

Risawandi

MUDAH MENGUASAI
PHP & MySQL

DALAM **24** JAM



Dilengkapi Pemrograman Embed Siaran Televisi
Beberapa Negara ke Situs Pribadi Anda

UNIMAL PRESS

**MUDAH MENGUASAI
PHP & MySQL**

DALAM **24** JAM



universitas
MALIKUSSALEH

RISAWANDI

**MUDAH MENGUASAI
PHP & MySQL
DALAM **24** JAM**

UNIMAL PRESS

Judul: **MUDAH MENGUASAI PHP & MySQL DALAM 24 JAM**

viii + 72 hal., 15 cm x 23 cm

Cetakan Pertama: Januari, 2019

Hak Cipta © dilindungi Undang-undang. *All Rights Reserved*

Penulis:

RISAWANDI

Perancang Sampul &

Penata Letak: **Eriyanto**

Pracetak dan Produksi: **Unimal Press**

Penerbit:

UNIMAL PRESS

Unimal Press

Jl. Sulawesi No.1-2

Kampus Bukit Indah Lhokseumawe 24351

PO.Box. 141. Telp. 0645-41373. Fax. 0645-44450

Laman: www.unimal.ac.id/unimalpress.

Email: unimalpress@gmail.com

ISBN:

978 – 602 –464- 074-3

ISBN 978-602-464-074-3



Dilarang keras memfotocopy atau memperbanyak sebahagian atau seluruh buku ini tanpa seizin tertulis dari Penerbit

Kata Pengantar

Pembahasan mengenai PHP hampir selalu hadir dan berpasangan dengan MySQL. PHP dan MySQL merupakan bahasa pemrograman dan database yang terpopuler di dunia saat ini. Semakin populernya, banyak para programmer non-PHP dari yang awal hingga yang tingkat mahir tergoda untuk ingin mempelajarinya. Selain perangkat lunaknya yang open-source, PHP memiliki keunggulan dalam hal kecepatan pengeksekusian kode-kode programnya, sehingga menjadi pilihan utama pemrograman bagi para programer di Indonesia, alih-alih PHP didukung dengan database MySQL yang memiliki kapasitas penyimpanan datanya yang cukup besar.

Buku ini menjembantani para programer non-PHP dan para pemula yang awam dengan programming, untuk mengenal secara praktis dan singkat mengenai PHP. Untuk itu penulis hanya memilih beberapa materi pokok dasar saja, untuk menghiasi bab demi bab materi buku ini. Materi yang dibahas dibuka dengan sejarah PHP, instalasi PHP, menggunakan PHP Designer, sejarah MySQL, tipe data, operator, operasi pada string, dilanjutkan dengan pernyataan kondisi, perulangan, array, fungsi, bekerja dengan form, koneksi database MySQL, dan diakhiri dengan Streaming Televisi melalui situs pribadi.

Tidak ada kesempurnaan tanpa kekurangan, dan buku ini-pun begitu. Penulis sebelumnya meminta maaf sebesar-besarnya jika di dalam buku ini masih banyak terdapat kekurangan atau kekhilafan penulis yang alpa menuliskan referensi dari bahan-bahan materi yang disadur. Akhir kata, penulis mengucapkan selamat membaca semoga pengetahuan di dalamnya tidak begitu mengecewakan bagi para pembaca.

Penulis

RISAWANDI

Terima Kasih Kepada:

Bapak M. Arhami, M.Kom., Bapak Dr. Setiawan Hadi (UNPAD), Bapak Ir. Syamsul Bahri, M.Si., Lettu Inf. Ismail Mangki., Bapak Dr. Nasir Usman., M.Pd., Ibu Dr. Murniati., M.Pd., Bapak Nasir Ismail Mangki., Ibunda Nezi Akhtar., Bapak Apridar, SE, M.Si., Nurlaila, M.Pd., Iskandarsyah., Muhammad Iqbal., Lisa Pebrina., Mirza Zamzami., Muchsin., Pak A. Razaq, M.Pd., dan seluruh rekan-rekan yang telah membantu secara moril kepada penulis untuk terus merealisasikan buku ini.

Daftar Isi

Kata Pengantar	v
Terima Kasih Kepada:.....	vi
Daftar Isi.....	vii

Bab 1

PENDAHULUAN	1
1.1 Sejarah PHP	1
1.2 Keunggulan PHP dari Bahasa Pemrograman Lain.....	2
1.3 Instalasi PHP	2
1.4 Instalasi PHP Designer	9
1.5 Sejarah MySQL	14

Bab 2

MULAI MENGGUNAKAN PHP	21
2.1 Sintaks PHP	21
2.2 Variabel dalam PHP	23
2.3 Operator dalam PHP.....	24
2.4 Berbagai Operasi pada String	25

Bab 3

PERNYATAAN KONDISIONAL	29
3.1 if ... else ... dalam PHP	29
3.2 Pernyataan switch dalam PHP	31

Bab 4

PERULANGAN	35
4.1 Perulangan dalam PHP	35
4.2 while.....	35
4.3 do ... while.....	36
4.4 Pernyataan for	36
4.5 Pernyataan foreach.....	37

Bab 5

ARRAY	39
5.1 Array dalam PHP.....	39
5.2 Array Numerik.....	39
5.3 Array Asosiatif	40
5.4 Array Multidimensi.....	40

Bab 6	43
FUNGSI & FORM	43
7.1 Fungsi dalam PHP	43
7.2 Bekerja dengan Beberapa Form	46
7.3 \$_GET	47
7.4 \$_POST	48
 Bab 7	 49
PEMROGRAMAN DATABASE MENGGUNAKAN MySQL	49
7.1 Pengenalan MySQL	49
7.2 Koneksi Database MySQL	54
7.3 CREATE DATABASE	55
7.4 CREATE TABLE	56
7.5 INSERT INTO	58
7.6 Menginput Data dari Form ke Database	59
7.7 SELECT	60
7.8 WHERE	62
7.9 ORDER BY	63
7.10 UPDATE	64
7.11 DELETE	65
 Bab 8	 67
MENINTEGRASIKAN BEBERAPA STASIUN TELEVISI DI SITUS PRIBADI	67
8.1 Pendahuluan	67
8.2 Embed Kode untuk Televisi	69
 DAFTAR PUSTAKA	 72

Bab 1

PENDAHULUAN

1.1 Sejarah PHP

Pada awalnya PHP digunakan sebagai bahasa pemrograman untuk sebuah server-side HTML-embedded dengan nama **Personal Home Pages**. PHP pertama kali dibuat oleh **Rasmus Lerdorf** pada tahun 1995. Pada waktu itu PHP masih bernama FI (Form Interpreted), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data form dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI, kependekan dari *Hypertext Preprocessing'/Form Interpreter*. Dengan perilisannya kode sumber ini menjadi *open source*, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.¹

1.2 Keunggulan PHP dari Bahasa Pemrograman Lain.

Beberapa keunggulan yang ditawarkan PHP dibandingkan beberapa bahasa pemrograman lainnya adalah :

- ✘ Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaanya.
- ✘ Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
- ✘ Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
- ✘ Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
- ✘ PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah sistem.

1.3 Instalasi PHP

Sebelum kita masuk ke dalam bagian penginstalasian PHP, maka pastikan kita telah memiliki software :



appserv-win32-2.5.10



phpdesigner_setup
PHP Designer 2005 Setup
MPSOFTWARE

dan kedua software tersebut dapat didownload pada situs :

- ✘ <http://www.apache.org>
- ✘ <http://www.php.net>
- ✘ <http://www.mysql.com>
- ✘ <http://www.phpmyadmin.net>.

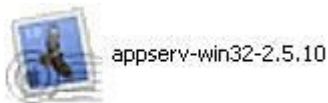
¹Diperoleh dari "[http://id.wikibooks.org/wiki/ Pemrograman_PHP/ Pendahuluan/Sejarah_PHP](http://id.wikibooks.org/wiki/Pemrograman_PHP/Pendahuluan/Sejarah_PHP)."

Appserv² merupakan software paket instalasi open source yang di dalamnya memuat :

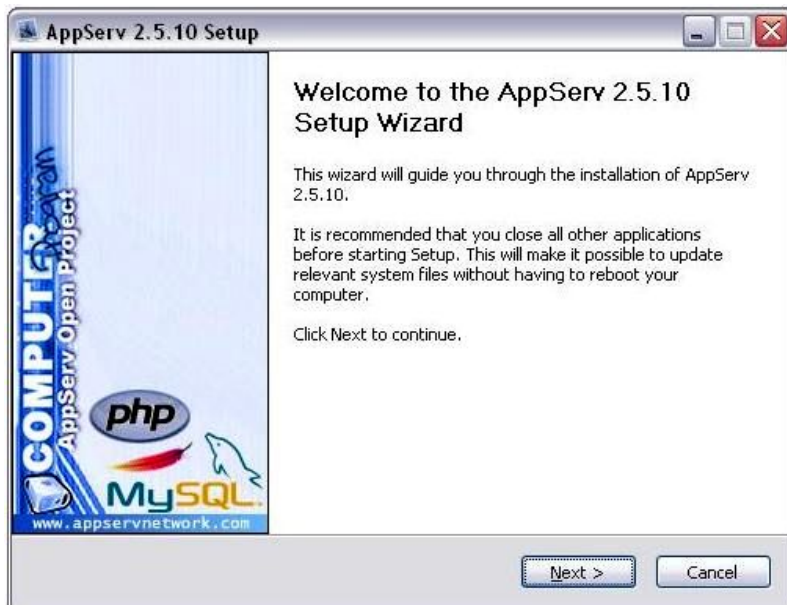
- ✗ Apache Web Server Version 2.2.8
- ✗ PHP Script Language Version 5.2.6
- ✗ MySQL Database Version 5.0.51b
- ✗ phpMyAdmin Database Manager Version 2.10.3.

Sedangkan PHP designer³ merupakan software yang memfasilitasi penyuntingan berbagai kode pemrograman PHP. Langkah-langkah yang diperlukan dalam menginstalasi PHP adalah sebagai berikut :

1. Klik 2x pada file



2. Setelah muncul tampilan berikut :

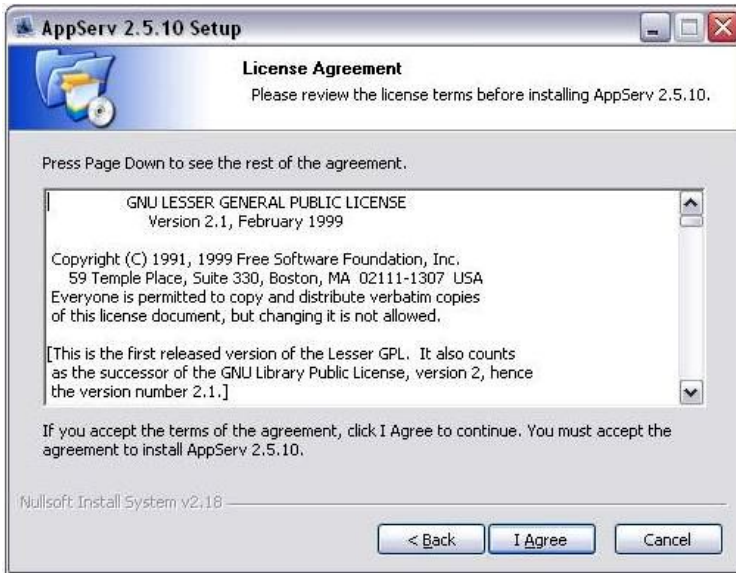


klik button [Next].

² Program dapat dipesan gratis melalui email Fadlisyah@ymail.com.

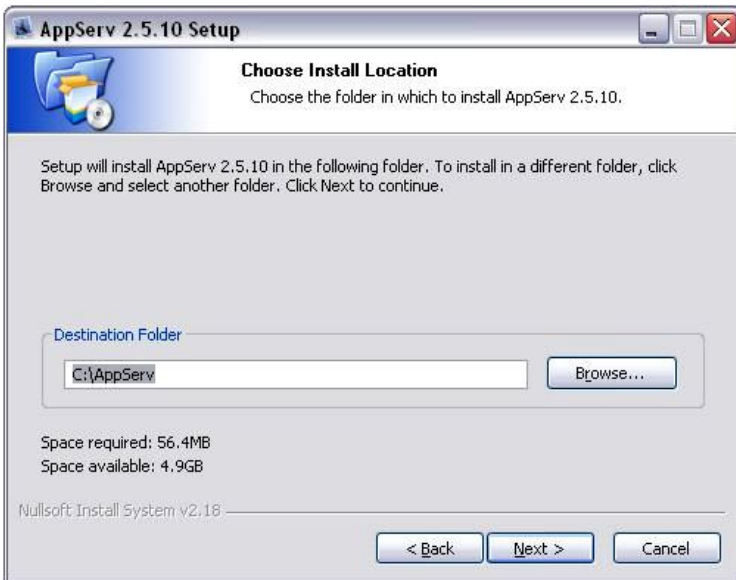
³ idem

3. Tampilan sekarang akan berupa :



selanjutnya klik button [I Agree].

4. Tampilan berikutnya :



dan klik button [Next].

5. Setelah muncul tampilan :



klik button [Next].

6. Lengkapi nama server [localhost] dan Administrator's Email Address [sesuaikan dengan email anda] untuk tampilan :



dan klik button [Next].

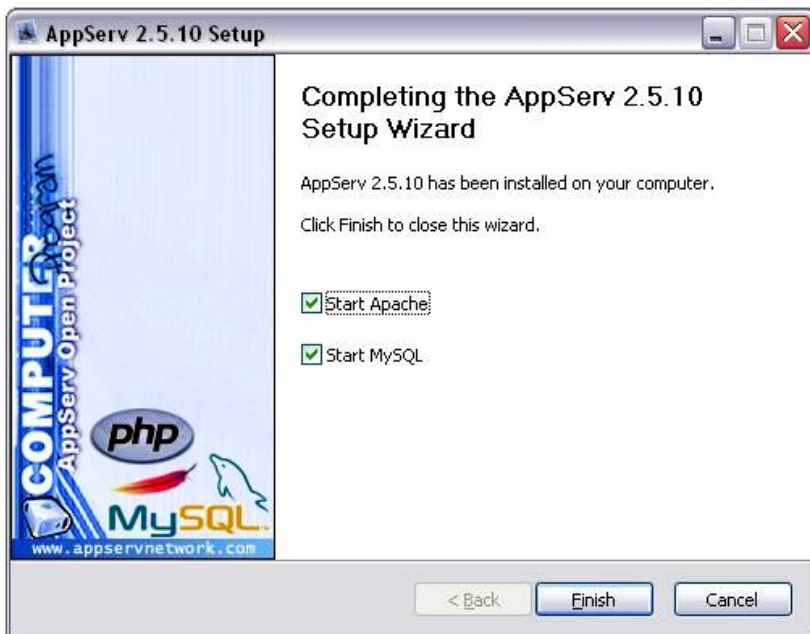
7. Masukkan password root [root] untuk server MySQL.



klik button [Install], dan tunggu hingga proses instalasi selesai.



8. Setelah proses instalasi selesai, maka muncul tampilan :



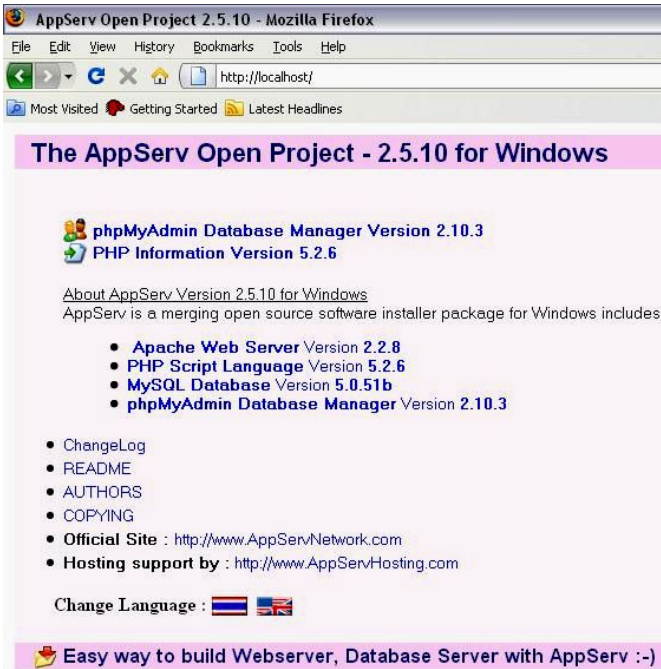
dan akhiri dengan mengklik button [Finish].

Sekarang pada direktori C:\ dalam harddisk anda terdapat folder :

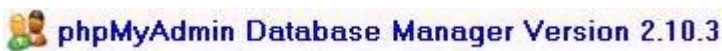


dengan folder www sebagai tempat penyimpanan berkas-berkas pemrograman PHP yang akan kita buat.

Coba jalankan salah satu software browser web seperti : Internet Explorer, Opera, atau Mozilla Firefox. Lalu ketikkan **localhost** pada dialog browser tersebut, maka akan muncul tampilan AppServ hasil dari instalasi yang kita lakukan tadi.



Sekarang jika kita mengklik pada item menu :

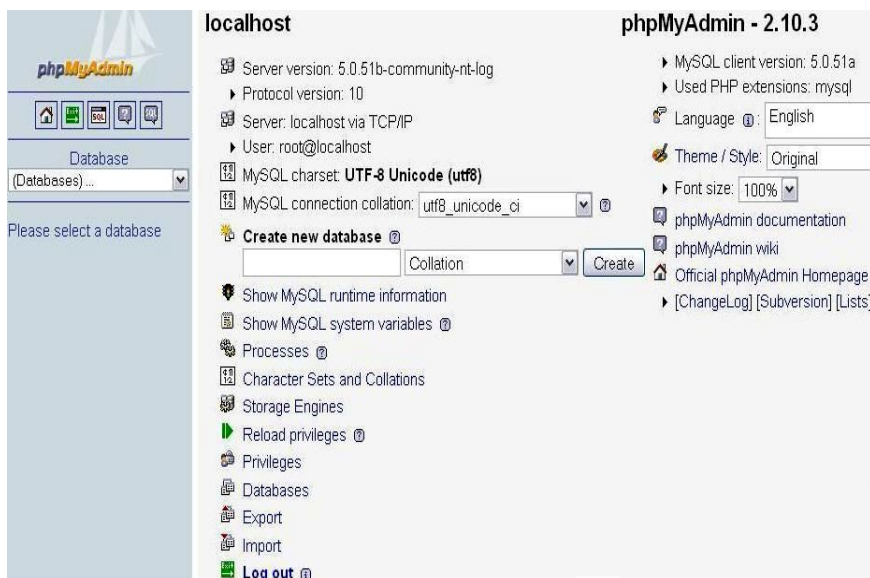


dan jika muncul dialog konfirmasi password, seperti tampilan berikut :



maka isikan User Name dan Password, keduanya dengan **root**.

Tampilan yang muncul selanjutnya adalah,



dan tampilan inilah nantinya kita perlukan untuk mendesain database yang akan dikoneksikan ke PHP. Sampai di sini, tugas instalasi dan penjelajahan secara sederhana telah selesai kita lakukan. Sekarang kita dapat keluar dari browser web kita.

1.4 Instalasi PHP Designer

PHP Designer kita gunakan sebagai tempat untuk menulis dan menyunting berbagai kode program kita. Untuk menginstalasi PHP Designer, maka kita ikuti langkah-langkah berikut :

1. Klik 2x pada file berikut :

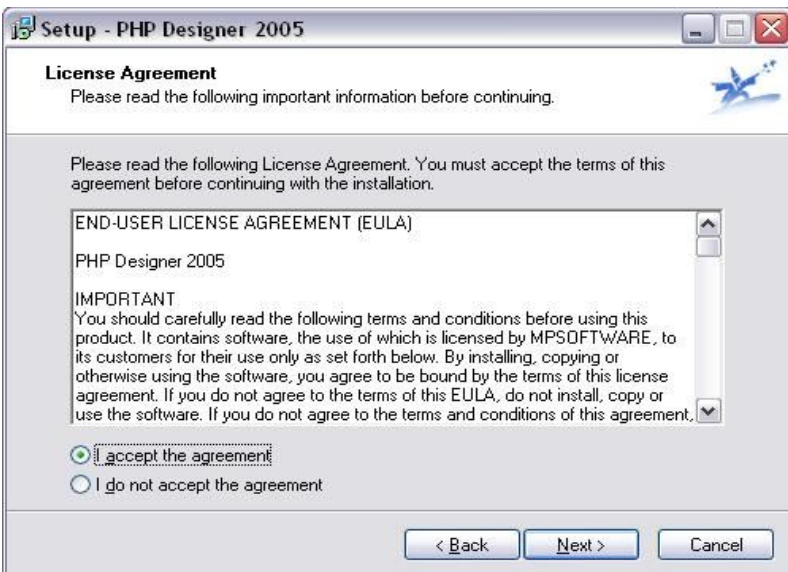


2. Setelah muncul tampilan :



selanjutnya klik button [Next].

3. Lalu muncul form konfirmasi untuk meminta persetujuan pemakai.



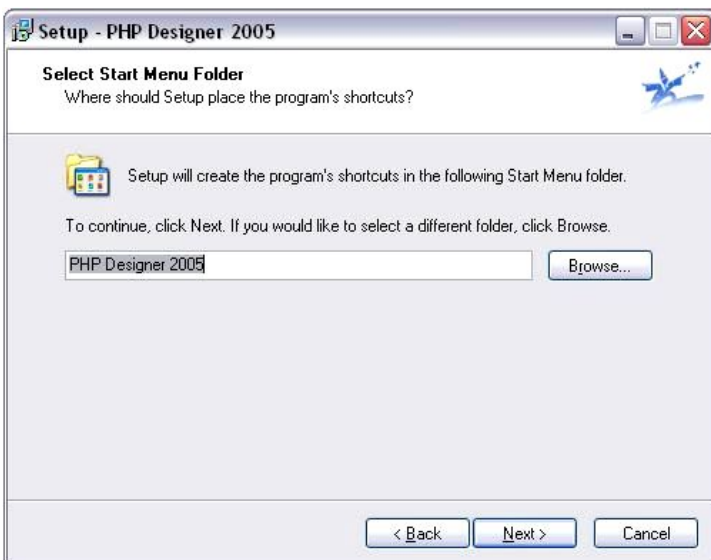
dan klik button [Next].

4. Dan muncul tampilan,



selanjutnya klik button [Next].

5. Tampilan sekarang akan berupa,

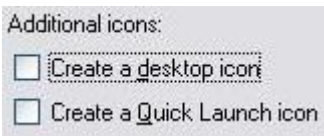


masih klik button [Next].

6. Tampilan berikut merupakan tampilan konfirmasi,



jika kita ingin tampilan icon PHP Designer pada desktop dan pada quicklaunch maka check pada menuitem berikut,



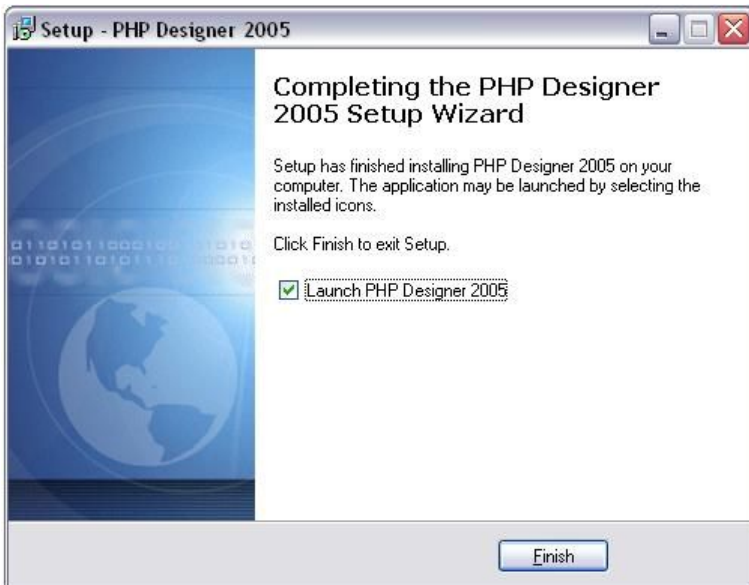
dan jika tidak diperlukan, biarkan kosong dan langsung ke form konfirmasi instalasi dengan mengklik button [Install].



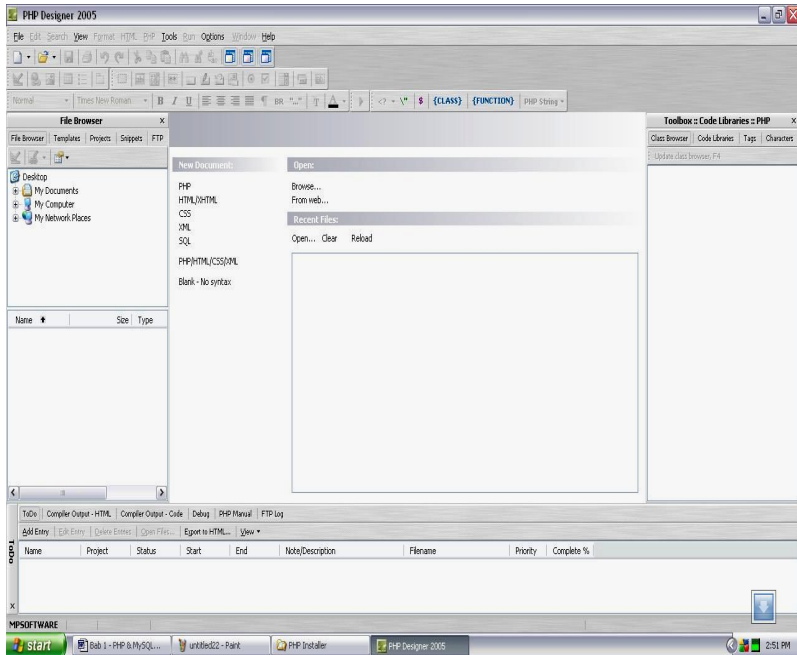
klik button [Next].



7. Setelah proses instalasi selesai klik [Finish] pada tampilan berikut,



dan tak lama kemudian muncul IDE PHP Designer.



Sampai di sini tugas penginstalasian PHP Designer telah selesai.

1.5 Sejarah MySQL

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama MySQL AB, yang kala itu bernama TcX DataKonsult AB, sejak sekitar 1994–1995, meski cikal bakal kodenya bisa disebut sudah ada sejak 1979. Tujuan mula-mula TcX membuat MySQL pada waktu itu juga memang untuk mengembangkan aplikasi Web untuk klien—TcX adalah perusahaan pengembang software dan konsultan database. Kala itu Michael Widenius, atau “Monty”, pengembang satu-satunya di TcX, memiliki aplikasi UNIREG dan rutin ISAM yang dibuat sendiri dan sedang mencari antarmuka SQL untuk ditempelkan di atasnya. Mula-mula TcX memakai mSQL, atau “mini SQL”. Barangkali mSQL adalah satu-satunya kode database open source yang tersedia dan cukup sederhana saat itu, meskipun sudah ada Postgres. Namun ternyata, menurut Monty, mSQL tidaklah cukup cepat maupun fleksibel. Versi pertama mSQL bahkan tidak memiliki indeks. Setelah mencoba menghubungi David Hughes—pembuat mSQL—dan ternyata mengetahui bahwa David tengah sibuk mengembangkan versi dua, maka keputusan yang diambil Monty yaitu membuat sendiri mesin SQL yang antarmukanya mirip dengan

mSQL tapi memiliki kemampuan yang lebih sesuai kebutuhan. Lahirlah MySQL.

Nama MySQL (baca: mai és kju él) tidak jelas diambil dari mana. Ada yang bilang ini diambil dari huruf pertama dan terakhir nama panggilan Michael Widenius, Monty. Ada lagi yang bilang kata My diambil dari nama putri Monty, yang memang diberi nama My—karena Monty memang aslinya seorang Finlandia. Tapi sebetulnya kalau source code MySQL dilirik, prefiks my memang sudah terbubuhi di mana-mana—prefiks ini sering menjadi prefiks umum kalau seseorang membuat kode kustom tersendiri untuk sesuatu. Kalau Anda betul-betul penasaran mana yang benar, mungkin bisa bertanya langsung kepada Monty.

MySQL versi 1.0 dirilis Mei 1996 secara terbatas kepada empat orang. Baru di bulan Oktober versi 3.11.0 dilepas ke publik. Namun mula-mula kode ini tidak diberikan di bawah lisensi General Public License, melainkan lisensi khusus yang intinya kurang lebih begini: “Source code MySQL dapat dilihat dan gratis, serta server MySQL dapat dipakai tanpa biaya tapi hanya untuk kebutuhan nonkomersial. Untuk kebutuhan komersial (mis: mengemas dan menjual MySQL, atau menyertakan MySQL dalam program komersial lain) Anda harus bayar lisensi.” Sementara distribusi Windows MySQL sendiri dirilis secara shareware. Barulah pada Juni 2000 MySQL AB mengumumkan bahwa sejak versi 3.23.19, MySQL adalah software bebas berlisensi GPL. Artinya, “Source code MySQL dapat dilihat dan gratis, serta server MySQL dapat dipakai tanpa biaya untuk kebutuhan apa pun. Tapi jika Anda memodifikasi source code, Anda juga harus melepasnya di bawah lisensi yang sama, yaitu GPL.” Kini perusahaan MySQL AB, yang beranggotakan sekitar 10 prografer dan 10 karyawan lain itu, memperoleh pemasukan terutama dari jasa konsultasi seputar MySQL.

Versi publik pertama, yang hanya berjalan di Linux dan Solaris serta sebagian besar masih belum terdokumentasi itu, dengan berangsur-angsur diperbaiki dan ditambah fitur demi fiturnya—tapi tetap dengan fokus utama pengembangan pada kelangsingan dan kecepatan. Artinya, fitur yang menyebabkan MySQL menjadi lambat tidaklah ditambahkan, atau ditunda dulu, atau ditambahkan tapi menjadi fitur yang opsional.

Versi awal MySQL ini, meski sudah bisa dipakai untuk aplikasi Web sederhana, belumlah memadai sama sekali untuk aplikasi bisnis. Contohnya, JOIN sederhana sudah ada, tapi tidak ada HAVING—baru di bulan Desember ditambahkan. Sudah ada tipe data TIMESTAMP

dan kolom autoupdate, tapi tidak ada system-generated number (sequence)—baru di akhir 1996 juga ditambahkan modifier kolom AUTO_INCREMENT. Sudah ada LIMIT tapi GROUP BY dan ORDER BY memiliki keterbatasan. Dan seterusnya.

Barulah di versi-versi akhir 3.22—sepanjang 1998–1999—MySQL menjadi semakin populer dan dilirik orang. Stabilitasnya sudah baik. Kecepatannya meningkat. Sudah tersedia di berbagai platform, termasuk Windows. Seri 3.22 ini banyak dipakai di berbagai instalasi, mungkin hingga sekarang, sehingga MySQL AB tetap memberikan dukungan technical support untuk seri ini.

Kalau di seri 3.22 MySQL mulai diadopsi banyak orang dan meningkat populasi penggunaannya, maka di seri 3.23 dan 4.0-lah terjadi banyak peningkatan dari sisi teknologi. Ini tidak terlepas dari tuntutan pemakai yang semakin mengandalkan MySQL, namun membutuhkan fitur-fitur yang lebih banyak lagi. Sejak dari sebelum 3.22 dan awal 3.23, MySQL terus dikritik praktisi database maupun penggunaannya mengenai tidak adanya fasilitas transaksi (COMMIT dan ROLLBACK). Pengembang MySQL menyarankan penggunaan LOCK TABLES untuk mengatasi masalah update atomik, tapi ini tentu saja bukan pengganti transaksi, karena tidak adanya kemampuan membatalkan efek perubahan di tengah jalan. Di seri 3.23-lah MySQL mulai memiliki kemampuan transaksi, row-level locking, dan foreign key constraint. Meski beberapa fasilitas populer lainnya masih belum ada—seperti subselect, view, dan trigger—ini sudah direncanakan dan akan mulai ditambahkan di seri 4.x.

Seri 3.23. Di seri 3.23 MySQL menambahkan tiga jenis tabel baru: pertama MyISAM, yang sampai sekarang menjadi tipe tabel default; kedua BerkeleyDB, yang pertama kali menambahkan kemampuan transaksi pada MySQL; dan ketiga InnoDB, primadona baru yang potensial. MySQL memang memiliki arsitektur yang memungkinkan tiap tabel ditangani oleh *handler* yang berbeda. Handler ini menerima perintah akses dan modifikasi dari lapisan MySQL yang lebih atas dan mewujudkannya secara fisik dari dan ke disk. Tabel tipe MyISAM merupakan tabel yang lebih cepat dari tabel ISAM, karena pola aksesnya telah disesuaikan dan dioptimasi untuk pola akses SQL. Selain itu MyISAM mendukung indeks pada kolom bertipe TEXT dan BLOB, serta mendukung tipe indeks FULLTEXT. Tabel tipe BerkeleyDB, atau BDB, menggunakan database embedded BerkeleyDB yang sudah terkenal itu untuk memanfaatkan kemampuan transaksinya. Sejak memiliki handler BDB, MySQL naik statusnya menjadi database yang ACID compliant, sesuatu yang amat

penting bagi keamanan data. Namun BerkeleyDB tidaklah terlalu optimal untuk sebuah database SQL, sehingga akhirnya sebuah perusahaan Finlandia Innobase Oy membuat handler table baru bagi MySQL yang menggunakan database embedded InnoDB-nya—kalah itu bernama Innobase, namun berganti nama karena masalah trademark produk Innobase yang sudah ada sebelumnya.

InnoDB membuat MySQL menarik karena peningkatan kecepatan dan kemampuan tambahan yang dimungkinkannya. Pertama-tama, InnoDB memiliki fitur transaksi dengan sistem multiversi. Artinya, jika sebuah klien memulai transaksi, maka perubahan yang dilakukan klien tersebut tidak akan terlihat oleh klien lain. Klien lain akan melihat kondisi tabel sebelum transaksi. Barulah jika transaksi oleh klien pertama dicommit, perubahan ini menjadi terlihat di semua klien lain. Bandingkan ini dengan tabel BDB misalnya, di mana klien yang ingin mengakses sebuah tabel yang sedang dipakai dalam sebuah transaksi akan diblok. Dengan kata lain, multiversi menghindari bloking yang tidak perlu. Ini ditambah lagi dengan row-level locking yang disediakan InnoDB, sehingga mempertinggi konkurensi dan mencegah terlalu banyak bloking. Handler tabel MySQL yang lain hanya mendukung table-level locking, sehingga sebuah klien hanya bisa memblok dengan tingkat granulasi tabel dan berpotensi memblok banyak klien lain yang ingin mengakses tabel yang sama. PostgreSQL dan beberapa database komersial sudah memiliki kemampuan-kemampuan ini. Berkat InnoDB, MySQL pun kini memiliki keduanya. Selain itu, InnoDB menambahkan foreign key constraint, sesuatu yang sering didambakan pengguna MySQL karena bisa menyederhanakan logika pemrograman dan menjaga kekonsistenan database mereka.

Seri 4.x. Di seri yang baru berjalan hingga 4.0 tahap alfa ini, pengembang MySQL berjanji akan menjadikan MySQL satu derajat lebih tinggi lagi. Fitur-fitur yang sejak dulu diminta akan dikabulkan, seperti subselect (di 4.1), union (4.0), foreign key constraint (4.0 atau 4.1—meski InnoDB sudah menyediakan ini di 3.23.x), stored procedure (4.1), view (4.2), cursor (4.1 atau 4.2), trigger (4.1). MySQL AB tetap berdedikasi mengembangkan dan memperbaiki MySQL, serta mempertahankan MySQL sebagai database open source terpopuler.

Di versi 3.23 dan selanjutnya MySQL semakin menarik untuk dilirik oleh pengguna non-Web. Situs mysql.com di halaman depannya dengan bangga memajang profil para pengguna MySQL yang terkenal, seperti Yahoo! Finance (yang mulai mengaku

menggunakan MySQL beberapa waktu lalu di milis mysql), NASA (yang membuang Oracle demi memotong budget), atau perusahaan sekuritas Jepang Aizawa. MySQL tidak lagi sekedar dipercaya menyimpan data posting komentar pengunjung situs Web (seperti di slashdot.org, pengguna veteran MySQL) atau hit atau log Web, tapi juga kini sebagai tempat menyimpan data keuangan dan transaksi bisnis. Meskipun demikian, karakteristik pengguna database—yang biasanya terdiri dari perusahaan—adalah cenderung konvensional dalam mengadopsi produk baru. Jadi jangan terkejut kalau misalnya sampai sekarang Anda akan masih menjumpai orang yang menganggap MySQL belum ACID-compliant atau belum bisa melakukan transaksi. Padahal sudah dua tahun fitur ini ada di MySQL.

Sebetulnya babak baru pengembangan MySQL—yaitu saat MySQL dirilis sebagai produk open source sejati—dimulai saat perusahaan kecil MySQL AB ini menjalin kerja sama dengan sebuah perusahaan database Amerika bernama Progress Software di tahun 2000. Progress membantu meyakinkan Monty dkk agar mengganti lisensi produk MySQL menjadi GPL. Selain itu, Progress juga memasuki bisnis open source dengan membentuk NuSphere, yang akan menjual distribusi MySQL khusus. Model bisnis mengemas software open source nampaknya cukup laik di mata Progress, melihat kesuksesan distro-distro Linux dan perusahaan seperti Red Hat. Karena itu Progress juga berencana menyuntikkan dana hingga \$2,5 juta dolar kepada MySQL AB, dengan tujuan agar MySQL bisa menjadi lebih baik lagi. Satu lagi, Nusphere juga akan membuat table handler baru bernama Gemini, yang didasarkan pada software database Progress. Gemini akan menambahkan row-level locking dan transaksi pada MySQL.

So far so good. Tapi ternyata, beberapa tindakan yang dilakukan oleh Nusphere membuat MySQL AB kesal. Pertama, Nusphere mengambil domain mysql.org (MySQL AB sendiri memiliki mysql.com hasil pemberian Patrick Lynch) dan tidak membuat pernyataan yang jelas di situs tersebut bahwa MySQL dikembangkan oleh MySQL AB, bukan oleh Nusphere. Dengan kata lain Nusphere memakai nama MySQL dan seolah mengaku menjadi pengembang MySQL, padahal MySQL sejak awal dikembangkan oleh MySQL AB. Kedua, Nusphere dengan distribusinya yang bernama “Nusphere MySQL Advantage” melanggar GPL. Ini ironis, karena perusahaan induknya Progress-lah yang setahun sebelumnya membantu MySQL menjadi GPL. Distribusi binary Nusphere ini megandung kode Gemini

yang dilink statik ke MySQL. Menurut lisensi GPL, ini memang berarti kode Gemini pun harus dirilis sebagai GPL. Nyatanya, Nusphere tidak memberikan source code Gemini, tapi *hanya* berjanji akan melepaskannya nanti. MySQL AB mencoba meminta kembali domain mysql.org dan mendesak Nusphere merilis source code Gemini, tanpa hasil.

Perselisihan kemudian menjadi memanas ketika pada tanggal 15 Juni 2001 Nusphere dan Progress mengajukan tuntutan ke pengadilan atas David Axmark (salah satu karyawan awal dan “penginji” MySQL), Monty, dan MySQL AB, dengan tuduhan melanggar kontrak perjanjian. MySQL AB tidak mau kalah, ia menuntut balik Nusphere atas tuduhan pelanggaran trademark, perjanjian, dan lisensi GPL. Hingga buku ini ditulis, proses pengadilan masih berlangsung. Oleh beberapa pihak keputusan akhir pengadilan akan sangat dinanti, karena jika MySQL AB menang maka supremasi dan validitas lisensi GPL akan benar-benar terbukti “di lapangan.”

Di luar masalah ini, MySQL terus dikembangkan secara aktif oleh MySQL AB. Dan untung bagi para pemakai, Gemini kini praktis telah digantikan oleh InnoDB yang memberikan kemampuan serupa. Saat ini domain mysql.org juga telah diambil alih oleh MySQL AB.⁴

∞

⁴ Dikutip dari file:///C:/Downloads/fokus.html.

This page is intentionally left blank

Bab 2

MULAI MENGGUNAKAN PHP

2.1 Sintaks PHP

Ada beberapa jenis pasangan tag PHP yang dapat digunakan untuk menandai blok script PHP, antara lain :

```
<?  
?>
```

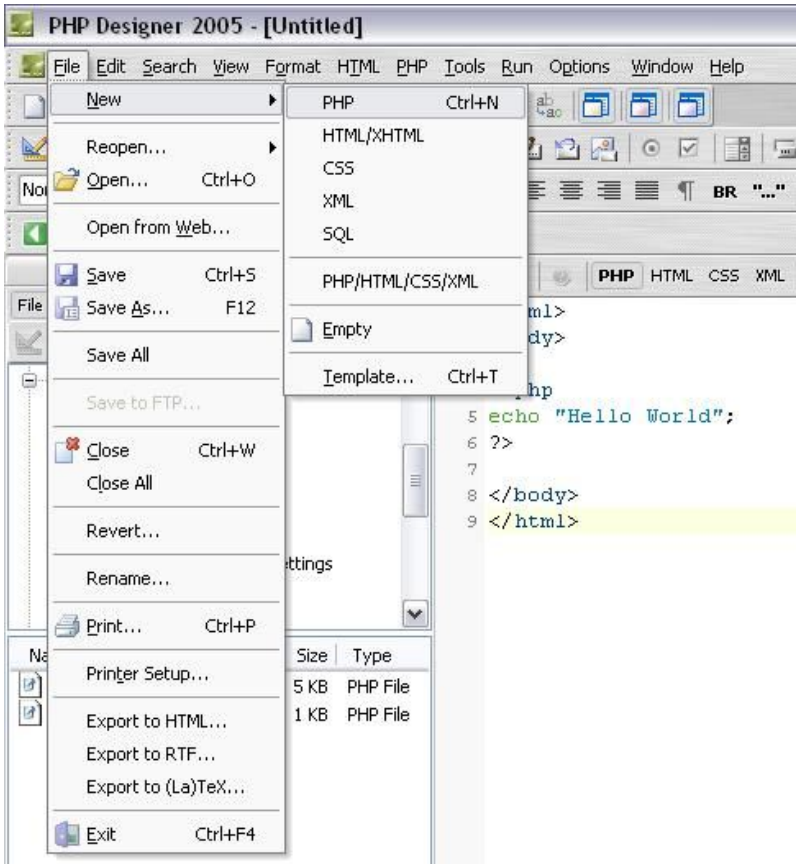
dan

```
<?php  
?>
```

tetapi untuk kekompatibilitas maksimum, maka penulis merekomendasikan jenis `<?php ... ?>` yang sebaiknya digunakan di dalam pemrograman PHP, dan juga jenis standar yang digunakan di dalam buku ini.

Sebuah file PHP secara normal akan berisikan tag HTML, seperti sebuah file HTML, dan ditambah beberapa kode script PHP. Sekarang kita mulai mencoba program pertama kita, untuk itu ikuti langkah-langkah berikut :

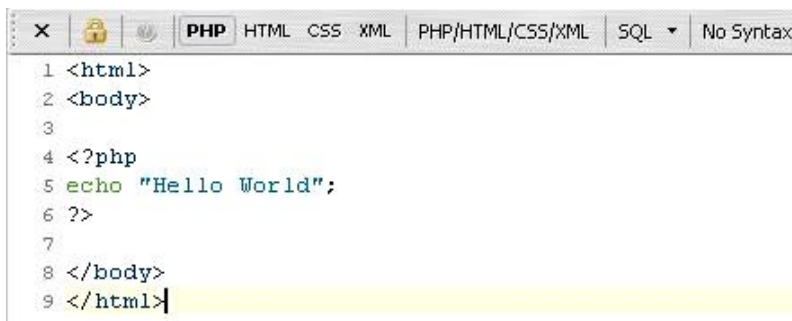
1. Jalankan PHP Designer.
2. Sebelum memulai menuliskan kode program, pastikan anda telah membuat folder baru di dalam folder `C:\AppServ\www\`, misal : `C:\AppServ\www\1`. Di dalam folder inilah file PHP yang akan kita buat disimpan.
3. Lalu pada menu utama PHP Designer, pilih File-New-PHP.



4. Setelah muncul halaman kosong, selanjutnya tuliskan script berikut :

```
<html>
<body>
<?php
echo "Hello World";
?>
</body>
</html>
```

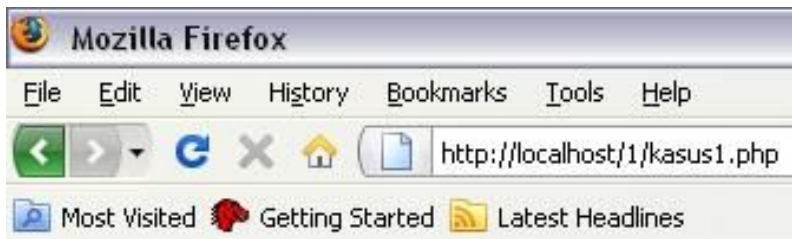
Listing dalam bentuk tampilan pada PHP Designer, kira-kira akan seperti :



```
1 <html>
2 <body>
3
4 <?php
5 echo "Hello World";
6 ?>
7
8 </body>
9 </html>
```

simpan ke direktori C:\AppServ\www\1 dengan nama kasus1 (file akan disimpan otomatis sebagai kasus1.php).

- 5. Untuk eksekusi, jalankan browser web anda, misal Mozilla Firefox, lalu tuliskan pada dialog browser, <http://localhost/1/kasus1.php> maka akan tampil keluaran berikut :



Hello World

- 6. Ok, sekarang kita telah berhasil mengeksekusi kasus pertama kita. Program yang baru kita buat adalah program untuk menampilkan kata "Hello World" pada web anda, dan statement yang bertugas untuk menampilkan string tersebut adalah statement echo yang diikuti dengan tanda " " .

2.2 Variabel dalam PHP

Variabel digunakan untuk menampung data sementara, baik string atau bilangan, yang selanjutnya data tersebut diolah dengan cara tertentu untuk menghasilkan suatu output.

Di dalam pemrograman PHP, variabel tidak perlu terlebih dahulu dideklarasikan seperti beberapa bahasa pemrograman yang membutuhkan kompiler untuk pengekseskuan programnya. PHP akan secara otomatis menentukan tipe data variabel yang anda

gunakan ke tipe data yang sesuai, tergantung dari sekumpulan nilai yang diberikan oleh perancang program. Semua nama variabel di dalam PHP harus dimulai dengan tanda \$ (baca : dollar). Misalnya :

\$Nama_Pendaftar = Fadlisyah;

ket : antara tanda \$ dan nama variabel tidak ada spasi, dan setiap akhir dari suatu sintaks selalu diberikan tanda ;

2.3 Operator dalam PHP

Berbagai operator yang digunakan di dalam PHP meliputi :

Operator Aritmatika

Operator	Deskripsi	Contoh	Hasil
+	Penambahan	x=2 x+2	4
-	Pengurangan	x=2 5-x	3
*	Perkalian	x=4 x*5	20
/	Pembagian	15/5 5/2	3 2.5
%	Modulus (Sisa Pembagian)	5%2 10%8 10%2	1 2 0
++	Increment	x=5 x++	x=6
--	Decrement	x=5 x--	x=4

Operator Assignment

Operator	Contoh	Ekivalen dengan
=	x=y	x=y
+=	x+=y	x=x+y
-=	x-=y	x=x-y
=	x=y	x=x*y
/=	x/=y	x=x/y
.=	x.=y	x=x.y
%=	x%=y	x=x%y

Operator Perbandingan

Operator	Deskripsi	Contoh
==	Sama dengan	5==8 (false)
!=	Tidak sama dengan	5!=8 (true)
>	Lebih besar dari	5>8 (false)
<	Lebih kecil dari	5<8 (true)
>=	Lebih besar dari atau sama dengan	5>=8 (false)
<=	Lebih kecil dari atau sama dengan	5<=8 (true)

Operator Logika

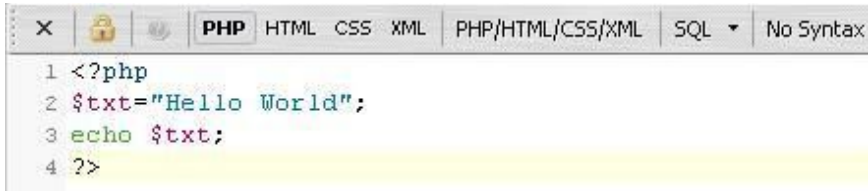
Operator	Deskripsi	Contoh
&&	AND	x=6 y=3 (x < 10 && y > 1) (true)
	OR	x=6 y=3 (x==5 y==5) (false)
!	NOT	x=6 y=3 !(x==y) (true)

2.4 Berbagai Operasi pada String

Variabel string digunakan untuk menyimpan berbagai karakter string yang selanjutnya dimanipulasikan untuk menghasilkan output tertentu. Untuk contoh perhatikan listing program berikut :

```
<?php
$txt="Hello World";
echo $txt;
?>
```

atau dalam bentuk tampilan PHP Designer berupa :



```

1 <?php
2 $txt="Hello World";
3 echo $txt;
4 ?>

```

akan menghasilkan output :



Hello World

string "Hello World."

Sekarang kita mencoba untuk memanipulasi string di atas tersebut dengan berbagai fungsi berikut.

Operator penyambung (Concatenation). Untuk menyatukan dua buah string yang terpisah menjadi satu kalimat, kita dapat menggunakan operator penyambung atau (.)⁵. Contoh kasus, perhatikan kode di bawah ini :

```

1 <?php
2 $txt1="Hello World";
3 $txt2="1234";
4 echo $txt1 . " " . $txt2;
5 ?>

```

kode di atas akan menghasilkan output :



Hello World 1234

⁵ Baca dot

string "Hello World 1234."

Fungsi `strlen()`. Fungsi `strlen()` digunakan untuk menentukan panjang string input. Contoh penggunaannya, perhatikan kode berikut :

```
1 <?php
2 echo strlen("Hello world!");
3 ?>
```

kode di atas akan menghasilkan output :

12

panjang string dari "Hello World!."

Fungsi `strpos()`. Fungsi `strpos()` digunakan untuk mencari sebuah string atau karakter di dalam sekumpulan string induk, dan jika ditemukan, maka program akan menghasilkan sebuah nilai yang berupa posisi awal string yang dicari pada sekumpulan string induk. Sebaliknya program akan mengembalikan nilai `false`, jika tidak ditemukannya satupun string atau karakter yang sesuai. Perhatikan contoh berikut :

```
<?php
echo strpos("Hello world!","world");
?>
```

kode di atas akan menghasilkan nilai :

6

posisi awal world di dalam Hello world!⁶

∞

⁶ file:///C:/Downloads/php_string.asp.htm

This page is intentionally left blank

Bab 3

PERNYATAAN KONDISIONAL

3.1 if ... else ... dalam PHP

Pernyataan if, else if, dan pernyataan else digunakan untuk melakukan suatu aksi yang berbeda berdasarkan suatu kondisi tertentu. Ada berbagai bentuk pernyataan If ... Else, di antaranya :
Pernyataan **if ... else ...**,

Format sintaks :

```
if {Kondisi}
    Eksekusi kode jika True
else {Kondisi}
    Eksekusi kode alternatif jika False
```

Contoh kasus :

```
<html>
<body>
<?php
$d=date("D");
if ($d=="Fri")
    echo "Have a nice weekend!";
else
    echo "Have a nice day!";
?>
</body>
</html>
```

Hasil eksekusi :

Jika hari ini jumat, maka hasil eksekusi program di atas adalah "Have a nice weekend." Dan sebaliknya "Have a nice day!."

Jika kode yang harus dieksekusi berdasarkan suatu kondisi, memiliki beberapa baris, maka beri pasangan tanda {} pada awal dan akhir baris kode yang akan dieksekusi. Contoh :

```
<html>
<body>
<?php
$d=date("D");
if ($d=="Fri")
{
echo "Hello!<br />";
echo "Have a nice weekend!";
echo "See you on Monday!";
}
?>
</body>
</html>
```

Pernyataan **if ... else if ... else ...**,

Jika kita dihadapkan pada berbagai kondisi yang lebih dari satu kondisi bernilai true, maka kita dapat menggunakan pernyataan if ... else if ... else

Format sintaks :

```
if {Kondisi}
    Eksekusi kode jika True
else if {Kondisi}
    Eksekusi kode jika True
else {Kondisi}
    Eksekusi kode alternatif jika False
```

Contoh :

```
<html>
```

```
<body>
<?php
$d=date("D");
if ($d=="Fri")
    echo "Have a nice weekend!";
elseif ($d=="Sun")
    echo "Have a nice Sunday!";
else
    echo "Have a nice day!";
?>
</body>
</html>
```

3.2 Pernyataan switch dalam PHP

Untuk menghindari kekompleksitasan penggunaan fungsi If, maka diperkenalkanlah fungsi Switch. Fungsi Switch digunakan untuk menentukan salah satu keputusan dari berbagai kondisi tertentu.

Format sintaks :

```
switch (ekspresi)
{
    case label1:
        eksekusi kode jika ekspresi = label1;
        break;
    case label2:
        eksekusi kode jika ekspresi = label2;
        break;
    default:
        eksekusi kode jika ekspresi berbeda dengan label1 atau label2;
}
```

Contoh :


```
<html>
<body>

<?php
switch ($x)
{
case 1:
    echo "Number 1";
    break;
case 2:
    echo "Number 2";
    break;
case 3:
    echo "Number 3";
    break;
default:
    echo "No number between 1 and 3";
}
?>

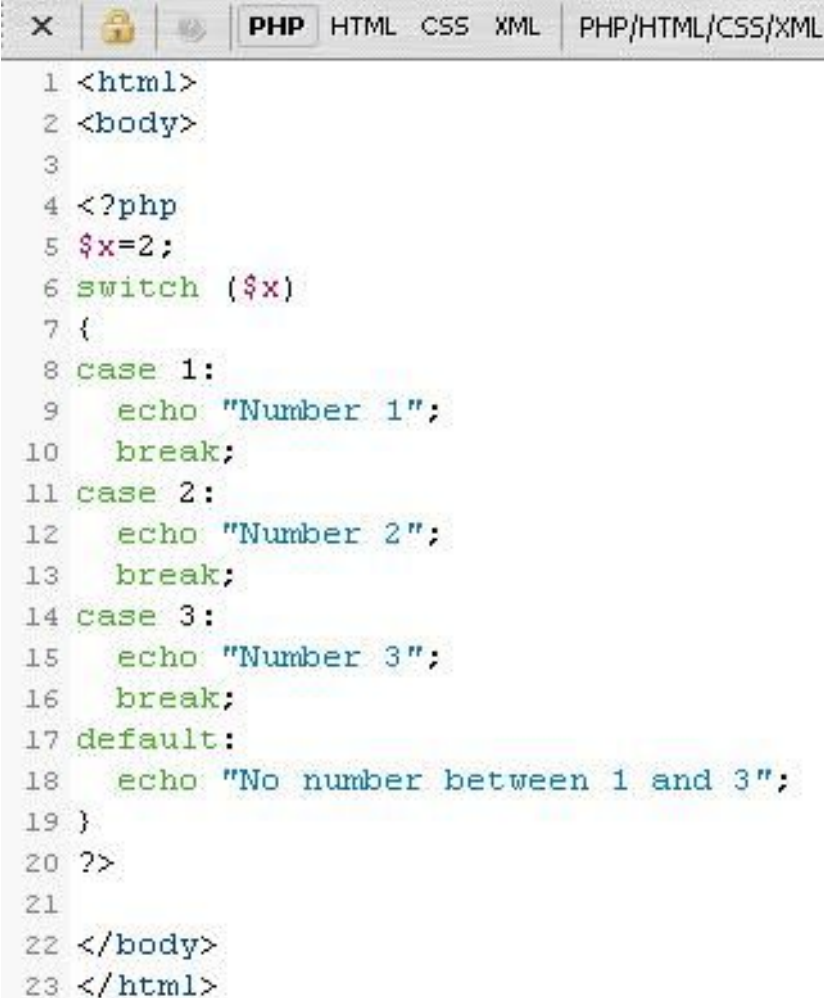
</body>
</html>
```

Hasil output dari dari program di atas adalah : No number between 1 and 3. Sebaliknya dengan memodifikasi sedikit program di atas, sehingga program keseluruhan akan berupa :

```
<html>
<body>
<?php
$x=2;
switch ($x)
{
case 1:
    echo "Number 1";
    break;
case 2:
    echo "Number 2";
    break;
case 3:
    echo "Number 3";
```

```
break;
default:
    echo "No number between 1 and 3";
}
?>
</body>
</html>
```

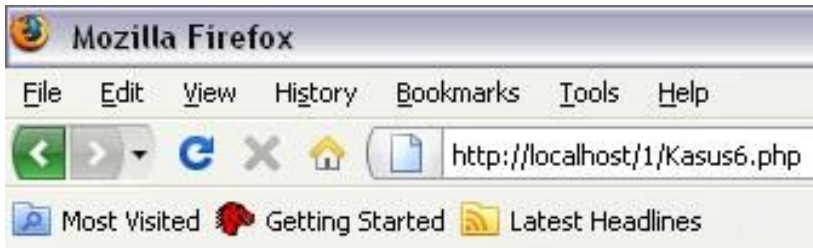
atau dalam bentuk tampilan pada PHP Designer berupa :



The screenshot shows a code editor window with a title bar containing icons for close, lock, and refresh, and tabs for PHP, HTML, CSS, XML, and a combined PHP/HTML/CSS/XML tab. The code is as follows:

```
1 <html>
2 <body>
3
4 <?php
5 $x=2;
6 switch ($x)
7 {
8 case 1:
9     echo "Number 1";
10 break;
11 case 2:
12     echo "Number 2";
13 break;
14 case 3:
15     echo "Number 3";
16 break;
17 default:
18     echo "No number between 1 and 3";
19 }
20 ?>
21
22 </body>
23 </html>
```

hasil eksekusi program adalah :



Number 2

∞

Bab 4

PERULANGAN

4.1 Perulangan dalam PHP

Pernyataan perulangan digunakan untuk mengeksekusi satu atau lebih baris kode sebanyak beberapa kali. Di dalam PHP ada beberapa bentuk pernyataan perulangan, di antaranya : while, do while, for, dan foreach.

4.2 while

Pernyataan while melakukan eksekusi sebarisan kode program jika dan sepanjang kondisi yang telah ditentukan tetap terpenuhi.

Sintaks :

```
while {kondisi}
    kode yang akan dieksekusi;
```

Contoh :

```
<html>
<body>
<?php
$i=1;
while($i<=5)
{
    echo "The number is " . $i . "<br />";
    $i++;
}
?>
</body>
</html>
```

hasil eksekusi :

```
The number is 1
The number is 2
The number is 3
```

The number is 4
The number is 5

4.3 do ... while

Pernyataan do ... while akan mengeksekusi sebarisan kode sedikitnya sekali, kemudian mengulang blok perulangan tersebut sepanjang kondisi yang ditentukan tetap terpenuhi.

Sintaks :

```
do
    kode yang dieksekusi
while {kondisi}
```

Contoh :

```
<html>
<body>
<?php
$i=0;
do
{
    $i++;
    echo "The number is " . $i . "<br />";
}
while ($i<5);
?>
</body>
</html>
```

hasil eksekusi :

The number is 1
The number is 2
The number is 3
The number is 4
The number is 5

4.4 Pernyataan for

Bentuk lain penggunaan perulangan di dalam PHP adalah pernyataan for.

Sintaks :

```
for (nilai awal; kondisi; increment)
{
    kode yang akan dieksekusi
}
```

Contoh :

```
<html>
<body>
<?php
for ($i=1; $i<=5; $i++)
{
    echo "Hello World!<br />";
}
?>
</body>
</html>
```

akan menghasilkan output "Hello World" sebanyak lima kali.

4.5 Pernyataan foreach

Pernyataan foreach bertugas mengulangi sebuah array sebanyak jumlah array yang memiliki nilai.

Sintaks :

```
foreach (array as value)
{
    kode yang akan dieksekusi
}
```

Contoh :

```
<html>
<body>
<?php
$arr=array("one", "two", "three");

foreach ($arr as $value)
{
    echo "Value: " . $value . "<br />";
}
?>
</body>
</html>
```

dan output yang dihasilkan adalah :

```
Value: one
Value: two
Value: three
```

This page is intentionally left blank

Bab 5

ARRAY

5.1 Array dalam PHP

Sebuah array dapat menyimpan satu atau lebih nilai-nilai di dalam satu variabel. Di dalam PHP array dapat digolongkan menjadi tiga jenis : array numerik, array asosiatif, dan array multidimensi.

5.2 Array Numerik

Array numerik merupakan tipe array yang menyimpan masing-masing elemen berdasarkan indeks numerik tertentu. PHP memungkinkan beberapa cara dalam menyajikan array numerik tersebut, di antaranya :

Cara 1

```
$names = array("Peter","Quagmire","Joe");
```

Cara 2

```
$names[0] = "Peter";
```

```
$names[1] = "Quagmire";
```

```
$names[2] = "Joe";
```

Cara 1 mempersilahkan PHP menentukan indeks secara otomatis, dan sementara cara 2, indeks ditentukan dengan campur tangan langsung oleh user.

Contoh program yang berhubungan dengan array numerik adalah :

```
<?php
$names[0] = "Peter";
$names[1] = "Quagmire";
$names[2] = "Joe";
echo $names[1] . " and " . $names[2] .
" are " . $names[0] . "'s neighbors";
?>
```

dengan output :

Quagmire and Joe are Peter's neighbors.

5.3 Array Asosiatif

Pada array asosiatif, masing-masing indeks dikaitkan langsung dengan nilai yang dikandunginya array tersebut.

Misalkan :

```
$ages = array("Peter"=>32, "Quagmire"=>30, "Joe"=>34);  
dapat juga disajikan ke dalam bentuk :  
$ages['Peter'] = "32";  
$ages['Quagmire'] = "30";  
$ages['Joe'] = "34";
```

dan contoh program yang berkenaan ditampilkan sebagai :

```
<?php  
$ages['Peter'] = "32";  
$ages['Quagmire'] = "30";  
$ages['Joe'] = "34";  
echo "Peter is " . $ages['Peter'] . " years old."  
?>
```

output yang dihasilkan adalah "Peter is 32 years old."

5.4 Array Multidimensi

Di dalam array multidimensi, masing-masing elemen di dalam array utama dapat juga berfungsi sebagai array, dan masing-masing elemen di dalam sub-array juga berfungsi sebagai array, dan seterusnya.

Ambil kasus berikut :

```
$families = array  
(  
  "Griffin"=>array  
  (  
    "Peter",  
    "Lois",  
    "Megan"  
  ),  
  "Quagmire"=>array  
  (  
    "Glenn"  
  ),  
)
```

```
"Brown"=>array
(
    "Cleveland",
    "Loretta",
    "Junior"
)
```

atau dengan pernyataan yang setara, dapat dituliskan sebagai berikut :

```
Array
(
    [Griffin] => Array
        (
            [0] => Peter
            [1] => Lois
            [2] => Megan
        )
    [Quagmire] => Array
        (
            [0] => Glenn
        )
    [Brown] => Array
        (
            [0] => Cleveland
            [1] => Loretta
            [2] => Junior
        )
)
```

dan jika kita ingin menampilkan salah satu nilai array di atas, maka dapat ditambahkan kode :

```
echo "Is " . $families['Griffin'][2] .
```

```
" a part of the Griffin family?";
```

atau dalam bentuk kode PHP utuh, berupa :

```
1 <?php
2 $families = array
3 {
4   "Griffin"=>array
5   {
6     "Peter",
7     "Lois",
8     "Megan"
9   },
10  "Quagmire"=>array
11  {
12    "Glenn"
13  },
14  "Brown"=>array
15  {
16    "Cleveland",
17    "Loretta",
18    "Junior"
19  }
20 };
21 echo "Is " . $families['Griffin'][2] .
22 " a part of the Griffin family?";
23 ?>
```

program akan menghasilkan output :
Is Megan a part of the Griffin family?.

∞

Bab 6

FUNGSI & FORM

7.1 Fungsi dalam PHP

Kekuatan PHP sebenarnya adalah karena fungsi yang terdapat padanya. Lebih dari 700 fungsi yang dibangun di dalamnya. Pada bab kali ini, kita akan mencoba membahas bagaimana membangun fungsi-fungsi tersebut secara sederhana.

Sintaks :

```
Functions nama_fungsi()
{
}
```

Untuk contoh perhatikan program di bawah ini.

```
<html>
<body>
<?php
function writeMyName()
{
    echo "Fadlisyah";
}
writeMyName();
?>
</body>
</html>
```

Program akan menghasilkan output "Fadlisyah."

Contoh lainnya adalah :

```
<html>
<body>
<?php
function writeMyName()
{
    echo "Fadlisyah";
}
```

```
echo "Hello world!<br />";
echo "Apakah";
writeMyName();
echo ".<br />Apakah, ";
writeMyName();
echo " adalah nama saya.";
?>
</body>
</html>
```

akan menghasilkan output :
Hello world!
Apakah Fadlisyah.
Apakah, Fadlisyah adalah nama saya.

Beberapa contoh program di atas, menerapkan penggunaan fungsi secara sederhana, dan hanya mampu menuliskan berbagai string yang telah ditentukan. Agar lebih dinamis, perlu ditambahkan satu atau lebih parameter pada fungsi yang dideklarasikan, dan untuk itu perhatikan contoh program berikut :

```
<html>
<body>
<?php
function writeMyName($fname)
{
    echo $fname . " Informatika.<br />";
}
echo "My name is ";
writeMyName("Bustami");
echo "My name is ";
writeMyName("Rizal");
echo "My name is ";
writeMyName("Dahlan");
?>
</body>
</html>
```

akan menghasilkan ouput :
My name is Bustami Informatika.
My name is Rizal Informatika.
My name is Dahlan Informatika.

Contoh lainnya, fungsi menggunakan dua buah parameter.

```
<html>
<body>
<?php
function writeMyName($fname,$punctuation)
{
    echo $fname . "Unimal" . $punctuation . "<br />";
}
echo "My name is ";
writeMyName("Risawandi", ".");
echo "My name is ";
writeMyName("Nurdin", "!");
echo "My name is ";
writeMyName("Ikhwanus", "...");
?>
</body>
</html>
```

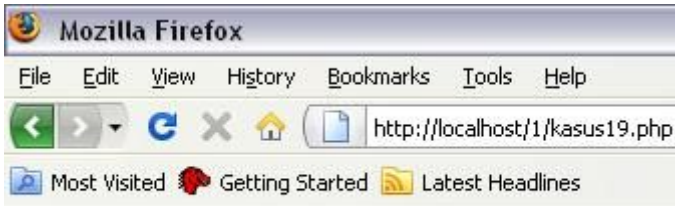
output yang dihasilkan :

```
My name is RisawandiUnimal.
My name is NurdinUnimal!
My name is IkhwanusUnimal...
```

Fungsi dapat juga digunakan untuk mengembalikan suatu nilai yang dikalkulasikan di dalam tubuh fungsi, keluar dari fungsi. Contoh

```
:
<html>
<body>
<?php
function add($x,$y)
{
    $total = $x + $y;
    return $total;
}
echo "76 + 55 = " . add(76,55);
?>
</body>
</html>
```

output yang dihasilkan :



76 + 55 = 131

7.2 Bekerja dengan Beberapa Form

Pada pembahasan kita kali ini, kita akan mencoba membuat program antara form menggunakan PHP.

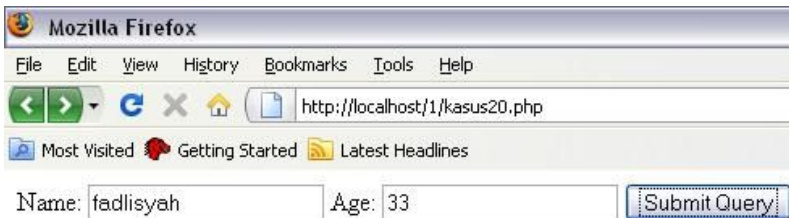
Tulis program berikut :

```
<html>
<body>

<form action="welcome.php" method="post">
Name: <input type="text" name="name" />
Age: <input type="text" name="age" />
<input type="submit" />
</form>

</body>
</html>
```

dan simpan dengan nama file kasus20.php. Program di atas akan menghasilkan output berupa :



Selanjutnya buat program ke-2, dan simpan dengan nama Welcome.php.

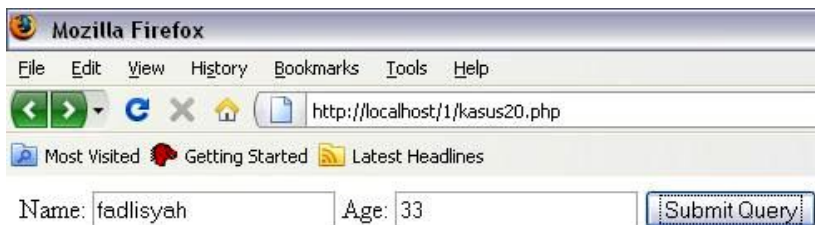
```
<html>
<body>

Welcome <?php echo $_POST["name"]; ?>.<br />
You are <?php echo $_POST["age"]; ?> years old.
```

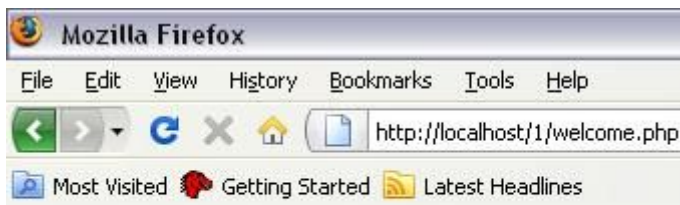
```
</body>
```

```
</html>
```

Sekarang coba eksekusi kembali program kasus20.php.



Setelah kita menekan button Submit Query, maka akan keluar tampilan :



Welcome fadlisyah.

You are 33 years old.

7.3 \$_GET

Fungsi `&_GET` digunakan untuk mengambil suatu nilai dari suatu form pengirim dengan menggunakan metode="get." Informasi kiriman dari form akan terlihat oleh setiap orang yang mengakses dan jumlah informasi pengiriman memiliki keterbatasan, tidak lebih dari 100 karakter. Contoh :

```
<form action="welcome.php" method="get">  
Name: <input type="text" name="fname" />  
Age: <input type="text" name="age" />  
<input type="submit" />  
</form>
```

simpan dengan file kasus21.php. Lalu buat program ke-2 sebagai berikut :

```
<html>
```

```
<body>
```

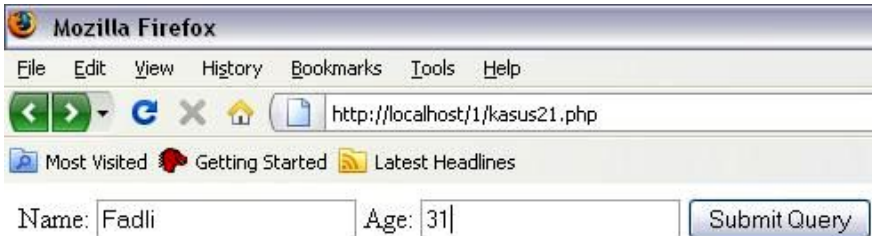
```
Welcome <?php echo $_GET["name"]; ?>.<br />
```

```
You are <?php echo $_GET["age"]; ?> years old.
```



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

dan simpan dengan nama `welcome.php`.
setelah dieksekusi, tampilan program akan berupa :



setelah menekan button Submit Query, pada dialog akan muncul tulisan :

`http://localhost/1/welcome.php?fname=Fadli&age=31.`

7.4 \$_POST

Fungsi `&_GET` digunakan untuk mengambil suatu nilai dari suatu form pengirim dengan menggunakan metode="post." Informasi kiriman dari form tidak terlihat oleh siapapun yang mengakses dan jumlah informasi pengiriman memiliki tidak terbatas. Contoh :

```
<form action="welcome.php" method="get">
Name: <input type="text" name="fname" />
Age: <input type="text" name="age" />
<input type="submit" />
</form>
```

simpan dengan nama file `kasus22.php`. Lalu modifikasi program `Welcome.php` di atas sehingga berbentuk :

```
<html>
<body>

Welcome <?php echo $_POST["fname"]; ?>!<br />
You are <?php echo $_POST["age"]; ?> years old.

</body>
</html>
```

ketika program dieksekusi dan user telah menginput data, URL yang terlihat di atas adalah :

`http://localhost/1/welcome.php`

Bab 7

PEMROGRAMAN DATABASE MENGUNAKAN MySQL

7.1 Pengenalan MySQL

MySQL merupakan sebuah sistem database open-source yang saat ini sangat populer digunakan. Berbagai data di dalam MySQL disimpan ke dalam objek database yang disebut tabel. Sebuah tabel akan terdiri dari berbagai entri yang saling berhubungan yang disajikan dalam bentuk hubungan kolom dan baris.

Sekarang, coba buat tabel dengan struktur berikut :

Nama	Varchar	20	
Nip	Varchar	9	*
Mata_Kuliah	Varchar	20	

Dengan menggunakan database MySQL.

Isi tabel dengan data atau record berikut :

Nama	Nip	Mata_Kuliah
Fadlisyah	132321540	Pengolahan Citra
Rizal	132321541	Pemrograman Web
Arhami	132321542	Kecerdasan Buatan
Bustami	132321543	Metode Numerik
Nurdin	132321544	Basis Data

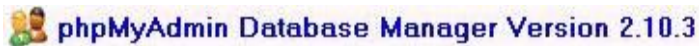
Untuk membuat dan mengisi tabel menggunakan MySQL, ikuti langkah-langkah berikut :

1. Jalankan browser seperti Mozilla Firefox.

2. Lalu pada dialog URL, tuliskan `http://localhost/` dan akhiri dengan menekan tombol enter pada keyboard.
3. Setelah muncul tampilan :



klik sekali pada



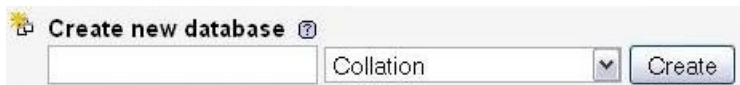
dan isi :

User Name : root, dan Password : root.

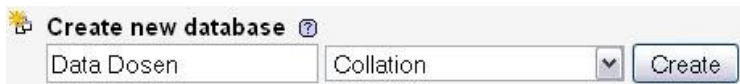
4. Tidak lama kemudian muncul tampilan :



dan pada item,



isikan misal : Data_Dosen, seperti tampilan berikut :



lalu tekan button Create. Tidak lama kemudian muncul tampilan berikut :



- 5. Isikan nama tabel, misal Tabel_Dosen, dengan jumlah field 3 buah.



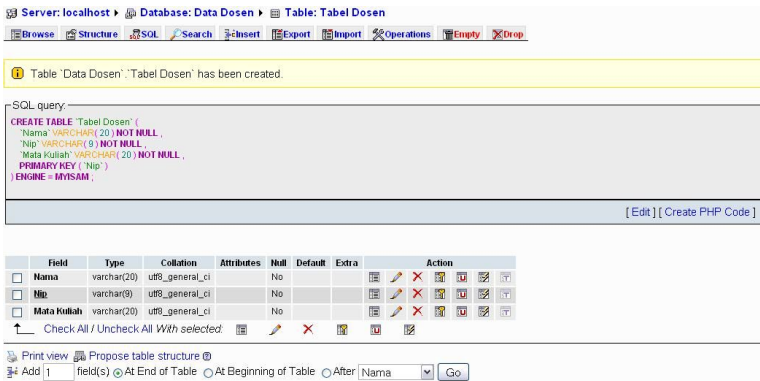
lalu klik button Go.

- 6. Selanjutnya tentukan nama-nama field yang diperlukan,



tipe field, dan panjang field. Pada tampilan di atas hanya kolom Field, Type, dan Length/Values yang ditentukan, dan pada Radiobutton Primary untuk Field Nip diklik sekali untuk memberitahukan kepada MySQL, bahwa Nip adalah kunci utama (Primary Key).

Jika telah selesai mendesain struktur tabel, silahkan klik button Save. Tidak lama kemudian muncul tampilan :



dengan kode pada SQL query :

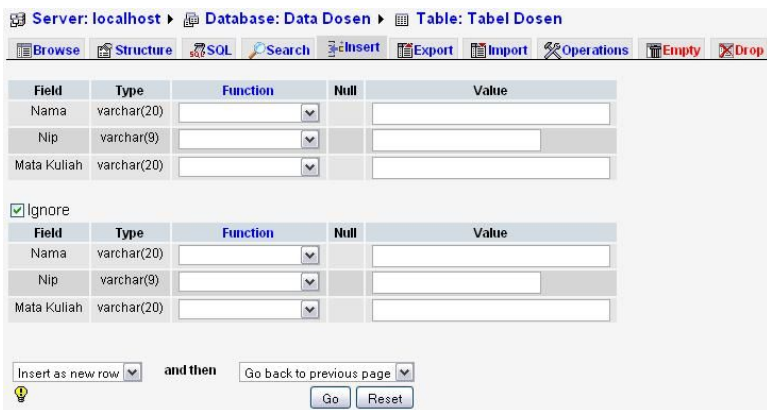
```

CREATE TABLE `Tabel Dosen` (
`Nama` VARCHAR( 20 ) NOT NULL ,
`Nip` VARCHAR( 9 ) NOT NULL ,
`Mata_Kuliah` VARCHAR( 20 ) NOT NULL ,
PRIMARY KEY ( `Nip` )
) ENGINE = MYISAM ;
    
```

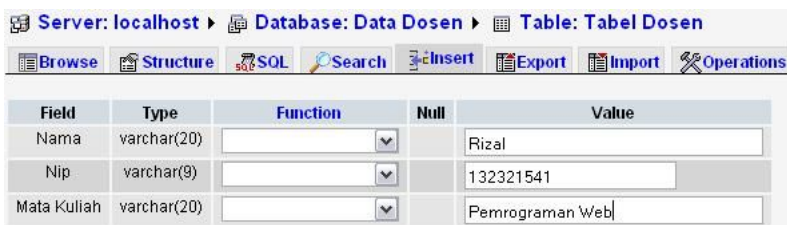
dan struktur tabel seperti tampilan berikut.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> Nama	varchar(20)	utf8_general_ci		No			
<input type="checkbox"/> Nip	varchar(9)	utf8_general_ci		No			
<input type="checkbox"/> Mata Kuliah	varchar(20)	utf8_general_ci		No			

- Untuk mengisi data dalam tabel, klik pada tab Insert. Setelah muncul tampilan :



Isi data yang diperlukan pada sisi kanan bagian atas kolom Value. Misal :



dan akhiri dengan menekan button Go. Ulangi terus-menerus langkah ke-7 sampai keseluruhan record terpenuhi.

Nama	Nip	Mata Kuliah
Fadlisyah	132321540	Pengolahan Citra
Rizal	132321541	Pemrograman Web
Arhami	132321542	Kecerdasan Buatan
Bustami	132321543	Metode Numerik
Nurdin	132321544	Basis Data

Dalam bentuk tampilan pada browser keseluruhan record direpresentasikan seperti :



- Tuliskan kode program berikut, menggunakan PHP Designer.

```
<?php
$con = mysql_connect("localhost","root","root");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
```

```

    }
    mysql_select_db("data_dosen", $con);

    $result = mysql_query("SELECT * FROM tabel_dosen");

    while($row = mysql_fetch_array($result))
    {
        echo $row['Nama'] . " " . $row['Nip'] . " " .
        $row['Mata_Kuliah'];
        echo "<br />";
    }
    mysql_close($con);
?>

```

9. Output program, akan berupa tampilan :

```

Fadlisyah 132321540 Pengolahan Citra
Rizal 132321541 Pemrograman Web
Arhami 132321542 Kecerdasan Buatan
Bustami 132321543 Metode Numerik
Nurdin 132321544 Basis Data

```

7.2 Koneksi Database MySQL

Sebelum kita dapat mengakses data di dalam database, terlebih dahulu kita harus melakukan koneksi ke database. Di dalam PHP, koneksi ke database dapat dilakukan melalui perintah :

```

mysql_connect(servername,username,password);

```

untuk kasus kita, misalnya :

```

mysql_connect("localhost","root","root");

```

di mana, nama server yang kita gunakan adalah localhost, username = root, dan password = root.

Contoh program :

```

<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// Kode program
?>

```

atau dengan modifikasi program di atas,

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
```

```
// Kode program
```

```
mysql_close($con);
?>
```

Keterangan : `mysql_close($con);` berfungsi untuk menutup koneksi database.

7.3 CREATE DATABASE

Instruksi `CREATE DATABASE` digunakan untuk menciptakan sebuah database di dalam MySQL. Kasus berikut adalah contoh penerapan instruksi `CREATE DATABASE` untuk menciptakan database `my_db`.

```
<?php
$con = mysql_connect("localhost","root","root");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

if (mysql_query("CREATE DATABASE my_db",$con))
{
    echo "Database created";
}
else
{
    echo "Error creating database: " . mysql_error();
}

mysql_close($con);
?>
```


7.4 CREATE TABLE

Pernyataan CREATE TABLE digunakan untuk menciptakan sebuah tabel di dalam database pada MySQL.

Sintaks :

```
CREATE TABLE nama_tabel
(
nama_field1 tipe_data(lebar_data),
nama_field2 tipe_data(lebar_data),
nama_field3 tipe_data(lebar_data),
...
)
```

Sebelum melakukan CREATE TABLE, kita terlebih dahulu harus menambahkan fungsi `mysql_query()` untuk pengecekan status dari koneksi database yang telah dilakukan menggunakan perintah : `mysql_connect(servername,username,password);`

Pada contoh berikut, kita mencoba membuat tabel biodata, dengan tiga field, `nama_awal`, `nama_akhir`, dan `usia`.

```
<?php
$con = mysql_connect("localhost","root","root");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// Create database
if (mysql_query("CREATE DATABASE my_db",$con))
{
    echo "Database created";
}
else
{
    echo "Error creating database: " . mysql_error();
}

// Create table
mysql_select_db("my_db", $con);
$sql = "CREATE TABLE biodata
(
nama_awal varchar(15),
nama_akhir varchar(15),
```

```
usia int
)";

// Execute query
mysql_query($sql,$con);

mysql_close($con);
?>
```

Jika eksekusi sukses, lihat pada folder C:\AppServ\MySQL\data, akan terdapat folder baru  my_db .

Sekarang bagaimana kita dapat menentukan field kunci, melalui kode program PHP?. Untuk itu, perhatikan penggal program berikut :

```
$sql = "CREATE TABLE biodata
(
NIP int NOT NULL AUTO_INCREMENT,
PRIMARY KEY(NIP),
nama_awal varchar(15),
nama_akhir varchar(15),
usia int
)";
mysql_query($sql,$con);
```

di mana field NIP memiliki sifat unik dan auto increment. Untuk keseluruhan kode programnya disajikan berikut :

```
<?php
$con = mysql_connect("localhost","root","root");
if (!$con)
{
die('Could not connect: ' . mysql_error());
}

// Create database
if (mysql_query("CREATE DATABASE my_db",$con))
{
echo "Database created";
}
else
{
```

```
echo "Error creating database: " . mysql_error();
}

// Create table dan menentukan Primary key
mysql_select_db("my_db", $con);
$sql = "CREATE TABLE biodata
(
NIP int NOT NULL AUTO_INCREMENT,
PRIMARY KEY(NIP),
nama_awal varchar(15),
nama_akhir varchar(15),
usia int
)";

// Execute query
mysql_query($sql,$con);

mysql_close($con);
?>
```

7.5 INSERT INTO

Perintah INSERT INTO digunakan untuk menambahkan record baru ke dalam suatu tabel database.

Pernyataan INSERT INTO, dapat ditulis ke dalam 2 bentuk :

```
INSERT INTO table_name
VALUES (value1, value2, value3,...)
```

atau

```
INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)
```

dan lagi-lagi instruksi INSERT INTO harus didahului oleh fungsi `mysql_query()`. Fungsi `mysql_query()` diperlukan untuk mengirimkan query atau perintah ke koneksi MySQL.

Masih menggunakan database `my_db` dan tabel `biodata`, kita mencoba merancang sebuah program menginput data menggunakan instruksi INSERT INTO.

```
<?php
$con = mysql_connect("localhost","root","root");
if (!$con)
{
die('Could not connect: ' . mysql_error());
```

```
}  
  
mysql_select_db("my_db", $con);  
  
mysql_query("INSERT INTO biodata (NIP, nama_awal, nama_akhir,  
usia)  
VALUES ('132321555', 'Quraish', 'Shihab', '35')");  
  
mysql_query("INSERT INTO biodata (NIP, nama_awal, nama_akhir,  
usia)  
VALUES ('132321556', 'Alwi', 'Shihab', '37')");  
  
mysql_close($con);  
?>
```

7.6 Menginput Data dari Form ke Database

Pada sub materi kali ini, kita mencoba membuat program yang mampu menyimpan data yang dimasukkan melalui sebuah form. Untuk itu ikuti langkah-langkah berikut :

Melalui PHP Designer, tuliskan kode program berikut :

```
<html>  
<body>  
<form action="insert.php" method="post">  
NIP :<input type="text" name="NIP" />  
Nama awal: <input type="text" name="nama_awal" />  
Nama akhir: <input type="text" name="nama_akhir" />  
Usia: <input type="text" name="usia" />  
<input type="submit" />  
</form>  
</body>  
</html>
```

dan simpan dengan nama file, misal : kasus28.php. Lalu buat program ke-2 dengan nama insert.php.

```
<?php  
$con = mysql_connect("localhost", "root", "root");  
if (!$con)  
{  
    die('Could not connect: ' . mysql_error());  
}
```

```
mysql_select_db("my_db", $con);
```

```
$sql="INSERT INTO biodata (NIP, nama_awal, nama_akhir, usia)
VALUES
('$_POST[NIP]','$_POST[nama_awal]','$_POST[nama_akhir]','$_POST[
usia]')";
```

```
if (!mysql_query($sql,$con))
{
    die('Error: ' . mysql_error());
}
echo "1 record added";
```

```
mysql_close($con)
?>
```

Tampilan eksekusi kasus28.php, kira-kira berupa :



lalu misalkan kita input nilai :

```
NIP          : 132333333
Nama awal    : Fadli
Nama akhir   : Muhammad
Usia        : 50
```

Dan akhiri dengan menekan button Submit Query, maka akan muncul pesan :

```
1 record added,
pernyataan bahwa data yang diinput telah berhasil disimpan.
```

7.7 SELECT

Pernyataan SELECT digunakan untuk memilih sekumpulan data dari sebuah database.

Sintaks :

```
SELECT column_name(s)
FROM table_name
```

Kode program berikut akan menampilkan seluruh data pada nama_awal dan nama_akhir saja dalam tabel biodata.

```
<?php
```

```
$con = mysql_connect("localhost","root","root");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM biodata");

while($row = mysql_fetch_array($result))
{
    echo $row['nama_awal'] . " " . $row['nama_akhir'];
    echo "<br />";
}

mysql_close($con);
?>
```

Hasil eksekusi yang diperoleh :

Fadli Muhammad

Salbiah Hanum

Jika kita ingin menampilkan data-data yang dipilih di dalam tabel HTML, maka kita dapat membuat program seperti berikut :

```
<?php
$con = mysql_connect("localhost","root","root");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM biodata");

echo "<table border='1'>
<tr>
<th>nama_awal</th>
<th>nama_akhir</th>
</tr>";

while($row = mysql_fetch_array($result))
```

```

{
echo "<tr>";
echo "<td>" . $row['nama_awal'] . "</td>";
echo "<td>" . $row['nama_akhir'] . "</td>";
echo "</tr>";
}
echo "</table>";

mysql_close($con);
?>

```

Tampilan output yang dihasilkan akan berupa :



7.8 WHERE

Klausula WHERE digunakan untuk mengekstraksi suatu record berdasarkan kriteria atau parameter tertentu.

Sintaks :

```

SELECT column_name(s)
FROM table_name
WHERE column_name operator value.

```

Untuk contoh, program berikut akan menampilkan seluruh baris pada field nama_awal yang bernilai Fadli di dalam tabel biodata.

```

<?php
$con = mysql_connect("localhost","root","root");
if (!$con)
{
die('Could not connect: ' . mysql_error());

```

```
}  
  
mysql_select_db("my_db", $con);  
  
$result = mysql_query("SELECT * FROM biodata  
WHERE nama_awal='Fadli'");  
  
while($row = mysql_fetch_array($result))  
{  
    echo $row['nama_awal'] . " " . $row['nama_akhir'];  
    echo "<br />";  
}  
?>
```

Output yang dihasilkan :
Fadli Muhammad.

7.9 ORDER BY

Keyword ORDER BY digunakan untuk mengurutkan suatu data di dalam kumpulan berbagai record. Secara default instruksi ORDER BY akan melakukan pengurutan data secara ascending. Jika kita ingin mengurutkan data secara descending, PHP telah menyediakan keyword DESC.

Sintaks :

```
SELECT column_name(s)  
FROM table_name  
ORDER BY column_name(s) ASC|DESC
```

Contoh program berikut adalah program mengurutkan data pada tabel biodata berdasarkan field usia.

```
<?php  
$con = mysql_connect("localhost","root","root");  
if (!$con)  
{  
    die('Could not connect: ' . mysql_error());  
}  
  
mysql_select_db("my_db", $con);
```

```
$result = mysql_query("SELECT * FROM biodata ORDER BY usia");
```



```
while($row = mysql_fetch_array($result))
{
    echo $row['NIP'];
    echo $row['nama_awal'];
    echo " " . $row['nama_akhir'];
    echo " " . $row['usia'];
    echo "<br />";
}

mysql_close($con);
?>
```

Output yang dihasilkan akan berupa :

```
132321544    Salbiah Hanum           29
132333333    Fadli Muhammad         50.
```

7.10 UPDATE

Istruksi UPDATE digunakan untuk memperbaharui data yang telah ada di dalam sebuah tabel.

Sintaks :

```
UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value.
```

Program berikut merupakan program memperbaharui usia Fadli, dari 50 menjadi 25.

```
<?php
$con = mysql_connect("localhost","root","root");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

mysql_query("UPDATE biodata SET usia = '25'
WHERE nama_awal = 'Fadli' AND nama_akhir = 'Muhammad'");

mysql_close($con);
?>
```

Keadaan data setelah diperbaharui sekarang menjadi :

132321544	Salbiah Hanum	29
132333333	Fadli Muhammad	25.

7.11 DELETE

Instruksi DELETE digunakan untuk menghapus satu record yang terdapat di dalam tabel.

Sintaks :

```
DELETE FROM table_name
```

```
WHERE some_column = some_value.
```

Contoh program berikut akan menghapus seluruh data Salbiah.

```
<?php
```

```
$con = mysql_connect("localhost","root","root");
```

```
if (!$con)
```

```
{
```

```
die('Could not connect: ' . mysql_error());
```

```
}
```

```
mysql_select_db("my_db", $con);
```

```
mysql_query("DELETE FROM biodata WHERE  
nama_awal='Salbiah'");
```

```
mysql_close($con);
```

```
?>
```

Setelah eksekusi, data yang ada pada tabel hanya akan berisikan :

132333333	Fadli Muhammad	25.
-----------	----------------	-----

∞

This page is intentionally left blank

Bab 8

MENGINTEGRASIKAN BEBERAPA STASIUN TELEVISI DI SITUS PRIBADI

8.1 Pendahuluan

Menonton televisi langsung melalui internet bukanlah hal yang baru. Banyak situs-situs lokal dan dunia yang menyediakan berbagai fasilitas TV online yang dapat dinikmati langsung secara gratis. Beberapa situs yang menyediakan fasilitas TV online diantaranya adalah :

URL	Channel	Jumlah Channel
http://www.ongoz.co.cc/	Lokal dan Luar	±20
http://televisi-fadli.blogspot.com	Lokal dan Luar	±25
http://www.jakartacityview.com/	Lokal	±4
http://www.tvone.co.id/	Lokal	1
http://www.indoweb.tv/	Lokal	±4
http://indotv-online.tk/	Lokal dan Luar	>11
http://www.imediabiz.tv/	Lokal	±5
http://www.unduh.com/	Lokal	±4

Keseluruhan situs memiliki tampilan yang menarik, yang nantinya dapat kita ambil untuk menghiasi situs pribadi kita.

Beberapa pilihan tampilan yang akan kita gunakan :
Tampilan web pada domain <http://indotv-online.tk/>.



Tampilan web pada domain <http://www.unduhan.com/>.



Tampilan web pada domain <http://www.ongoz.co.cc/>.



8.2 Embed Kode untuk Televisi

Jika anda belum memiliki situs atau domain, maka anda dapat mencoba beberapa situs gratis yang disediakan di internet. Penulis sendiri memanfaatkan fasilitas blogger gratis dengan nama domain www.senyumvision.blogspot.com, untuk memasang televisi online di situsnya.

Jika anda telah memiliki situs, maka anda dapat mencoba mengikuti langkah-langkah pemasangan TV online di situs anda.

1. Jika anda pemakai situs gratis berekstensi “blogspot.com”, anda dapat langsung sign in.

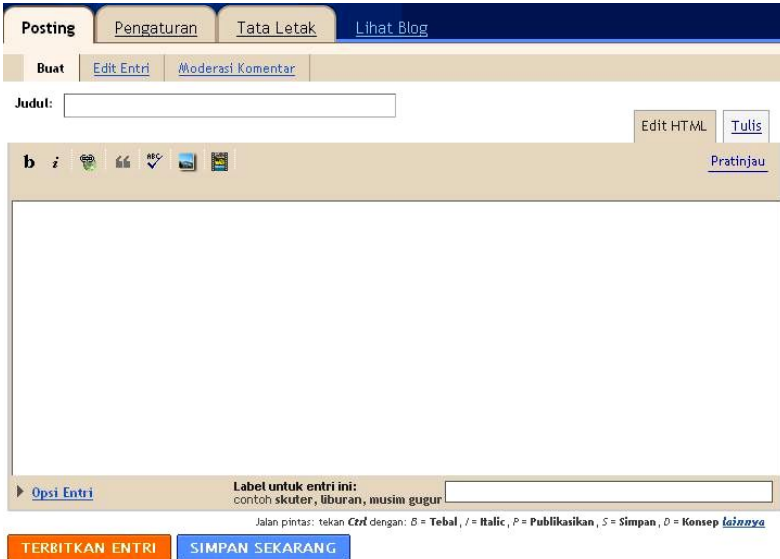


2. Lalu setelah muncul tampilan :



klik pada .

3. Tidak lama kemudian muncul tampilan :




dan tuliskan kode berikut pada halaman posting.

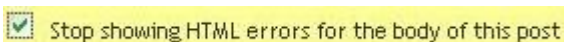
```
<center><iframe marginheight="0" marginwidth="0"
src="http://unduh.com/" scrolling="no" width="80%"
align="middle" frameborder="0" height="440"></iframe>
```

Lalu klik .

4. Jika muncul tampilan :



maka berikan tanda check pada :



dan akhiri dengan mengklik .

Tidak lama kemudian muncul tampilan :



sekarang kita telah sukses memposting televisi di situs pribadi kita

∞

DAFTAR PUSTAKA

Fadlisyah. 2006. *Diktat Kuliah PHP & MySQL*, Universitas
Malikussaleh.

www.w3schools.com.

www.Mantisa.blogspot.com

PEMROGRAMAN PHP & MySQL

Tidak dipungkiri, PHP merupakan bahasa pemrograman yang paling banyak digemari saat ini, dan PHP hampir selalu hadir berpasangan dengan database MySQL. Semakin populernya, Banyak para programmer non-PHP dari yang awal hingga yang tingkat mahir tergoda untuk ingin mempelajarinya. Selain perangkat lunaknya yang open-source, PHP memiliki keunggulan dalam hal kecepatan pengekseskuan kode-kode program-nya, sehingga menjadi pilihan utama pemro-graman bagi para programer di indonesia, Alih-alih PHP didukung dengan database MySQL yang memiliki kapasitas penyimpanan datanya yang cukup besar. Buku ini menjembantani para programer non-PHP dan para pemula yang awam dengan programming, Untuk mengenal secara praktis dan singkat mengenai PHP. Adapun materi yang dibahas dibuka dengan sejarah PHP, instalasi PHP, Menggunakan PHP designer, Sejarah MySQL, Tipe data, Operator, Operasi pada string, Dilanjutkan dengan pernyataan kondisi, Perulangan, Array, Fungsi, Bekerja dengan form, Koneksi database MySQL, dan diakhiri dengan streaming televisi melalui situs pribadi.

UNIMAL PRESS

ISBN 978-602-464-074-3

