomain', 'Surabaya', 'joni', 'Jl. Mail 21',
'070707070', 'Bank Localhost - Rek. 0123456789 - a.n Mr. Jono'),
('joko', '0da656c1352c7ec167b91122dd6d1aff', 'Mr. Joko',
'joko@localhost.localdomain', 'Surabaya', 'joni', 'Jl. Netbios 139',
'060606060', 'Bank Localhost - Rek. 5432167890 - a.n Mr. Joko');

-- masukkan ke tbl_traffic
INSERT INTO tbl_traffic VALUES
('admin', 15), ('joni', 10), ('jono', 5), ('joko', 4);

-- masukkan pemesan
INSERT INTO tbl_pemesan VALUES
('26-10-2005', 'jojo', 'pas123', 'Mr. Jojo', 'jojo@localhost.localdomain',
'Surabaya', 'jono', 50003),
('26-10-2005', 'josi', 'pas123', 'Mr. Josi', 'josi@localhost.localdomain',
'Surabaya', 'joko', 50004);

-- masukkan beberapa orang ke prospek
INSERT INTO tbl_prospek VALUES
('26-10-2005', 'gatot@kaca.com', 'Gatot Kaca', 'joni'),
('26-10-2005', 'semar@mendem.com', 'Semar Sakti', 'joni'),
('26-10-2005', 'os@linux.com', 'Mr. Open Source', 'joko');

-- set nilai transfer ke 50005 karena yang terakhir adalah 50004
-- yaitu si josи
INSERT INTO tbl_transfer VALUES (50005);

```

Simpan kembali dengan mengklik File > Save.

Selanjutnya file yang kita edit adalah *fungsi.php* yang berada pada direktori include. Pengeditan hanya kita lakukan pada bagian fungsi `konek_db()`. Karena kita sesuaikan dengan database dan username yang ada di web hosting kita.

Fungsi `konek_db()` pada file *fungsi.php* setelah diedit

```

// fungsi untuk mengkoneksikan ke MySQL server
// sesuaikan dengan username, database dan host anda di 100webspace.com
function konek_db()
{
 // untuk username = secure dan password = secure123
 @ $koneksi = mysql_connect('hollosite.com', 'rioast8_reseller',
 'reseller123');
 // jika gagal melakukan koneksi tampilkan pesan kesalahannya
 if (!$koneksi)
 {
 // jika kode anda sudah dirilis sebaiknya jangan menampilkan pesan
 // kesalahannya seperti berikut.
 // Gunakan metode ini hanya jika dalam tahap development saja.
 echo "Error: ".mysql_errno()."
\n";
 }
}

```

```

echo "Keterangan: ".mysql_error()."
\n";
exit;
}
else
{
 // pilih database yang digunakan
 mysql_select_db('rioast8_reseller');
 return true;
}
}

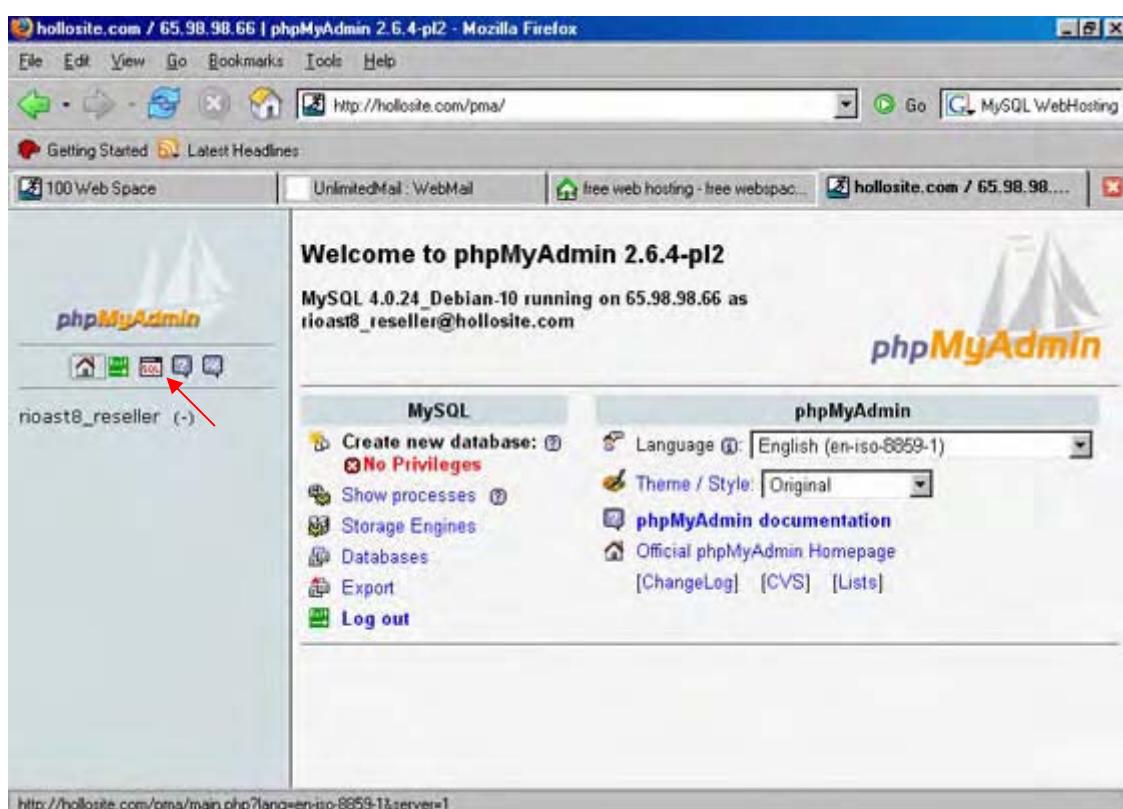
```

Simpan kembali file fungsi.php. Yang kita ubah hanya username dan nama database yang kita gunakan di webhosting. Untuk itu sesuaikan dengan nama database, username, dan host anda yang diberikan oleh 100webspace.com.

## Memasukkan data ke database

Sekarang saatnya kita memasukkan data ke database. Login ke member area 100webspace.com. Klik link phpMyAdmin pada control panel. Atau klik nama database yang baru anda buat. Setelah itu anda akan dibawa ke halaman phpMyAdmin.

Tampilannya sama dengan phpMyAdmin yang ada pada komputer kita. Klik gambar  pada pojok kiri atas untuk membuka query window.



Gambar 6.10: phpMyAdmin pada hollosite.com

Setalah window query muncul klik link import files. Arahkan ke direktori tempat dimana anda menyimpan file db\_reseller.sql. Setelah itu klik **Go**, phpMyAdmin kemudian akan memproses query-query yang ada pada file tersebut.

## Mengupload File

Setelah konfigurasi database telah kita lakukan. Saatnya untuk mengupload file-file ladangbisnis.localhost. Pastikan anda terkoneksi ke internet. Jalankan program AceFTP3Freeware atau software FTP lainnya.

Klik menu **File > Connect** untuk membuat koneksi baru. Setelah itu muncul window baru, klik menu **File > Create > New Site Profile**. Muncul window New Site Profile Wizard. Lihat gambar 6.11



Gambar 6.11: Pembuatan profil koneksi baru

Pada isian descriptive isikan sembarang misalnya ladang bisnis. Pada server isikan alamat website anda misalnya id\_anda.hollosite.com. Pada user ID isikan username anda. Pada password isikan password untuk login ke FTP.

Password untuk login ke FTP pada mulanya sama dengan password untuk masuk ke member area. Namun jika anda mengupdate password member area anda. Password untuk FTP belum berubah dan masih sama seperti password awal saat registrasi ke 100webspace.com.

Setelah selesai klik tombol **Finish**. Sekarang profil koneksi yang baru anda buat muncul pada daftar. Untuk melakukan koneksi double klik nama profil tersebut atau klik kanan lalu klik **connect**.

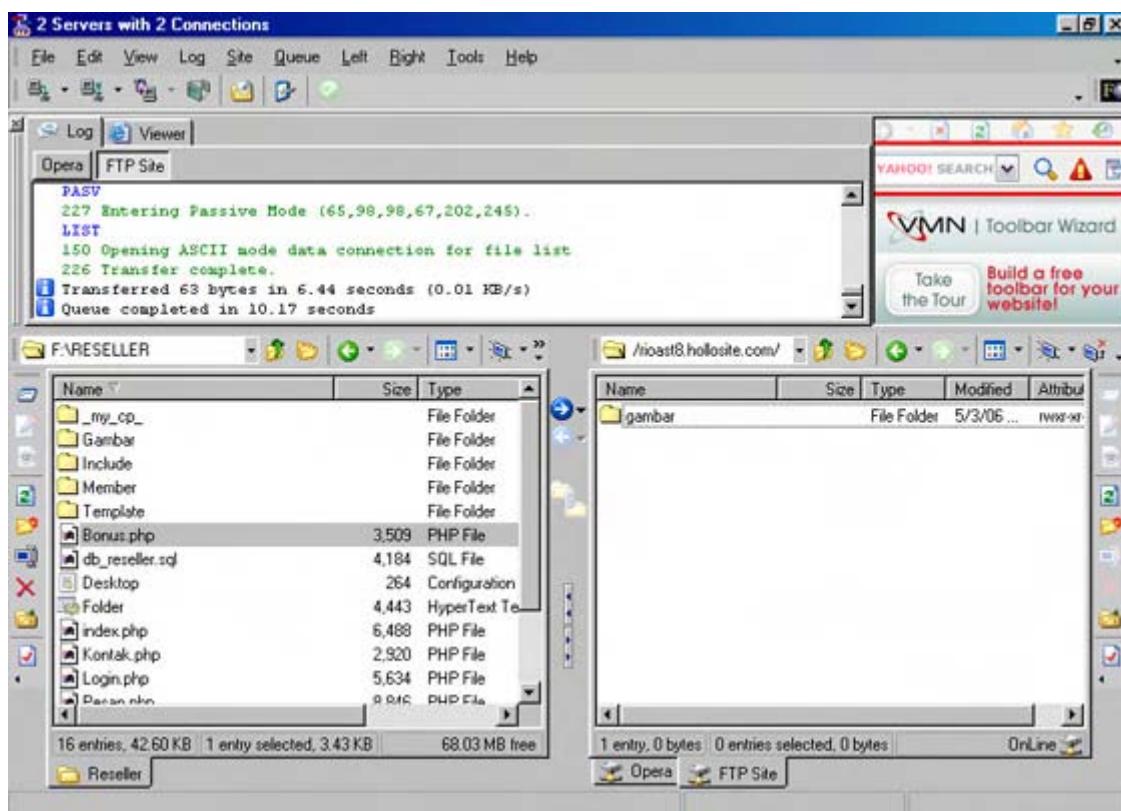
Setelah anda klik connect AceFTP3Freeware langsung melakukan koneksi ke website anda. Setelah koneksi berhasil dilakukan. Lihat pada bagian **Remote Computer** akan muncul sebuah direktori yaitu id\_anda.hollosite.com. Direktori ini merupakan direktori root(paling atas) dari account anda. File utama untuk website anda harus anda letakkan pada direktori id\_anda.hollosite.com ini.

Untuk masuk pada direktori id\_anda.hollosite.com klik ganda direktori tersebut. Setelah itu jendela remote file akan nampak kosong karena belum ada direktori dan file. Pada direktori inilah kita akan membuat folder-folder seperti gambar, template, include, \_my\_cp\_, dan member.

Untuk membuat direktori baru pada Remote Computer ikuti langkah-langkah berikut. Namun sebelumnya pastikan anda sudah masuk pada direktori id\_anda.hollosite.com.

1. Klik kanan pada area Remote Computer lalu klik **Create New Folder**
2. Beri nama folder tersebut **gambar**
3. Ulangi langkah tersebut untuk direktori **member**, **include**, **\_my\_cp\_**, dan **template**

Setelah direktori-direktori telah dibuat. Kita tinggal mengupload file-file dari komputer kita ke server.



Gambar 6.12: Pembuatan direktori pada remote computer

Pada AceFTP3Freeware anda dapat mengupload single atau multiple file sekaligus. namun saya tidak menyarankan anda untuk mengupload semuanya sekaligus. Lebih baik lakukan direktori demi direktori. Ikuti langkah-langkah berikut untuk mengupload file:

1. Pada local computer pastikan anda sudah masuk pada direktori **reseller**
2. Pilih semua file PHP yang ada pada direktori reseller menggunakan **Ctrl-Click** atau **Shift-Click**
3. Setelah semua file PHP terseleksi **Drag** (seret) ke jendela Remote Computer lalu **Drop** (lepaskan) pada bagian kosong dan jangan didrop ke direktori.
4. Atau klik kanan seleksi tersebut **klik Copy**, pada jendela Remote Computer klik kanan lalu **Paste**.
5. Pada jendela Local Computer klik ganda direktori `_my_cp_` untuk masuk ke dalamnya.
6. Seleksi semua file PHP yang ada, **Drag** lalu **Drop** ke direktori `_my_cp_` yang ada pada Remote Computer
7. Ulangi langkah tersebut untuk direktori gambar, include, member, dan template.

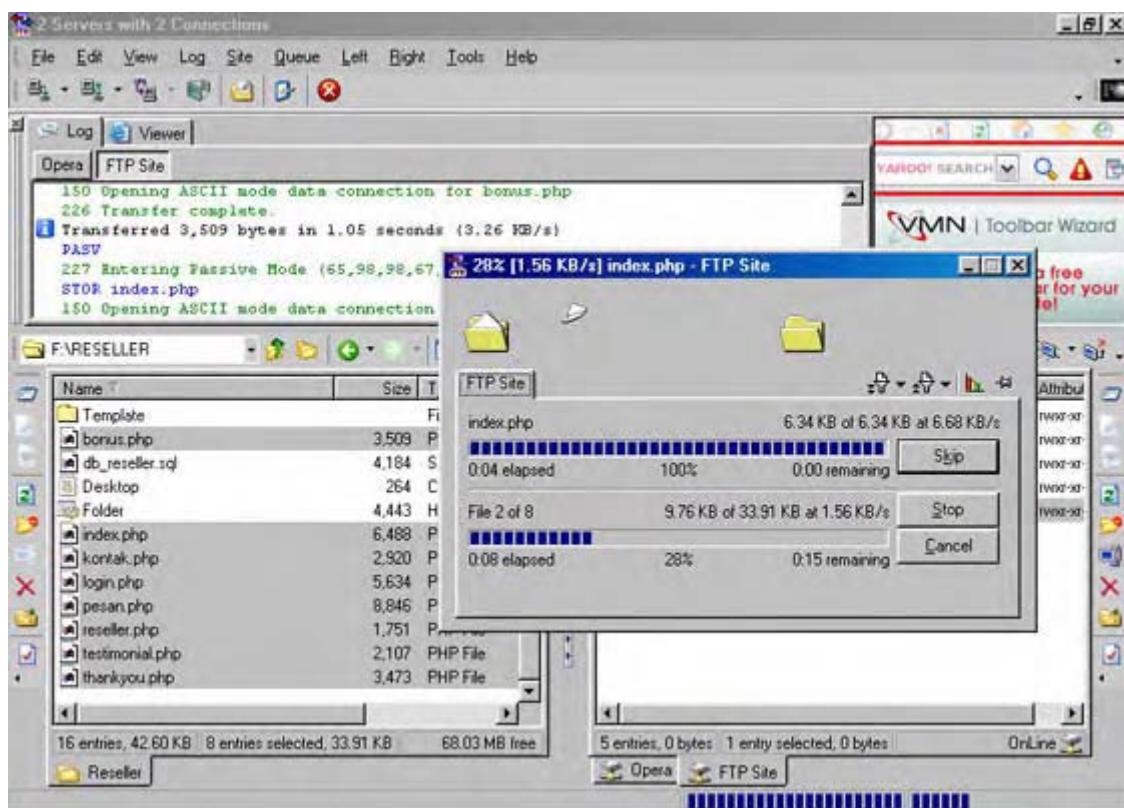
Setelah semua file berhasil terupload. Saatnya untuk mengetes tampilan perdana website kita di dunia maya!?!.

**Catatan:** Pada Windows 98, sebelum mengupload file ke server. Periksalah nama masing-masing file apakah sudah huruf kecil semua. Biasanya windows 98 secara otomatis mengkapitalkan huruf pertama dari nama file. Hal ini perlu diperhatikan karena server yang menggunakan sistem operasi Linux bersifat case sensitive terhadap nama file.

Saudara-saudara untuk menyambut peresmian ladangbisnis.localhost sambutlah presiden direktur kita Mr. Anda.!!. Ini apa-apaan..?. bercanda....

Buka browser anda arahkan ke alamat website anda misalnya id\_anda.hollosite.com. Jika anda ingin mengetahui hasil upload milik saya. Arahkan browser anda ke <http://rioast8.hollosite.com/> atau alamat URL redirection di <http://www.ladang-bisnis.co.nr/>.

Lihat gambar 6.14 untuk melihat aksi ladangbisnis.localhost di internet.



Gambar 6.13: Proses upload sedang berlangsung

Jadikan Internet Ladang Bisnis Paling Menguntungkan - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

Getting Started Latest Headlines file:///d:/Documents...

Jadikan Internet Ladang Bisnis Palin... (Untitled)

100WebSpace.com free web space | Fax to Email | linux hosting | web hosting | dating | reseller hosting | report abuse | links

**ladangbisnis.localhost**

Jadikan internet sumber penghasilan anda

HOME PESAN TESTIMONIAL LOGIN RESELLER BONUS KONTAK

Reseller: Mr. Jono - Surabaya

**Bagaimana Menyiasati Internet Agar Menghasilkan Profit Yang Melimpah!**

**Temukan rahasianya hanya di ladangbisnis.localhost**

Saudara netter, tahukah anda bahwa sesuatu yang kita sebut dengan "INTERNET" ternyata menyimpan segudang misteri. Jangan takut misteri ini bukan berhubungan dengan hal-hal mistis. Melainkan sesuatu yang pasti anda kejar seumur hidup, apalagi kalau bukan "UANG".

Dengan internet kita dapat menghasilkan penghasilan yang melimpah, hanya jika kita tahu CARANYA. Dan untuk itulah e-book "Internet Sebagai Ladang Bisnis" hadir.

Hari ini: Kamis, 11 Mei 2006

**E-BOOK GRATIS!**

Dapatkan e-book tentang bagaimana membangun mailing list berskala global dengan cepat dan mudah secara GRATIS!!.

Cukup isikan nama dan email anda di form berikut. Alamat link download akan kami kirim ke email anda saat ini juga.

Nama:

Email:

Kirim

Gambar 6.14: Aksi ladangbisnis di internet

# Domain dan Web Hosting

Pembahasan berikut saya fokuskan kepada anda yang ingin membeli domain dan web hosting komersial. Ada beberapa hal yang perlu anda perhatikan sebelum akhirnya menjatuhkan pilihan pada salah satu penyedia web hosting.

Untuk anda yang memiliki Credit card, anda memiliki pilihan lebih banyak karena anda dapat membeli web hosting dari luar. Namun bagi anda yang tidak memiliki credit card mungkin pilihan web hosting hanya terbatas pada provider dari Indonesia saja. Tapi hal tersebut tidak masalah toh layanan yang diberikan oleh penyedia lokal tidak kalah dengan milik asing.

Dan harga dari provider-provider web hosting sangat bersaing. Ini tentu menguntungkan kita sebagai konsumen.

Berikut ini adalah daftar beberapa penyedia web hosting di Indonesia:

<http://www.indohoster.com/>  
<http://www.superbighosting.com/>  
<http://www.masterwebnet.com/>  
<http://www.webhosting.i2.co.id/>

Untuk web hosting lainnya anda dapat mencarinya di google. Caranya pergi ke <http://www.google.co.id/> pilih Halaman dari Indonesia. Dan pada isian cari ketikkan *Web Hosting*. Silahkan lakukan review masing-masing provider.

Para penyedia web hosting biasanya membagi-bagi paket web hosting mereka sesuai dengan harga. Semakin mahal harga paket tersebut semakin lengkap fasilitas yang anda peroleh. Misalnya kapasitas penyimpanan, jumlah database, jumlah sub domain, FTP account, dan lain sebagainya.

Silahkan lakukan perbandingan masing-masing paket dalam satu web hosting juga lakukan perbandingan antar web hosting agar dapat menentukan pilihan dengan lebih cermat sesuai kebutuhan anda.

## Memilih domain

Pada umumnya penyedia layanan web hosting sudah membundel pembelian nama domain menjadi satu dengan pembelian web hosting. Khusus untuk nama domain biasanya masa berlakunya hanya satu tahun. Dan sebelum masa aktif domain habis anda diminta untuk memperpanjang kontrak anda. Jika tidak domain anda dapat dinonaktifkan.

Untuk memilih nama domain biasanya kita sesuaikan dengan apa yang ingin kita tampilkan pada website tersebut. Misalnya pada aplikasi proyek untuk Bab5 yaitu ladangbisnis.localhost, nama domain yang cocok untuk proyek tersebut adalah ladangbisnis.com.

Jika anda ingin membuat website tentang bisnis atau sesuatu yang bersifat umum. Jangan menggunakan nama pribadi anda sebagai nama domain. Hal tersebut hanya akan mengurangi minat user untuk mengunjungi website anda.

Kita juga harus cermat memilih akhiran dari domain kita apakah .com, .biz, .org, .net dan lainnya. Masing-masing akhiran sebenarnya mengindikasikan content dari website tersebut. Misalnya .com, isi dari website tersebut umumnya adalah komersial (*commercial*). .biz untuk website bisnis, .org untuk organisasi non profit.

Orang akan lebih mudah mengingat alamat dengan akhiran .com daripada akhiran lainnya. Jadi saya sarankan kepada anda untuk memilih akhiran .com.

## Kalender



**Selamat Anda Telah Menjadi  
WebMaster**

---

# Penutup

Tidak terasa sudah 30 hari anda telah membaca dan mengaplikasikan materi dari e-book ini. Dan saya yakin pada hari yang ke-30 ini anda telah menjadi webmaster.

Sekarang anda dapat membuat website seperti rahasia-webmaster.com dengan mudah. Karena anda telah memahami bagaimana membuat sebuah website yang dinamis. Yang terdapat sistem database.

Dengan berbekal materi yang telah saya berikan pada e-book ini dari hari ke-1 sampai hari ke-30. Anda dapat membuat website lain seperti website untuk iklan baris, website reseller dan masih banyak lagi.

Sekarang penulis yakin jika anda mengunjungi suatu website, anda dapat berkata, "Hmm, Rasanya aku bisa membuat website seperti ini".

Ada sebuah kalimat yang wajib anda pegang:

Dalam dunia programming tidak ada kata "berhenti" untuk terus belajar dan meningkatkan kemampuan.

Dunia pemrograman terus berkembang dari hari ke hari. Untuk itu anda dituntut untuk mengikutinya jika anda tidak ingin tertinggal. Rajin-rajinlah untuk mengunjungi situs-situs pemrograman. Download berbagai script sebagai bahan referensi. Anda dapat mencari situs-situs pemrograman lewat google. Ketikkan saja *PHP Programming* atau *PHP Forum* atau *Free PHP Script*.

Selalu update software anda dengan yang terbaru. Untuk itu kunjungi website PHP, MySQL dan Apache secara regular untuk mengetahui perkembangan terbaru.

Komentar, kritik dan saran anda untuk e-book ini sangat saya harapkan. Anda dapat memberikan komentar, kritik atau saran anda melalui form testimonial yang ada di halaman member area.

Satu kalimat terakhir, Mari bersama memajukan dunia TI di Indonesia!.

**"Menjadi WebMaster Dalam 30 Hari"**

Rio Astamal – admin@rahasia-webmaster.com

© Copyright 2005-2006 www.rahasia-webmaster.com