

TUTORIAL PEMBUATAN SISTEM INFORMASI DENGAN MUDAH DAN MENYENANGKAN MENGGUNAKAN FRAMEWORK CODEIGNITER

Penulis:

Mohamad Nurkamal Fauzan, S.T., M.T.
Si Made Angga Dwitya P

TUTORIAL PEMBUATAN SISTEM INFORMASI DENGAN MUDAH DAN MENYENANGKAN MENGGUNAKAN FRAMEWORK CODEIGNITER

**Mohamad Nurkamal Fauzan, S.T., M.T.
Si Made Angga Dwitya P
Informatics Research Center**



Kreatif Industri Nusantara

Penulis:

Mohamad Nurkamal Fauzan, S.T., M.T.
Si Made Angga Dwitya P

ISBN: 978-602-53897-8-8

Editor:

Mohamad Nurkamal Fauzan, S.T., M.T.

Penyunting:

Mohamad Nurkamal Fauzan, S.T., M.T.
Si Made Angga Dwitya P

Desain sampul dan Tata letak:

Muchamad Innal Kariem

Penerbit:

Kreatif Industri Nusantara

Redaksi:

Jl. Ligar Nyawang No. 2
Bandung 40191
Tel. 022 2045-8529
Email : awangga@kreatif.co.id

Distributor:

Informatics Research Center
Jl. Sariyah No. 54
Bandung 40151
Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara apapun tanpa ijin tertulis dari penerbit

Puji syukur penulis panjatkan kehadiran Allah SWT karena atas berkat rahmat dan karunia-Nyalah sehingga penulis telah diberikan kekuatan dan kesabaran dalam proses pembuatan dan penyelesaian buku ini

Penulis Membuat Buku yang berjudul “Tutorial Pembuatan Sistem Informasi Dengan Mudah Dan Menyenangkan Menggunakan Framework Codeigniter”

Penulis juga berharap semoga Buku dapat berguna bagi pembaca dalam membuat sistem informasi dengan menggunakan Framework Codeigniter. Selama menyusun Buku ini, penulis banyak memperoleh bimbingan, pengarahan, saran, dukungan serta bantuan dari berbagai pihak mulai dari saat mempersiapkan, menyusun, hingga dapat menyelesaikan Buku . Maka pada kesempatan ini penulis ingin menyampaikan rasa terima kasih kepada semua pihak yang turut membantu dalam penyusunan buku yaitu:

1. M. Yusril Helmi Setyawan, S.Kom., M.Kom. selaku Ketua Program Studi D4 Teknik Informatika.
2. Nisa Hanum Harani, S.Si.,M.T. selaku Koordinator Internship 1.
3. Mohamad Nurkamal Fauzan, S.T.,M.T. selaku Dosen Pembimbing .
4. Nisa Hanum Harani, S.Si.,M.T. sebagai dosen wali penulis yang sangat membantu dan memberikan dukungan menyelesaikan Buku.
5. Seluruh dosen program studi Teknik Informatika yang telah memberikan ilmu pengetahuan dan wawasan yang berguna bagi penulis.
6. Seluruh pengurus perpustakaan, yang telah menyediakan banyak referensi yang tentunya sangat mendukung penulis dalam penyelesaian laporan Internship 1 ini.
7. Orang tua khususnya Ibu serta Kakak Penulis yang telah memberikan dukungan baik moril maupun materil.
8. Teman – teman seperjuangan penulis Teknik Informatika 4B
9. Seluruh pihak yang telah membantu dalam penyelesaian penyusunan laporan ini sesuai yang diharapkan yang tidak bisa penulis sebutkan satu persatu.

Semoga Allah SWT melimpahkan kasih sayang-Nya serta membala kebaikan semua pihak yang telah membantu dalam penyusunan laporan ini.

Dalam penulisan, penulis menyadari bahwa masih banyak kekurangan dalam laporan ini mengingat keterbatasan pengetahuan penulis. Oleh karena itu penulis senantiasa menerima segala kritikan dan saran yang sifatnya membangun dalam penyempurnaan laporan ini. Sehingga laporan ini dapat menjadi lebih baik dan bermanfaat bagi semua pembaca.

Bandung, 19 Januari 2020

Penulis

DAFTAR ISI

DAFTAR ISI.....	5
BAB I PENGENALAN WEB	9
1.1 Pengenalan Pemrograman Web.....	9
1.2 Keuntungan Pemrograman Web.....	10
1.3 Cara Kerja Pemrograman Web.....	10
1.4 Alur yang Dilakukan dengan Pemrograman Web	11
1.5 Pemrograman Web dan Jenisnya.....	11
1.6 Pembagian Pemrograman Web	13
1.7 Pengenalan <i>CSS</i>	14
1.8 Sejarah <i>CSS</i>	15
BAB II PENGENALAN BEBERAPA BAHASA PEMROGRAMAN	17
2.1 Pengenalan Bahasa Pemrograman.....	17
2.2 Pengertian <i>HTML</i>	18
2.2.1 Kelebihan <i>HTML</i>	18
2.2.2 Kekurangan <i>HTML</i>	19
2.2.3 Perkembangan <i>HTML</i>	19
2.2.4 Unsur <i>HTML</i>	21
2.2.5 Struktur Dasar <i>HTML</i>	21
2.2.6 <i>Tags HTML</i>	22
2.3 Pengertian PHP.....	31
2.3.1 Sejarah PHP	31
2.3.2 Kelebihan PHP.....	32
2.3.3 Kekurangan PHP.....	34
2.3.4 Perbedaan Dari Beberapa Versi PHP	35
2.3.5 Tiga Model Penggunaan PHP.....	38

2.3.6 Sintaks PHP	39
2.4 Hubungan Antara PHP dan HTML	74
2.5 Bahasa Pemrograman C	75
2.6 Bahasa Pemrograman Java	77
2.7 Bahasa Pemrograman Python	80
2.8 Bahasa Pemrograman Ruby	87
2.9 Bahasa Pemrograman COBOL	88
2.10 Bahasa Pemrograman Visual Basic	90
BAB III INSTALASI TOOLS YANG DIGUNAKAN	93
3.1 XAMPP	93
3.1.1 Cara Instalasi XAMPP	94
3.1.2 Cara Menjalankan XAMPP	98
3.2 Visual Studio Code	100
3.2.1 Cara Menginstal Visual Studio Code	102
BAB IV MENGENAL MYSQL DAN DATABASE	106
4.1 MYSQL	106
4.2 Sejarah Mysql	106
4.3 Kelebihan Mysql	107
4.4 Pengenalan Basis Data	109
4.4.1 Struktur Basis Data	109
4.4.2 Tujuan Sistem Basis Data	110
4.4.3 Keuntungan dan Kerugian Basis Data	110
4.4.4 Konsep Dasar Basis Data	112
4.4.5 Komponen Basis Data	115
4.5 Bahasa Basis Data	120
4.6 Cara Membuat Database dan Tabel	121
BAB V PENGENALAN FRAMEWORK CODEIGNITER	142
5.1 Pengenalan Framework	142

5.2 Alasan Menggunakan Framework	142
5.3 Manfaat Menggunakan Framework	143
5.4 Kekurangan Menggunakan Framework	143
5.5 Perbedaan CMS dan Framework	144
5.6 Pengertian CodeIgniter	145
5.7 Pengenalan OOP.....	146
5.8 Pengenalan Object	146
5.9 Pengenalan Struktur OOP	147
5.10 Apa Itu MVC?	152
5.11 Contoh File Model, Controller dan View.....	154
5.12 Alur Kerja Framework CodeIgniter.....	156
5.13 Fungsi Dari CodeIgniter.....	156
5.14 Kelebihan CodeIgniter	157
5.15 Kekurangan CodeIgniter	157
5.16 CodeIgniter 3.....	158
5.17 Perbedaan Dari Versi Framework CodeIgniter	159
5.18 Perbedaan Framework CodeIgniter Dengan Framework Lainnya	160
5.19 URL Framework CodeIgniter.....	166
5.20 Cara Instalasi CodeIgniter.....	166
5.21 Konfigurasi Framework CodeIgniter	170
5.22 Membuat Hello World di Framework CodeIgniter	178
5.23 Menghilangkan Index.php pada URL.....	182
5.24 Controller dan View.....	184
BAB VI CONTOH PROGRAM	188
6.1 Persiapan Pembuatan Aplikasi	188
6.2 Memasukan Tempate Ke CodeIgniter	188
6.2.1 Mengenal Template.....	188
6.2.2 Mengintergrasikan Tempate SB Admin Ke CodeIgniter	189

6.3 Rancangan Database	203
6.4 Membuat Model	204
6.5 Membuat Login	207
6.6 Membuat form Inputan User	211

BAB I

PENGANTAR WEB

Dalam bab ini menerangkan tentang segala sesuatu yang berhubungan dengan pemrograman web

1.1 Pengenalan Pemrograman Web



Pemrograman web terdiri dari pemrograman dan web. Pemrograman yang dalam bahasa inggris adalah *programming* dan diartikan proses, cara, atau perbuatan program (secara bahasa indonesia).

Definisi Web adalah sejumlah jaringan komputer yang terdiri dari kumpulan situs internet yang menampilkan teks, grafik, suara dan sumber daya animasi melalui protokol transfer *hypertext*. Orang banyak mengenal web dengan istilah WWW (*world wide web*), *World Wide Web* adalah layanan internet yang paling banyak digunakan pada saat ini internet mulai dikenal dan digunakan secara luas setelah adanya layanan WWW.

WWW adalah halaman-halaman website yang dapat saling terkoneksi satu dengan lainnya (*hyperlink*) yang membentuk informasi. WWW berjalan dengan protokol *HyperText Transfer Protokol* (HTTP). Halaman Web merupakan file teks murni (*plain text*) yang berisi sintaks-sintaks HTML yang dapat dibuka dan diterjemahkan dengan Internet Browser. Sintaks HTML mampu memuat konten text, gambar, audio, video dan animasi. Kini internet identik dengan web, karena kepopuleran web sebagai standar interface pada layanan-layanan yang ada di internet, dari awalnya sebagai penyedia informasi, ini digunakan juga untuk komunikasi dari email sampai dengan chatting, sampai dengan melakukan transaksi bisnis (*commerce*)

1.2 Keuntungan Pemrograman Web

Banyak keuntungan yang diberikan oleh Aplikasi berbasis Web dari pada aplikasi berbasis destop, sehingga aplikasi berbasis web telah diadopsi oleh perusahaan sebagai bagian dari strategi teknologi informasinya, karena beberapa alasan:

1. Akses informasi mudah
2. Setup server lebih mudah
3. Informasi mudah didistribusikan
4. Bebas platform
5. Informasi dapat di sajikan oleh browser web pada sistem operasi mana saja karena adanya standar dokumen berbagai tipe data dapat disajikan

1.3 Cara Kerja Pemrograman Web

Sebelum menjelaskan cara kerja pemrograman web, sebaiknya anda mengetahui tentang HTML bahwa HTML bukan bahasa pemrograman tapi adalah *markup language* (bahasa penandaan yang terdiri dari TAG). HTML hanya mendeskripsikan bahwa bagian tertentu dalam sebuah halaman web adalah isi yang harus ditampilkan oleh browser dengan cara tertentu.

Bagaimana cara kerja pemrograman web? Secara sederhana pemrograman web hanya dilakukan dengan menyisipkan perintah diantara TAG HTML. Misal:

```
<html>
<head>
<title>Contoh</title>
</head>
<body>
<?php
echo date("d/m/Y");
?>
</body>
</html>
```

Pada saat kita mengakses halaman tersebut, *web server* langsung merespon apa yang kita minta dengan melakukan proses *parsing* (membaca isi halaman baris perbaris) dan jika menemukan baris perintah bahasa pemrograman maka web server akan mengeksekusi/ menjalankan perintah tersebut dan setelah semua selesai baru akan mengirimkannya ke browser. Contoh diatas web server akan mengirimkan hasil eksekusi baris perintah `echo date("d/m/Y")` dalam bentuk tanggal dinamis yang dianggap sebagai bagian dari HTML yang bersifat statis.

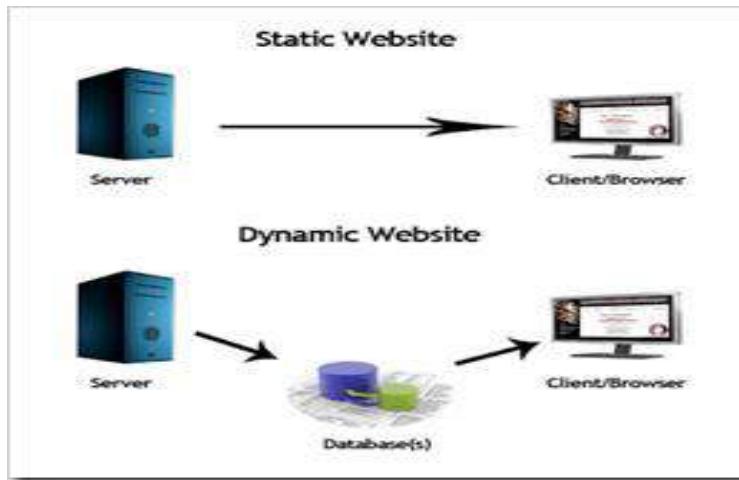
1.4 Apa Yang Dapat Dilakukan Dengan Pemrograman Web

1. Pemrograman web memungkinkan kita dapat membuat halaman web yang dinamis dengan perintah-perintah yang cukup sederhana tapi dapat menghasilkan informasi yang beragam.
2. Pemrograman web dapat melakukan interaksi sehingga pengguna dapat memperoleh informasi yang sesuai/relevan seperti pencarian, pengiriman data, dll.
3. Pemrograman web dapat dihubungkan dengan berbagai macam basis data/ database seperti mysql, oracle, sql server, dll.

1.5 Pemrograman Web dan Jenisnya

Web adalah fasilitas dari Hypertext yang memiliki fungsi untuk menampilkan data berupa text, gambar, suara, animasi dan data multimedia dan jika ingin dapat menguasai web maka diperlukan mengenal Bahasa pemrograman Web yaitu HTML dan PHP. HTML termasuk kedalam kategori Script Client Side sedangkan PHP termasuk Kedalam Script-Server Side yang akan dimasukan perintah perintah didalam suatu pemrograman web.

Web mengalami perkembangan yang sangat pesat mulai dari situs web E-commerce sampai dengan non-profit situs. Dan dapat dikategorikan menjadi 2 yaitu:



1. Web Statis

Web statis ialah web yang berisi tentang informasi yang memiliki sifat statis (tetap) atau pengguna tidak dapat berinteraksi dengan website tersebut, web statis dapat dilihat dari tampilan website tersebut jika suatu web hanya berhubungan dengan halaman web lain yang berisi informasi tetap maka web tersebut termasuk kedalam kategori web statis, pada web statis pengguna hanya dapat melihat isi web tersebut dan jika di klik hanya akan berpindah pada halaman lainnya. Dalam web statis interaksi pengguna sangatlah terbatas

2. Web Dinamis

Web dinamis adalah web yang dapat menampilkan informasi serta dapat membuat pengguna berinteraksi seperti dengan form input, button sehingga dapat mengolah informasi yang di tampilkan pada web tersebut, web dinamis bersifat tidak kaku dan terlihat lebih enak dipandang.

1.6 Pembagian Pemrograman Web

1. Client-Side Scripting

Client-Side Scripting digunakan ketika browser (pengguna) klien memiliki semua kode dan halaman tersebut diubah berdasarkan informasi klien (pengguna). Browser Web mengeksekusi skrip sisi klien yang terletak di dalam komputer pengguna. Skrip sisi klien juga dikenal sebagai skrip tertanam (karena mereka sering disematkan dalam dokumen HTML atau XHTML).

Browser mendapatkan halaman yang dikirim oleh server & mengeksekusi skrip sisi klien. Skrip sisi klien tidak dapat digunakan untuk bergabung dengan database di server web. Skrip sisi klien tidak bisa mendapatkan sistem file yang terletak di server web.

Catatan dan pengaturan yang bersifat lokal di komputer klien (pengguna) dapat didekati menggunakan bahasa skrip sisi Klien. Secara umum diamati bahwa respons dari skrip sisi klien lebih cepat bila dibandingkan dengan bahasa skrip sisi server saat skrip disiapkan di komputer lokal.

Contoh Script Paling Populer

1. Java Script
2. XML
3. CSS

2. Server-Side Scripting

Server-Side Scripting adalah teknik yang digunakan dalam pengembangan web yang melibatkan penggunaan skrip pada server web yang menghasilkan respons yang disesuaikan untuk permintaan setiap pengguna (klien) ke situs web. Alternatifnya adalah untuk server web itu sendiri untuk memberikan halaman web statis. Skrip dapat ditulis dalam salah satu dari sejumlah bahasa skrip sisi server yang tersedia (lihat di bawah). Skrip sisi server dibedakan dari skrip sisi klien di mana skrip tertanam, seperti JavaScript, dijalankan sisi klien dalam browser web, tetapi kedua teknik ini sering digunakan bersama.

Server-Side Scripting sering digunakan untuk menyediakan antarmuka yang disesuaikan untuk pengguna. Skrip ini dapat mengumpulkan karakteristik klien untuk digunakan dalam menyesuaikan respons berdasarkan karakteristik tersebut, persyaratan pengguna, hak akses, dll. Skrip sisi server juga memungkinkan pemilik situs web untuk menyembunyikan kode sumber yang menghasilkan

antarmuka, sedangkan dengan sisi klien scripting, pengguna memiliki akses ke semua kode yang diterima oleh klien. Sisi buruk dari penggunaan skrip sisi server adalah bahwa klien harus membuat permintaan lebih lanjut melalui jaringan ke server untuk menunjukkan informasi baru kepada pengguna melalui browser web. Permintaan ini dapat memperlambat pengalaman bagi pengguna, menempatkan lebih banyak beban di server, dan mencegah penggunaan aplikasi saat pengguna terputus dari server.

Ketika server menyajikan data dengan cara yang umum digunakan, misalnya sesuai dengan protokol HTTP atau FTP, pengguna dapat memilih sejumlah program klien (sebagian besar browser web modern dapat meminta dan menerima data menggunakan kedua protokol tersebut). Dalam hal aplikasi yang lebih khusus, pemrogram dapat menulis sendiri server, klien, dan protokol komunikasi mereka, yang hanya dapat digunakan satu sama lain.

1.7 Pengenalan CSS



CSS adalah singkatan dari Cascading Style-Sheet, sebuah pengembangan atas kode HTML yang sudah ada sebelumnya. Dengan CSS, kita bisa menentukan sebuah struktur dasar halaman web secara lebih mudah dan cepat, serta irit size. CSS juga membantu kita untuk menyeragamkan seluruh halaman website dengan tampilan yang konsisten. Misalnya, kita mau seluruh font yang ada di website kita adalah font "Tahoma", maka dengan bantuan CSS kita bisa bikin proses itu menjadi otomatis tanpa harus mengganti-ganti font secara manual di setiap halaman.

Dahulu kala, sebelum CSS menjadi standar untuk mendesain halaman web seperti sekarang, halaman web di desain menggunakan <table>. jadi dibuat dulu desainnya, dalam format .psd atau jpeg, lalu di slice atau di potong potong menjadi bagian – bagian terpisah. setelah itu dibuat table dengan ukuran yang sesuai, lalu desain tadi di “tempel” pada table sebagai image yang melekat pada tabel, atau sebagai background. Kelemahan dari cara seperti ini adalah, halaman web menjadi berat karena kolom – kolom penyusun table <tr> dan <td> diberi tambahan atribut image source. Side effect dari hal ini adalah halaman web menjadi padat dan tidak SEO friendly.

Pada perkembangannya CSS sudah masuk level 3 untuk sekarang, dimana dimulai CSS level 1 atau yang sering di sebut CSS aja, kemudian level 2 yang merupakan Penyempurnaan dari CSS level sebelumnya, yaitu CSS level 1. CSS merupakan alternatif bahasa pemrograman web masa yang akan datang,

Dimana mempunyai banyak keuntungan, diantaranya :

1. Ukuran file lebih kecil
2. Load file lebih cepat
3. Dapat berkolaborasi dengan Javascript
4. Pasangan setia XHTML
5. Menghemat pekerjaan tentunya, dimana hanya membuat 1 halaman
6. CSS.
7. Mudah mengganti-ganti tampilan dengan hanya merubah file CSS nya saja.

1.8 Sejarah CSS

Pada tanggal 17 Agustus 1996, World Wide Web Consortium (W3C) menjadikan CSS sebagai bahasa pemrograman standart dalam pembuatan dokumen web. Tujuannya adalah mengurangi pembuatan tag-tag baru oleh Netscape dan Internet Explorer, karena kedua browser pada saat itu bersaing mengembangkan TAG sendiri untuk mengatur tampilan web.

CSS level 1 mendukung pengaturan tampilan dalam hal:

1. Font (jenis, ketebalan)
2. Warna, teks, background dan elemen lainnya
3. Text attributes, misalnya spasi antar baris, kata dan huruf
4. Posisi teks, gambar, table dan elemen lainnya
5. Marjin, border dan padding.

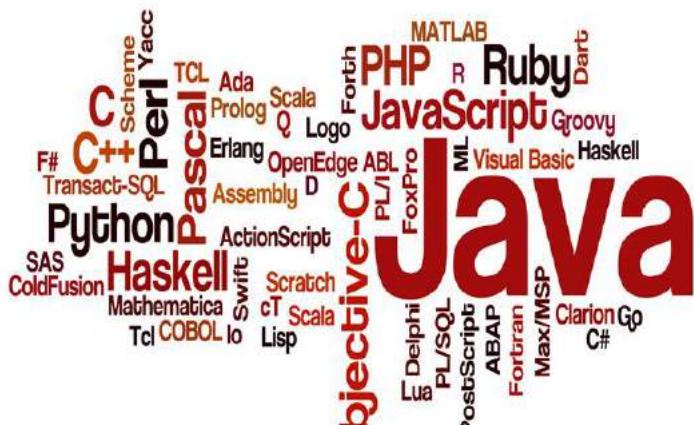
Selanjutnya di tahun 1998, W3c menyempurnakan CSS awal dengan menciptakan standar CSS2 (CSS level 2) yang menjadi standar hingga saat ini. Pada CSS level 2 ini, di masukkan semua atribut dari CSS1, serta diperluas dengan penekanan pada Internasional accessibility and capability khususnya media-specific CSS. Bahkan pada tahun 2000, tidak lama setelah CSS2 diimplementasikan. CSS2 ini sampai sekarang masih terus dikembangkan, spesifikasinya dibagi pada beberapa topik atau modul.

BAB II

PENGENALAN BEBERAPA BAHASA PEMROGRAMAN

Di bab ini diharapkan pembaca dapat mengerti mengenai beberapa Bahasa pemrograman yang dijelaskan

2.1 Pengenalan Bahasa Pemrograman



Bahasa pemrograman dibagi menjadi dua yaitu *client side scripting* dan *server-side scripting*. Client-side scripting merupakan script atau Bahasa pemrogramannya atau eksekusinya dilakukan oleh *web browser*. Dengan kata lain aplikasi dapat berjalan langsung di web browser yang berada di sisi client. Contoh *HTML, Javascript* dan *Css*, Sedangkan server-side scripting bahasa pemrograman yang eksekusinya berada di server. Contoh *PHP, ASP*.

2.2 Pengertian HTML



HTML merupakan turunan atau pengembangan dari SGML (Standar Generalized Markup Language). HTML sendiri dikembangkan oleh Tim Berners-Lee sewaktu masih bekerja di CERN yang pertama kali dipopulerkan oleh browser Mosaic yang dikembangkan NCSA. Selama awal tahun 1990an, HTML semakin memiliki perkembangan yang sangat cepat. Akan tetapi pengembangan resmi HTML baru dikeluarkan oleh Internet Enggineering Task Force (IETF) yang dikeluarkan pada tahun 1995. HTML2 yang dikembangkan ini merupakan turunan dari HTML+ pada tahun 1993. HTML3 yang juga dirilis pada tahun 1995 mempunyai kemampuan yang jauh lebih bagus dari versi sebelumnya. Merupakan hasil dari usaha yang dikembangkan oleh World Wide Web Consortium's (W3C) yang kemudian menghasilkan HTML3 di tahun 1996 dan rilislah HTML4 pada akhir tahun tersebut yaitu 1997 dan 1998.

2.2.1 Kelebihan HTML

1. Merupakan bahasa penkodean yang lintas platform (cross platform), maksudnya HTML dapat digunakan pada berbagai jenis mesin komputer yang berbeda dan berbagai macam sistem operasi yang berbeda. Jadi bersifat fleksibel karena ditulis cukup dengan menggunakan editor karakter ASCII.
2. Dapat disisipi gambar baik gambar statis atau dinamis (animasi) termasuk menggunakan gambar untuk dijadikan hyperlink. Gambar di sini digunakan untuk merujuk pada suatu halaman web, dimana setiap titik-titik yang sudah didefinisikan berupa rectangular (kotak), poligon (kurva tak beraturan) atau

- lingkaran digunakan untuk ‘jump’ ke halaman lain, atau link ke halaman di luar web yang bersangkutan.
3. Dapat disisipi animasi berupa Java Applet atau file-file animasi dari Macromedia Flash atau Macromedia Shockwave (untuk keperluan ini, browser harus memiliki plug-in khusus untuk menjalankan file-file animasi ini).
 4. Dapat disisipi bahasa pemrograman untuk mempercantik halaman web seperti Javascript, VBScript, Active Server Pages, Perl, Tcl, PHP dan sebagainya.
Bukan merupakan bahasa pemrograman jadi tidak memerlukan kompiler. Cara menjalanakannya cukup dengan menggunakan browser.

2.2.2 Kekurangan HTML

1. Menghasilkan halaman yang statis, yang saya tahu untuk memperoleh halaman yang dinamis harus menggunakan bahasa pemrograman tertentu seperti Javascript atau VBScript dan animasi seperti Flash atau Shockwave.
2. Memiliki tag-tag yang begitu banyak sehingga susah dipelajari untuk yang masih awam.
3. Tidak dapat menghasilkan halaman yang interaktif. Interaktif di sini maksudnya client dapat berinteraksi dengan server. Untuk keperluan itu, HTML harus disisipi bahasa pemrograman yang dapat menangani hal tersebut, contohnya Perl atau Tcl.

2.2.3 Perkembangan HTML

Sebelum suatu HTML disahkan sebagai suatu dokumen HTML standar, ia harus disetujui dulu oleh W3C untuk dievaluasi secara ketat. Setiap terjadi perkembangan suatu versi HTML, maka mau tak mau browser pun harus memperbaiki diri agar bisa mendukung kode-kode HTML yang baru tersebut. Sebab jika tidak, browser tak akan bisa menampilkan HTML tersebut.

1. HTML versi 1.0

Kemampuan yang dimiliki versi 1.0 ini antara lain heading, paragraph, hypertext, list, serta cetak tebal dan miring pada teks. Versi ini juga mendukung peletakan image pada dokumennya tanpa memperbolehkan meletakkan teks disekelilingnya (wrapping).

2. HTML versi 2.0

Pada versi ini, penambahan kualitas HTML terletak pada kemampuannya untuk menampilkan suatu form pada dokumen. Dengan adanya form ini, maka kita dapat memasukkan nama, alamat, serta saran/kritik. HTML versi 2.0 ini merupakan pionir dari adanya homepage interaktif.

3. HTML versi 3.0

HTML versi 3.0 menambahkan beberapa fasilitas baru seperti table. Versi ini yang disebut juga sebagai HTML+ tidak bertahan lama dan segera digantikan HTML versi 3.2.

4. HTML versi 4.0

HTML versi 4 ini merupakan HTML versi terakhir pada saat sumber ini diambil. HTML ini memuat banyak sekali perubahan dan revisi dari pendahulunya. Perubahan ini terjadi di hampir segala perintah-perintah HTML seperti table, image, link, text, meta, imagemaps, form, dan lain- lain.

Kemudian lahir HTML versi 4.01 yang dikeluarkan secara resmi oleh W3C pada tanggal 24 April 1998. HTML 4.01 menjadi standart pada tahun 1999. HTML 4.01 merupakan perbaikan dari HTML versi 4.0 yang terlebih dahulu diterbitkan (18 Desember 1997). HTML versi HTML 4.01 masih standart resmi sampai sekarang ini .

5. HTML 5

HTML 5 merupakan perkembangan html generasi saat ini dan merupakan generasi terakhir dan mungkin akan berkembang lagi mengikuti perkembangan jaman Ada

2.2.4 Usur HTML

- **Tag**

Tag adalah simbol khusus (markup) berupa 2 karakter “<” dan “>” yang mengapit suatu teks sebagai nama tag. Contohnya tag <body> adalah tag dengan nama body.

- **Atribut**

Atribut yaitu property yang mengatur bagaimana elemen dari suatu tag akan ditampilkan. Atribut ada yang memiliki nilai dan ada yang tidak memiliki nilai. Nilai suatu atribut ditulis di dalam tanda petik ganda (“...”), dipisahkan dengan symbol sama dengan (=) dari nama. Contohnya, <p align=”center”>.

- **ELEMENT**

Element adalah bagian dari skrip HTML yang terdiri dari tag pembuka, isi element dan tag tutup. Jika element ditampilkan di browser maka hanya isi element saja yang akan tampil. Contohnya, <p> Politeknik Pos Indonesia </p> maka jika ditampilkan pada browser akan tampil hanya tulisa Politeknik Pos Indonesia

2.2.5 Struktur dasar HTML

Dalam penulisan HTML ada beberapa tag yang wajib dituliskan dengan struktur yang telah ditentukan. Tag tersebut adalah sebagai berikut :

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Poltekpos</title>
  </head>
  <body>
    Politeknik Pos Indonesia
  </body>
</html>
```

- <!DOCTYPE HTML> adalah tag awal dari setiap dokumen HTML yang berfungsi untuk menginformasikan pada browser bahwa dokumen yang sedang dibuka adalah dokumen HTML.
- <html>...</html> adalah tag yang menunjukkan pembuka dan penutup dokumen HTML.
- <head>...</head> adalah tag yang digunakan untuk menyimpan berbagai informasi tentang dokumen HTML.
- <title>....</title> adalah tag yang digunakan untuk membuat judul website yang nanti akan muncul di browser.
- <body>....</body>? Adalah tag yang menunjukkan bagian utama website. Semua yang akan ditampilkan pada halaman browser dituliskan di tag ini.

2.2.6 HTML Tags

2.2.6.1 Struktur Dasar

```
<span style="font-size: small;">
  <html>
    <head>
      <!--ini adalah bagian header-->
      <title>judul web</title>
    </head>
    <body>
      <!--tempat menuliskan kode-->
    </body>
  </html>
</span>
```

2.2.6.2 <a>

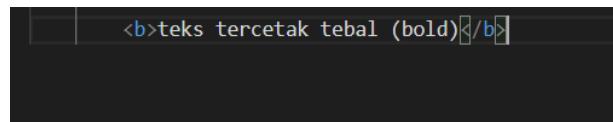
Mendefinisikan tag anchor. Digunakan untuk membuat link ke dokumen lain menggunakan atribut href dan juga untuk membuat bookmark dokumen menggunakan atribut name.

```
<body>
  <a href="http://www.google.com">kunjungi website google</a>
</body>
```

**2.2.6.3 **

Digunakan untuk mencetak tebal suatu teks.

Contoh:

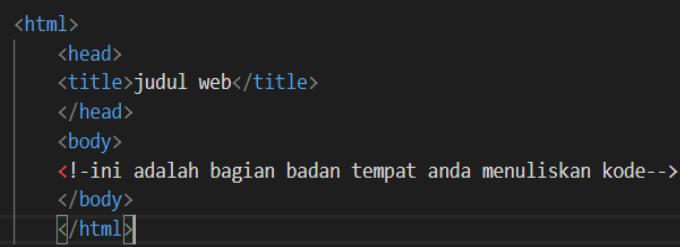


```
<b>teks tercetak tebal (bold)</b>
```

2.2.6.4 <body>

Digunakan untuk mendefinisikan badan html, yang didalamnya merupakan tempat untuk meletakan elemen-elemen html lainnya, seperti gambar, teks, table, form, dll

Contoh:

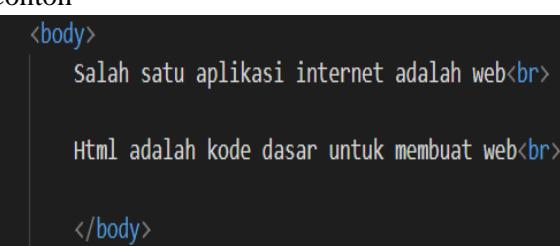


```
<html>
  <head>
    <title>judul web</title>
  </head>
  <body>
    <!--ini adalah bagian badan tempat anda menuliskan kode-->
  </body>
</html>
```

**2.2.6.5
**

Digunakan untuk menyisipkan sebuah baris(ganti baris)

Contoh



```
<body>
  Salah satu aplikasi internet adalah web<br>

  Html adalah kode dasar untuk membuat web<br>

</body>
```

2.2.6.6 <center>

Digunakan untuk menampilkan teks ditengah halaman

Contoh:

```
<center> teks ini akan tampil ditengah halaman</center>
```

2.2.6.7 <div>

Digunakan untuk mendefinisikan bagian dari sebuah halaman web. Seringkali digunakan untuk mengelompokan elemen-elemen HTML dengan format style tertentu.

Contoh:

```
<div style='color:#ff0000'>
    <h3>ini merupakan judul</h3>
    <p>ini merupakan paragraph</p>
</div>
```

2.2.6.8 <form>

Digunakan untuk mendefinisikan sebuah form masukan dari pengguna

Contoh:

```
<form action="action.html" method="get">
    Nama: <input type="text" name="f_nama"/>

    email: <input type="text" name="f_email"/>
    |
    <input type="submit" value='simpan'/>
</form>
```

2.2.6.9 <h1> sampai dengan <h6>

Digunakan untuk mendefinisikan bagian atas (heading) teks. Dimulai dari ukuran besar<h1> hingga ukuran kecil<h6>

Contoh:

```
<h1> HTML adalah kode dasar membuat web</h1>
<h2> HTML adalah kode dasar membuat web</h2>
<h3> HTML adalah kode dasar membuat web</h3>
<h4> HTML adalah kode dasar membuat web</h4>
<h5> HTML adalah kode dasar membuat web</h5>
<h6> HTML adalah kode dasar membuat web</h6>
```

2.2.6.10 <head>

Digunakan untuk mendefinisikan bagian atas(head) sebuah dokumen html. Bagian ini biasanya digunakan untuk mendefinisikan tag <base>, <link>, <meta>, <script>, <style>, dan <title>.

Contoh:

```
<head>
|   <title>judul web</title>
|   </head>
```

2.2.6.11<hr>

Digunakan untuk mendefinisikan sebuah garis horizontal dalam halaman html

Contoh:

```
~ | <html>
~ | <head>
| | <title>HR </title>
| | </head>
~ | <body>
| | Baris Pertama
| | <hr>
| | Baris Kedua
| | <hr>
| | Baris Ketiga
| | <hr>
| | </body>
~ | </html>
```

2.2.6.12<i>

Digunakan untuk mencetak miring suatu teks.

Contoh:

```
<body>
| <i>teks tercetak miring(italic)</i>
| </body>
```

2.2.6.13<p>

Digunakan untuk membuat paragraph baru.

Contoh:

```
<div style='color:#ff0000'>
  <h3>ini merupakan judul</h3>
  <p>ini merupakan paragraph</p>
</div>
```

2.2.6.14 <Select>

Digunakan untuk mendinisikan daftar pilihan

Contoh:

```
<select>
  <option value="detik">detik</option>
  <option value="viva">vivanews</option>
  <option value="glodokshop">glodok shop</option>
  <option value="bhineka">Bhinneka</option>
</select>
```

2.2.6.15

Digunakan untuk mempertegas tampilan suatu teks.

Contoh:

```
Kode dasar untuk membuat web adalah <strong>HTML</strong>
```

2.2.6.16 <style>

Digunakan untuk mendefinisikan informasi style pada dokumen html

Contoh:

```
<head>
  <style type="text/css">
    span.blue {color: #lightskyblue;
    font-weight:bold}
  </style>
</head>
<body>
  Kode dasar untuk membuat web adalah <span class="blue">HTML</span>
</body>
```

2.2.6.17 <table>

Digunakan untuk membuat table dalam halaman web.

Contoh:

```
<table border="1">
    <caption>daftar situs</caption><tr>
        <th>berita</th>
```

2.2.6.18 <title>

Digunakan untuk mendefinisikan judul pada dokumen html.

Contoh:

```
<head>
    <title>judul web</title>
</head>
```

2.2.6.19 <u>

Digunakan untuk menambahkan garis bawah suatu teks

Contoh:

```
<u>teks bergaris bawah(bold)</u>
```

2.2.6.20

Digunakan untuk membuat daftar yang tak berurut(unordered list) menggunakan symbol bullets.

Contoh:

```
<ul>
    <li>webpribadi</li>
    <li>blog</li>
    <li>facebook</li>
</ul>
```

2.2.6.21

digunakan untuk mendefinisikan huruf

Contoh:

```
<font size="4" color="green" > Font berwarna </font>
```

2.2.6.22<big>

digunakan untuk membuat teks berukuran besar
Contoh:

```
<big> Mengubah ukuran teks menjadi besar </big>
```

2.2.6.23<dt>

digunakan untuk membuat daftar item
Contoh:

```
<dl>
  <dt> daftar menu makanan </dt>
    <dd> Kentang goreng </dd>
    <dd> Empek-empek </dd>
    <dd> Telor dadar </dd>

  <dt> daftar menu minuman </dt>
    <dd> Jus alpukat </dd>
    <dd> Air putih </dd>
    <dd> Es teh manis </dd>
</dl>
```

2.2.6.24<button>

digunakan untuk mendefinisikan sebuah tombol
Contoh:

```
<button type="submit"> SIMPAN </button>
```

2.2.6.25

digunakan untuk menekankan sebuah teks
Contoh:

```
<em> Coba </em> adalah orang |
```

2.2.6.26 <framset>

digunakan untuk mendefinisikan sebuah window tertentu pada sebuah frameset

Contoh :

```
<frameset cols="25%,50%,25%">
  <frame src="framea.html">
  <frame src="frameb.html">
  <frame src="framec.html">
```

2.2.6.27<!-- .. -->

Digunakan untuk menuliskan komentar,komentar tersebut nantinya tidak akan ditampilkan pada web browser.

Contoh:

```
<!-- Contoh membuat Komentar | -->
```

2.2.6.28<!doctype>

Deklarasi ini tidak termasuk tag HTML, tetapi merupakan instruksi untuk web browser tentang versi bahasa markup yang digunakan.

Contoh:

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

2.2.6.29<textarea>

Tag untuk membuat sebuah kontrol input multibaris (text area)

Contoh;

```
<textarea rows="4" cols="50">
  Belajar membuat teksarea dengan benar
  </textarea>
```

2.2.6.30<Picture>

Menampilkan atau menambahkan gambar

Contoh:

```
<picture>
  <source media="(min-width: 650px)" srcset="img_pink_flowers.jpg">
  <source media="(min-width: 465px)" srcset="img_white_flower.jpg">
  
</picture>
```

2.2.6.31<td>

Tag untuk membuat sel dalam sebuah table

Contoh:

```
<table>
  <tr>
    <td>Cell A</td>
    <td>Cell B</td>
  </tr>
</table>
```

2.2.6.32<tr>

Tag untuk membuat baris dalam sebuah table

Contoh:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```

2.2.6.33<th>

Tag untuk membuat sebuah sel header table

Contoh:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```

2.3 Pengertian PHP



PHP Merupakan singkatan dari "PHP: Hypertext Preprocessor", yang merupakan sebuah bahasa scripting (kode untuk membangun suatu program) yang dikombinasikan pada HyperText Markup Language (HTML). Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat.

2.3.1 Sejarah PHP

PHP adalah singkatan dari “PHP: Hypertext Preprocessor”, yaitu bahasa pemrograman disisi server yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML. Ketika Anda mengakses sebuah URL, maka web browser akan melakukan request ke sebuah web server.

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilisan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram 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 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.

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. Server web bawaan ditambahkan pada versi 5.4 untuk mempermudah pengembang menjalankan kode PHP tanpa menginstall software server.

Versi terbaru dan stabil dari bahasa pemograman PHP saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.

2.3.2 Kelebihan Dari PHP

1. Memiliki Community yang besar

Programmer Web mana yang tidak mengetahui PHP, semua web programmer paling tidak pasti pernah mencoba PHP. Banyak sekali website yang menggunakan PHP sebagai bahasa pemrograman untuk membuat aplikasi web atau website nya. Facebook, Yahoo, Wikipedia, WordPress

adalah contoh website terkenal yang menggunakan PHP. Forum untuk membahas dan juga saling bertukar pikiran dalam pemrograman PHP juga telah banyak muncul di berbagai situs. Kebanyakan kuliah di bidang IT mengajarkan PHP sebagai bahasa pemrograman awal untuk mahasiswanya yang berkuliah di jurusan website development (baca juga: Daftar Mata Kuliah Teknik Informatika)

2. Mudah Dipelajari

PHP mudah di install dan dikonfigurasi. membuatnya menjadi bahasa pemrograman tingkat entry level yang mudah dipelajari bagi seseorang yang baru memulai belajar pengembangan web. Tutorial untuk memulai belajar pemrograman PHP dapat diperoleh dengan mudah secara online, di toko buku, ataupun di lembaga bimbingan kursus pengembangan website.

3. Pengembangan Cepat

Membuat Aplikasi menggunakan PHP jauh lebih cepat daripada mengembangkan aplikasi web menggunakan bahasa pemrograman lain. banyak sekali tools, boiler yang tersedia secara open source untuk bahasa pemrograman PHP. hal ini mempercepat proses dari start sampai dengan finish sebuah projek pembuatan aplikasi web.

4. Ringkas

Bagi Programmer web yang pernah mencoba bahasa ASP maupun java pasti mengetahui betul satu kelebihan ini. Mulai dari proses install yang tidak perlu setting berlebihan, konfigurasi dengan database yang mudah. hingga proses pengembangan yang tidak memerlukan waktu kompilasi. membuat PHP terasa sangat ringkas dan praktis berbeda dengan bahasa pemrograman lain yang membutuhkan proses kompilasi untuk dapat melihat website yang telah diselesaikan pembuatan kodennya. Bahkan, bahasa pemrograman php dapat digunakan didalam dokumen html, hmm ringkas bukan. (baca juga: Dasar Dasar HTML)

5. Maintenance mudah

Sekali web yang menggunakan PHP berjalan, programmer dapat dengan mudah melakukan update dari software PHP dengan mudah jika memang diperlukan. karena sifat PHP yang merupakan interpreter. Aplikasi web yang dibuat dengan menggunakan PHP dapat dengan mudah diupgrade versi PHP tanpa harus melakukan kompilasi ulang source code. berbeda sekali dengan bahasa pemrograman lain yang membutuhkan kompilasi ulang jika

melakukan upgrade versi dari bahasa pemrograman. PHP juga dapat berjalan pada berbagai macam web server seperti apache, nginx, dan IIS. (baca juga : Pengertian Web Server Menurut Para Ahli)

6. Open Source

PHP merupakan sebuah projek Open source dengan license yang dikeluarkan oleh PHP group yaitu PHP license V3.01. Inti dari license ini adalah setiap pengguna program PHP bebas menggunakan PHP secara gratis tanpa harus memberikan royalty apapun ke PHP group namun tetap wajib mencantumkan lisensi atas PHP yang dimiliki PHP Group. Dengan kata lain selama pemakai program PHP tidak mengakui produk PHP adalah buatannya maka perjual belian program yang menggunakan PHP diperbolehkan tanpa harus membayar lisensi apapun.

7. Perkembangan Pesa

Karena sifat PHP yang open source, banyak sekali bermunculan projek projek open source besar yang menggunakan PHP seperti Prestashop, WordPress, Drupal, dan lain lain. Hal ini menjadi keunggulan yang sangat besar bagi orang yang menguasai pemrograman PHP. Dengan sangat luasnya perkembangan PHP, maka kesempatan untuk bisnis ataupun kerja pada bidang pemrograman PHP sangatlah luas

2.3.3 Kekurangan Dalam PHP

Seperti yang telah di katakan, Selain memiliki kelebihan PHP, PHP juga memiliki beberapa kekurangan. Namun kekurangannya sangat sedikit. Lalu apa saja kekurangan dari PHP, yaitu diantaranya :

1. Tidak ideal jika untuk pengembangan skala besar.
2. Tidak mempunyai sistem pemrograman berorientasi objek yang sesungguhnya.
3. Tidak dapat memisahkan antara tampilan dengan logik dengan baik (Meskipun penggunaan template bisa memperbaikinya)
4. PHP mempunyai kelemahan security tertentu yang mana jika programmer tidak jeli dalam melakukan pemrograman dan kurang memperhatikan isu dan konfigurasi PHP
5. Kode PHP bisa dibaca semua orang, dan kompilasi hanya bisa dilakukan dengan tool yang mahal dari Zend (\$2000)

2.3.4 Perbedaan Dari beberapa Versi PHP

- PHP 3**

PHP 3.0 dibuat karena melihat kelemahan dari PHP/FI yang digunakan dalam aplikasi e-commerce. Kemudian, mereka menulisnya ulang dengan masih mengacu pada PHP/FI. Setelah PHP 3.0 dikeluarkan, mereka menyarankan untuk menghentikan proyek PHP/FI karena PHP 3.0 masih lebih baik. Alasan untuk mulai mengembangkan PHP, merupakan akronim dari Hypertext Preprocessor, dan memfokuskan diri pada PHP 3.0 ialah pengembangan versi ini secara meluas dalam mendukung berbagai jenis database, protokol dan API. Salah satu kelebihan PHP 3 adalah fleksibilitas fitur, penambahan API dan protokol, kemampuan mengakses berbagai macam database, kemampuan OOP, serta kemudahan syntax bahasa. Kelebihan-kelebihan inilah yang mengundang ribuan developer untuk bergabung dalam proyek PHP 3 untuk menambahkan modul-modul baru. Hal ini juga yang akhirnya merubah singkatan PHP yang pada awalnya ‘Personal Home Page’ dan kemudian secara resmi berganti menjadi ‘PHP Hypertext Processor’. Dengan dukungan yang semakin besar dari berbagai pihak yang menyumbangkan berbagai modul maka, pada tahun 1998, 10% dari seluruh webserver yang ada kala itu telah menginstalasi PHP versi 3.0.

- PHP 4**

PHP versi 4 diluncurkan untuk menangani kelemahan PHP 3, yaitu PHP4 meningkatkan performa aplikasi dan peningkatan kemampuan modular codebase secara keseluruhan. Kurangnya efisiensi waktu dan kinerja buruk diperbaiki dan ditulis ulang dari inti PHP 3. Dengan penambahan fitur baru, seperti session, output buffering dan penanganan input, menjadikan PHP 4 aman dari berbagai jenis bahasa pemrograman berbasis web. Selain itu, inti perbedaan mereka terletak pada penggunaan Engine baru yang dinamakan Zend Engine (dari kata Zeev dan Andi) adalah core dari PHP 4 yg mampu meningkatkan performa aplikasi dan secara resmi di release pada tahun 2000, hampir 2 tahun setelah pertama kali PHP 4 di kembangkan. Zend engine merupakan inti dari PHP. Sebagai bagian dari inti PHP, secara fungsional ia bertugas menangani input, menerjemahkan dan mengeksekusinya. Ia juga berperan menerjemahkan fungsi.

Fitur PHP 4

1. Meningkatkan penanganan pada resource yang digunakan. Kekurangan utama dari PHP 3.0 adalah skalabilitas. Para programmer masih menganggap PHP 3.0 belum mampu digunakan pada aplikasi-aplikasi berskala besar. Hal ini membuat para developer PHP berkerja keras untuk meningkatkan kemampuan penanganan resource, sehingga PHP mampu dijalankan pada aplikasi-aplikasi berskala besar.
2. Dukungan pada pemrograman berorientasi objek. Fitur ini memainkan peranan penting bagi user untuk dapat bekerja dengan bahasa pemrograman berorientasi objek.
3. Dukungan pada penanganan session. Penanganan pada session HTTP sebenarnya juga sudah ada pada versi PHP 3.0. Namun, masih menggunakan dukungan pihak ketiga, yaitu PHPLIB. Nah, pada versi 4 ini, session sudah mampu ditangani langsung oleh PHP. Fitur ini menawarkan interaksi dari website. contoh yang paling umum digunakan adalah pada proses login dan logout.
4. Enkripsi. MCrypt sudah digabung pada distribusi asli. Fitur ini menawarkan algoritma enkripsi diantaranya Blowfish, MD5, SHA1, dan tripleDES.
5. Dukungan ISAPI. Fitur ini menghubungkan webserver IIS (dari Microsoft) dan PHP dan dianggap sebagai modul ISAPI.
6. Dukungan COM/DCOM. Fitur ini memungkinkan PHP untuk mengakses objek COM dan menambah interoperabilitas dengan aplikasi yang berjalan di Windows.
7. Dukungan terhadap Java
8. Perl Compatible Regular Expressions (PCRE) library

• PHP 5

PHP 5 merupakan kelanjutan dari evolusi yang berjalan pada PHP. Walaupun pada PHP 4 sudah banyak library yang ditambahkan, PHP 5 dikembangkan dengan Zend Engine 2.0 dengan kemampuan jauh lebih powerfull dibandingkan PHP 4, terutama di sisi OOP, sehingga kemampuan PHP 5 nantinya dapat disejajarkan

dengan bahasa pemrograman lain yang lebih komplek seperti Java, ASP.NET, dsb. Survey Netcraft pada Februari 2004 melaporkan bahwa 15 juta server telah di install PHP, atau sekitar 32% dari keseluruhan domain yang ada di internet.

Fitur PHP 5

1. Peningkatan dari kemampuan pemrograman berorientasi objek
2. Exception handling, yang menstandarisasi logika atas pemberitahuan kesalahan pemrograman
3. Peningkatan terhadap penanganan string
4. Peningkatan dukungan terhadap XML dan Web Service, yang menggunakan libxml2
5. Dukungan terhadap SQLite database server
6. PHP versi 5 dapat membuat file swf dan applet java

• PHP 7

Di versi PHP 4 menggunakan mesin Zend, di versi PHP 5 menggunakan Zend II, untuk versi PHP 7 sendiri tetap menggunakan mesin Zend tapi menggunakan model mesin baru yang disebut PHPNG atau PHP Next Generation. Model terbaru ini bisa meningkatkan kinerja dua kali lipat penggunaan memori. Perusahaan pengembang membuktikannya dengan menggunakan model PHP terbaru tetapi dengan perangkat pendukung yang lebih sedikit dan tetap melayani pengguna sebelumnya dengan total pengguna yang tidak jauh berbeda.

Operator Tambahan

PHP 7 membuat kejutan dengan menambahkan operator baru di versinya, operator dengan nama **Spaceship** berkode <=>.

Fitur Tambahan

Setelah PHP 7 mendukung 64-bit secara konsisten, masih banyak lagi fitur-fitur lain yang disematkan di versi PHP 7. Diantaranya dukungan untuk Anonymous class, Isset Ternary Operator, Abstract syntax tree, null coalesce operator, lots of libraries, generator return expressions, generator delegation

2.3.5 Tiga Model Penggunaan PHP

Sebelum menginstall, pastikan terlebih tujuan penggunaan PHP. Terdapat 3 model penggunaan PHP, yaitu:

- Website dan Aplikasi Web (Server-Side Scripting)
- Command Line Scripting
- Aplikasi Desktop (GUI)

Model pertama merupakan penggunaan paling umum, dibutuhkan 3 hal: PHP, sebuah server web, dan browser web. Untuk platform Linux dan MacOS X, dapat menggunakan server web Apache, server web IIS digunakan untuk platform Windows. Browser web biasanya sudah ada dalam setiap komputer, terutama yang menggunakan Windows, yaitu Internet Explorer (IE). Selain IE, ada Mozilla FireFox, Google Chrome, Opera, dan lain-lain. Selain itu, dapat juga menyewa space web dari suatu layanan hosting di Internet. Dengan cara ini, yang perlu dilakukan hanya menulis script PHP, mengupload ke server yang disewa, dan melihat hasilnya pada browser. Dalam hal penggunaan PHP dengan cara menginstall sendiri, ada 2 pilihan metode menghubungkan PHP ke server. Pada sebagian server PHP mempunyai Direct Module Interface (SAPI). Termasuk kedalam kelompok server ini adalah Apache, Microsoft Internet Information 6 Server, Netscape dan server iPlanet. Sebagian server yang lain memiliki dukungan untuk ISAPI, Microsoft Module Interface (contohnya OmniHTTPD). Jika PHP tidak memiliki dukungan Modul untuk server web Anda, maka gunakan PHP sebagai processor CGI atau FastCGI. Artinya, server disetup untuk menggunakan CGI executable dari PHP untuk memproses semua file PHP yang diminta pada server. Jika PHP akan digunakan sebagai Command Line Scripting, misalnya untuk menulis script yang membangkitkan gambar secara otomatis secara offline, atau memproses file teks yang dikendalikan oleh beberapa argumen yang dikirimkan kepada script, maka dibutuhkan Command Line Executable. Dalam kasus ini, tidak diperlukan server dan

browser. PHP juga dapat digunakan untuk menulis aplikasi dekstop GUI. Untuk itu diperlukan ekstensi PHP-GTK. Cara ini memiliki pendekatan yang berbeda dari membuat halaman-halaman web, seperti tidak ada output HTML yang dibuat, tetapi mengelola windows dan objek yang berada didalamnya. PHP-GTK tidak disertakan dalam distribusi resmi PHP.

2.3.6 Sintaks PHP

Skrip PHP dieksekusi di server dan hasil eksekusi yang berupa kode HTML dikirim ke computer klien.

1. Simbol Pembuka dan Penutup Skrip PHP

Skrip PHP selalu diawali dengan tanda ‘`<?php`’ dan ditutup dengan ‘`?>`’. Skrip PHP dapat diletakkan dimana saja dalam suatu dokumen HTML. Beberapa server yang sudah diatur konfigurasi directive ‘shorthand-support’, dapat mengawali skrip dengan tanda ‘`<?`’ dan diakhiri dengan ‘`?>`’. Tetapi demi kompatibilitas maksimum, disarankan menggunakan bentuk standar ‘`<?php`’

2. Perluasan File PHP

File PHP harus disimpan dengan perluasan ‘.php’. File PHP biasanya berisi tag-tag HTML dan beberapa kode skrip PHP. Contoh:

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

3. Simbol Pemisah Setiap Perintah PHP

Setiap perintah PHP harus diakhiri dengan simbol semikolon ‘;’. Tugas simbol ini untuk membedakan atau memisahkan satu perintah PHP dengan perintah PHP lainnya.

4. Perintah Mencetak Hasil

Ada dua jenis perintah PHP untuk mencetak keluaran atau hasil yaitu ‘echo’ dan ‘print’. Contoh:

```
9  <html>
10 <body>
11 12
12 <?php
13 echo "Hello World";
14 print "I'm learning PHP Script Language now";
15 ?>
16 </body></html>
```

2.3.6.1 Variabel PHP

Sintaks Perintah Membuat Variabel Suatu variabel digunakan untuk menyimpan suatu nilai, dapat berupa teks, angka, atau array. Variabel dalam PHP menggunakan simbol ‘\$’ di awal namanya. Sintaks perintah membuat variabel:

```
14 $nama_var = nilai;
```

5. Tipe Data dari Variabel

Tipe data variabel tidak perlu dideklarasikan, PHP akan otomatis mengkonversi atau menentukan tipe data variabel berdasarkan nilai yang disimpannya. Contoh:

```
21
22 <?php
23 $nama='Alvina Khansa';
24 $nilai=90;
25 ?>
26
27
```

Catatan (Variabel nama diatas otomatis akan bertipe string, variabel nilai akan bertipe integer).

6. Aturan Pemberian Nama Variabel

- Harus dimulai dengan huruf atau garis bawah (underscore) ‘_’.
- Hanya dapat menggunakan karakter alphanumeric dan underscore
- (A-Z / a-z / 0 - 9, dan _)
- Sebaiknya tidak menggunakan spasi, jika nama variabel terdiri lebih dari satu kata, pisahkan dengan underscore (\$nama_depan, \$nilai_tugas) atau kapitalisasi (\$namaDepan, \$nilaiTugas)

7. Mendeklarasikan/Membuat Variabel

Tidak ada perintah khusus dalam PHP untuk membuat variabel. Sebuah variabel akan terbentuk pada saat sebuah nilai ditentukan kepadanya. Contoh perintah:

```
<?php  
$mobil='Avanza';  
?>
```

Setelah perintah tersebut dieksekusi maka variabel \$mobil akan berisi Avanza.

Catatan: untuk memberikan suatu nilai berupa teks/string kepada sebuah variabel, nilai tersebut harus diapit dengan tanda petik tunggal atau ganda.

8. Scope Variabel

Scope atau ruang lingkup variabel adalah bagian dari skrip yang dapat mereferensikan variabel tersebut. Ada 4 scope variabel dalam PHP:

9. Scope Local

Suatu variabel yang dibuat pada suatu fungsi akan menjadi variabel lokal (memiliki scope local) dan hanya bisa diakses di dalam fungsi. Nama variabel yang sama dapat dibuat dalam fungsi yang berbeda, sebab variabel lokal hanya dikenali oleh fungsi yang membentuk variabel tersebut. Variabel lokal akan dihapus setelah fungsi usai dieksekusi. Contoh:

```
26 <?php
27 function dicoba(){
28 $lokal='Saya hanya bisa diakses dari fungsi dimana saya berada';
29 echo $lokal; //mencetak var lokal
30 }
31 echo $lokal; //akan terjadi error
32 ?>
```

10. Scope Global

Scope global dimiliki oleh variabel yang dibuat diluar fungsi. Variabel dengan scope global dapat diakses dari bagian manapun dari program selama perintah tersebut ditulis diluar suatu fungsi. Variabel global dapat diakses dari dalam suatu fungsi dengan menggunakan kata kunci '**global**'. Contoh:

```
35 <?php
36 $a = 10; // scope global
37 function myTest()
38 {
39 // mengacu ke variabel scope global
40 echo global $a;
41 }
42 myTest();
43 ?>
```

PHP menyimpan semua variabel global dalam array bernama **\$GLOBAL** []. Indeks dari array adalah nama dari variabel-variabel tersebut. Array ini dapat diakses dari dalam fungsi, tetapi juga dapat digunakan untuk mengupdate variabel global secara langsung. Contoh:

```
46 <?php
47 global $a;
48 $a="abc";
49 // adalah sama dengan menggunakan perintah:
50 $GLOBALS["a"]="abc";
51 ?>
```

7. Scope Statik

Ketika suatu fungsi selesai digunakan, secara normal semua variabelnya akan dihapus. Jika diinginkan variabel-variabel tersebut tidak dihapus ketika fungsi selesai dipakai, gunakan kata kunci ‘**static**’ saat membuat variabel. Contoh pembuatan variabel static

```
53 <?php  
54 static $varstatik  
55 ?>
```

variabel \$varStatik sekarang menjadi variabel statik.

2.3.6.2 Operator

Operator Hitung

Operator	Penjelasan	Contoh	Hasil
+	Pertambahan	X=2 Y=x+2	Y=4
-	Pengurangan	x=2 y=5-x	Y=3
*	Perkalian	x=4 y=x*5	y=20
/	Pembagian	y=15/5	y=3
%	Sisa hasil bagi	x=10%5 y=10%8 z=5%2	x=0 y=2 z=1
++	Inkremen	x=5 x++	x=6
--	dekremen	x=5 x--	x=4

Contoh 1

```

59  <?php
60  //menghitung penjualan bersih
61  $jual=100000;
62  17
63  $potongan=0.1;
64  $net=$jual-($jual*$potongan);
65  echo "Penjualan bersih = Rp $net,00";
66  ?>

```

Contoh 2

```

68  <?php
69  //penggunaan operator modulus/sisa hasil bagi
70  $hal=10;
71  if ($hal % 2 == 0)
72  echo 'Halaman genap';
73  else
74  echo 'Halaman ganjil';
75  ?>

```

Contoh 3

```

77  <?php
78  //penggunaan inkrement
79  for($i=1;$i<=100;$i++){
80  echo "$i ";
81  }
82  ?>

```

Operator Penugasan

Operator	Contoh	Sama 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

<code>.=</code>	<code>x.=y</code>	<code>x=x.y</code>
<code>%=</code>	<code>x%-=y</code>	<code>x=x%y</code>

Operator Perbandingan

Operator	Penjelasan	Contoh
<code>==</code>	sama dengan	<code>5==8</code> mengembalikan nilai false
<code>!=</code>	tidak sama dengan	<code>5!=8</code> mengembalikan nilai true
<code><></code>	tidak sama dengan	<code>5<>8</code> mengembalikan nilai true
<code>></code>	lebih besar dar	<code>5>8</code> mengembalikan nilai false
<code><</code>	lebih kecil dari	<code>5<8</code> mengembalikan nilai true
<code>>=</code>	lebih besar dari atau sama dengan	<code>5>=8</code> mengembalikan nilai false
<code><=</code>	lebih kecil dari atau sama dengan	<code>5<=8</code> mengembalikan nilai true

Operator Logika

Operator	Penjelasan	Contoh
<code>&&</code>	and	<code>x=6; y=3; (x < 10 && y > 1)</code> mengembalikan true
<code> </code>	or	<code>x=6; y=3; 19 (x==5 y==5)</code> mengembalikan false
<code>!</code>	not	<code>x=6; y=3; !(x==y)</code> mengembalikan true

2.3.6.3 Struktur Kendali

Seringkali alur eksekusi suatu program tidak berjalan lurus dari baris kode pertama sampai baris terakhir. Kadang-kadang pada suatu baris tertentu

alur program bercabang (struktur keputusan). Di lain baris alur program akan kembali ke baris kode sebelumnya untuk mengulangi sekelompok perintah (struktur perulangan). Struktur keputusan dalam PHP mengenal struktur if dan switch. Untuk menerapkan suatu perulangan baris-baris kode program menggunakan struktur **for**, **while**, **do...while**, dan **foreach**. Berikut ini penjelasan masing-masing struktur tersebut.

2.3.6.4 Struktur Keputusan

Ketika Anda menulis kode program, suatu saat Anda perlu menggunakan mekanisme percabangan sehingga berdasarkan kondisi program Anda akan melakukan aksi yang berbeda. Misalnya saja Anda ingin membuat program yang dapat menampilkan salam yang berbeda tergantung jenis hari pada saat itu, maka kode programnya akan seperti ini:

```
84 <?php  
85 $hari=date("D");  
86 if ($hari=="Mon")  
87 echo "Selamat berlibur";  
88 else  
89 echo "Selamat bekerja dan berkarya";  
90 ?> |
```

Jika program diatas dijalankan pada hari Minggu, maka Anda akan memperoleh ucapan ‘Selamat berlibur’, tapi jika dijalankan selain hari Minggu akan memperoleh ucapan ‘Selamat bekerja dan berkarya’. Ya, contoh diatas adalah salah satu penggunaan struktur kondisi (**if**) yang dimiliki PHP. Struktur **if** mempunyai beberapa format, berikut penjelasannya.

IF

Struktur ini memiliki 3 jenis format pemakaian, yaitu:

- Pertama, bila struktur ini hanya akan mengeksekusi beberapa kode program hanya jika nilai kondisi true, maka format yang digunakan:
if (kondisi) kode_yang_akan_dieksekusi_bila_nilai_kondisi_true
Contoh:

```
<?php  
$hari=date("D");  
if ($hari=="Mon") echo "Selamat berlibur";  
?>
```

- Kedua, baik nilai kondisi if true atau false akan sama-sama mengeksekusi suatu kode program, maka format yang digunakan:

```
if(kondisi)  
kode_yang_akan_dieksekusi_bila_nilai_kondisi_true;  
else  
kode_yang_akan_dieksekusi_bila_nilai_kondisi_false;
```

Contoh:

```
<?php  
$hari=date("D");  
if ($hari=="Mon")  
echo "Selamat berlibur";  
else  
echo "Selamat bekerja dan berkarya";  
?> |
```

- Ketiga, bila ada beberapa kondisi yang perlu dievaluasi, maka format yang digunakan:

```
if(kondisi_1)  
kode_yang_akan_dieksekusi_bila_nilai_kondisi_1_true;  
elseif(kondisi_2)  
kode_yang_akan_dieksekusi_bila_nilai_kondisi_2_true;  
else  
kode_yang_akan_dieksekusi_bila_nilai_kondisi_1_dan_2_false;
```

Contoh:

```
<?php  
$hari=date("D");  
if ($hari=="Mon")  
echo "Selamat berlibur";  
elseif ($hari=="Sat")  
echo "Selamat libur panjang";  
else  
echo "Selamat bekerja dan berkarya";  
?> |
```

Jika kode program yang akan dieksekusi dalam if lebih dari satu, maka kode-kode program tersebut harus dikurung dengan tanda kurung kurawal buka dan tutup seperti ini:

```
<?php  
$hari=date("D");  
if ($hari=="Mon"){  
echo "Hai coy,<br />";  
echo "Selamat berlibur<br />";  
echo "Jangan lupa pesenanku ya";  
}  
?>
```

Switch

Jika dalam program ada banyak pilihan, maka lebih baik menggunakan switch. Sintaksnya:

```
switch(n) {  
case label_1 : pernyataan_yg_dieksekusi_jika_n=label_1;  
    break;  
case label_2 : pernyataan_yg_dieksekusi_jika_n=label_2;  
    break;  
default : dieksekusi_bila_nilai_n_bukan_label2_atau_label_1;  
23  
}  
Contoh:
```

```
<html>
<body>
<?php
$x=1;
switch ($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>
```

Struktur Perulangan (Loop)

PHP mempunyai beberapa struktur perulangan:4

for

Struktur perulangan for sering digunakan dalam pemrograman. Sintaksnya:

```
for (inisialisasi; kondisi; inkremen)
{
kode yang akan ndieksekusi;
}
```

Tiga parameternya:

- **inisialisasi** : nilai awal konter
- **kondisi** : parameter yang akan dievaluasi pada setiap iterasi perulangan. Jika hasil evaluasi true, perulangan akan dilanjutkan, bila false perulangan akan dihentikan
- **inkremen** : nilai pertambahan konter setiap satu iterasi perulangan diselesaikan

Setiap parameter di atas bersifat opsional.

Contoh:

```
<html>
<body>
<?php
for ($i=1; $i<=5; $i++)
{
echo "Bilangan ke-$i: " . $i . "<br />";
}
?>
</body>
</html>
```

Contoh for tanpa parameter kondisi:

```
<html>
<body>
<?php
for ($i=1;; $i++)
25
{
echo "The number is " . $i . "<br />";
if ($i>=5) exit;
}
?>
</body>
</html>
```

Foreach

Perulangan ini disediakan PHP untuk memudahkan kita mengakses elemen-elemen suatu array. Sintaks:

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

Pada setiap iterasi perulangan, nilai dari elemen array yang sedang ditunjuk oleh pointer akan diletakkan ke variabel \$value kemudian pointer akan bergerak menunjuk elemen array berikutnya sehingga pada iterasi selanjutnya, elemen array berikutnya yang akan diolah.

Contoh berikut akan mencetak nilai “satu”, “dua”, dan “tiga” secara berturut-turut:

```
<html>
<body>
<?php
$x=array("satu","dua","tiga");
foreach ($x as $value)
{
echo $value . "<br />";
}
?>
</body>
26
[]/html[]
```

While

Perulangan while akan menjalankan suatu blok kode (sekelompok kode) selama kondisinya bernilai true. Sintaks perulangan ini:

```
while (condition)
{
blok kode yang akan dieksekusi;
}
```

Contoh:

```
<html>
<body>
<?php
$i=1;
while($i<=5)
{
echo "Bilangan ke-$i: " . $i . "<br />";
$i++;
}
?>
</body>
</html>
```

Contoh while di atas akan terus menjalankan blok kodennya selama nilai dari variabel \$i lebih kecil atau sama dengan 5. Dan pada setiap iterasinya, nilai variabel \$i akan bertambah 1 (\$i++).

Do...while

Satu yang unik dari perulangan ini adalah, dia pasti akan melakukan iterasi, minimal satu kali, meskipun nilai kondisinya tidak pernah true. Hal ini disebabkan struktur perulangan yang menyebabkan blok kode akan dijalankan lebih dulu, lalu kondisinya dievaluasi belakangan. Sintaks perulangan ini:

```
Do
{
kode yang akan dieksekusi;
}
while (kondisi);
```

Contoh:

```
<html>
<body>
<?php
$i=1;
do
{
    $i++;
    echo "Bilangan ke-$i: " . $i . "<br />";
}
while ($i<=5);
?>
</body>
</html>
```

Contoh di atas akan menjalankan kode `$i++` dan `echo "Bilangan ke-$i: ". $i."
"`; terlebih dahulu, setelah itu baru kondisinya dievaluasi.

2.3.6.5 Fungsi

Fungsi merupakan salah satu teknik pemrograman modular. Sebuah aplikasi besar disusun dari modul-modul yang berupa sebuah fungsi atau prosedur. Fungsi berisi sekelompok kode dengan tugas dan tujuan spesifik. Fungsi tidak akan dieksekusi ketika program dijalankan. Fungsi hanya akan dieksekusi jika dilakukan pemanggilan terhadapnya. Pemanggilan dapat dilakukan dari mana saja dalam program. Keuntungan teknik ini, modul-modul yang dibuat dapat digunakan berkali-kali (reuse).

1. Membuat Fungsi

Sebuah fungsi dibuat dengan aturan sintaks:

```
function namaFungsi()
{
    kode-kode yang akan dieksekusi;
}
```

Beberapa petunjuk dalam membuat sebuah fungsi:

- Namai fungsi yang menggambarkan fungsinya
- Nama fungsi dimulai dengan huruf atau garis bawah (underscore), tidak boleh angka.

2. Pemanggilan Fungsi

Ketika fungsi sudah dibuat, dia dapat dijalankan dengan cara dipanggil. Pemanggilan suatu fungsi mengikuti pola: nama fungsi lalu diikuti tanda kurung dan nilai parameter jika ada. Contoh:

tambah(10,20);

Memanggil sebuah fungsi bernama tambah dengan nilai parameter 10 dan 20. Jika tidak ada nilai parameter, maka pemanggilan fungsi:

cetak();

3. Parameter Fungsi

Untuk menambah daya guna fungsi dapat ditambahkan parameter fungsi yang tidak lain adalah serupa variabel. Parameter ini dituliskan sesudah nama fungsi didalam tanda kurung. Dengan parameter ini, hasil dari fungsi dapat diatur sesuai dengan keinginan.

Contoh:

```
<html>
29
<body>
<?php
function namaProg ($fprogram) {
echo $fname . "<br />"
}
echo "Bahasa membuat struktur dan konten adalah ".namaProg("HTML");
echo "Unsur interaktif diberikan oleh ".namaProg("Javascript");
echo "Memperindah tampilan fungsi dari ".namaProg("CSS");
echo "Aplikasi web pengolahan data menggunakan ".namaProg("PHP");
?>
</body>
</html>
```

Hasil dari fungsi diatas (nama program) dapat diubah-ubah dengan memberi nilai berbeda pada parameter \$fprogram saat pemanggilan fungsi.

4. Nilai Balik Fungsi

Fungsi dapat diatur agar mengembalikan hasil berupa nilai dengan cara menggunakan kata kunci **return**.

Contoh:

```
<html>
<body>
<?php
function tambah(){
$total = $x + $y;
return $total;
}
echo "$x + $y = " . tambah(5,20);
?>
</body>
</html>
```

Nilai yang dikembalikan pada fungsi diatas adalah jumlah dari nilai variabel \$x dan \$y yang ada didalam variabel \$total. Hasil di layar browser adalah tampilan $5 + 20 = 25$.

2.3.6.6 Array

Array adalah jenis tipe data khusus yang menyimpan sejumlah nilai data dalam sebuah variabel. Nilai-nilai data tersebut, yang disebut elemen array, memiliki indeks yang menunjukkan urutannya dalam array. Elemen array pertama akan memiliki indeks 0, elemen kedua berindeks 1, dan seterusnya. Dengan array, proses pencarian data tertentu akan mudah dilakukan. Misalkan Anda memiliki data nama barang elektronik yang disimpan dalam variabel tunggal seperti ini:

```
$brg1="DVD";
$brg2="Televisi";
$brg3="Lemari es";
```

Bagaimana jika kita ingin mencari barang elektronik tertentu dengan menggunakan perulangan terhadap kelompok variabel tersebut? Tentu akan

reput dengan berbagai kode yang harus kita buat untuk menentukan nomor urut variabel tersebut terlebih dahulu dan sebagainya. Lain halnya jika menggunakan array, kita hanya perlu menggunakan indeks array untuk menemukan barang elektronik yang kita cari.

5. Membuat Array

Membuat array dalam PHP menggunakan fungsi array().

Contoh:

```
$brg = array ("DVD", "Televisi", "Lemari es");
```

Untuk mengakses elemen array tersebut dengan menggunakan indeksnya. Nilai data “Televisi” dapat diakses dengan kode \$brg[1]. Misalnya dicetak, maka perintahnya adalah:

```
echo "Elemen array kedua adalah : ". $brg[1];
```

Array juga dapat dibuat dengan cara menentukan indeksnya langsung seperti:

```
$brg[0] = "DVD";
$brg[1] = "Televisi";
$brg[2] = "Lemari es";
```

6. Jenis Array

Dalam PHP terdapat 3 jenis array, yaitu array numerik (indexed arrays), array assosiatif (associative array), dan array multidimensi (multidimensional array). Array numerik Jenis array yang berindeks numeris seperti contoh sebelumnya. Array assosiatif Jenis array yang memiliki indeks berupa string. Array multidimensi Jenis arary yang indeksnya juga array. Artinya, elemen dari array jenis ini berupa suatu array juga.

Contoh:

```
$mhs = array (
    array ('A12.2010.04567', 'Anita Larasati', 3.5);
    array ('A12.2010.05678', 'Dude Harmono', 3);
    array ('A12.2010.06789', 'Ernawati Listyani', 2.75);
```

Untuk mengakses elemen array multidimensi, misalkan akan dicetak, maka kodenya:

```
<?php  
echo "NIM : " . $mhs[0][0] . "Nama : " . $mhs[0][1] . "IPK : " . $mhs[0][2] . "<br />";  
echo "NIM : " . $mhs[1][0] . "Nama : " . $mhs[1][1] . "IPK : " . $mhs[1][2]; . "<br />";  
echo "NIM : " . $mhs[2][0] . "Nama : " . $mhs[2][1] . "IPK : " . $mhs[2][2];  
?>
```

7. Mencari Panjang Suatu Array

Seperti diketahui array terdiri dari sejumlah elemen array. Untuk mengetahui jumlah elemen dalam suatu array atau panjang suatu array dapat menggunakan fungsi count () .

Contoh:

```
$mobil = array ("Volvo","Jaguar", "Mercedez");  
echo "Panjang array adalah : " . count($mobil);|
```

Fungsi count () diatas akan mengembalikan nilai panjang array yaitu 3.

8. Mencetak Seluruh Elemen Array

Untuk mencetak seluruh elemen array dapat digunakan suatu perulangan for sebagai berikut:

```
$mobil = array ("Volvo","Jaguar", "Mercedez");  
32  
$jum = count($mobil);  
✓ for ($i=0; $i<$jum; $i++) {  
    echo $mobil[$i] . '<br />';  
}|
```

Array Assosiatif

Penulisan indeks untuk jenis array ini bagi Anda yang tidak terbiasa akan terbilang ganjil. Tetapi jika sudah terbiasa tidak akan menjadi masalah.

Kode pembuatan array jenis ini jika menggunakan fungsi array () tampak seperti berikut ini:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
echo "Joni berusia " . $umur("Joni") . " tahun";
```

Tampak dalam menentukan elemen-elemen array diatas menggunakan simbol “=>” yang dibuat dengan simbol sama dengan (“=”) dikombinasikan dengan simbol lebih besar (“>”). Untuk mencetak seluruh nilai elemen array jenis ini dapat menggunakan perulangan foreach, seperti ini:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
foreach ($umur as $x=>$nilaiX) {
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
}
```

5. Menyortir Array

Elemen-elemen array dapat diurutkan baik secara alfabet maupun numeris, menaik (ascending) atau menurun (descending). Berikutnya penjelasan tentang jenis-jenis fungsi pengurutan array dalam PHP.

Fungsi sort ()

Fungsi ini akan mengurutkan elemen secara menaik atau ascending. Jika elemen-elemen array berupa string, maka akan diurutkan menurut urutan alfabet (a, b, c, dan seterusnya). Dan jika berupa nilai numerik, maka elemen array akan diurutkan secara numeris (1, 2, 3, dan seterusnya). Berikut ini contoh keduanya :

Pengurutan terhadap data string:

```
$mobil = array ("Volvo","Jaguar", "Mercedez");
sort($mobil);
$pjg = count($mobil);
for ($i=0; $i<$pjg; $i++) {
    echo $mobil[$i] . "<br />";
}
```

Pengurutan terhadap data numerik:

```
$bilangan = array (5, 7, 3, 4, 2, 1, 6);
sort($bilangan);
$pjg = count($bilangan);
for ($i=0; $i<$pjg; $i++) [
    echo $bilangan[$i] . "<br />";
]
```

Fungsi rsort()

Fungsi ini akan mengurutkan elemen array secara menurun atau descending. Contoh untuk data string dan numerik:

Pengurutan terhadap data string:

```
$mobil = array ("Volvo", "Jaguar", "Mercedez");
rsort($mobil);
$pjg = count($mobil);
for ($i=0; $i<$pjg; $i++) [
    echo $mobil[$i] . "<br />";
]
```

Pengurutan terhadap data numeris:

```
$bilangan = array (5, 7, 3, 4, 2, 1, 6);
rsort($bilangan);
$pjg = count($bilangan);
for ($i=0; $i<$pjg; $i++) [
    echo $bilangan[$i] . "<br />";
]
```

Fungsi assort ()

Fungsi ini digunakan untuk jenis array assosiatif yang akan mengurutkan array secara menaik berdasarkan nilai data elemen array.

Contoh:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
asort($umur);
foreach ($umur as $x=>$nilaiX) [
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
]
```

Karena elemen ketiga memiliki nilai data terkecil (16) maka akan dicetak sebagai yang pertama.

Fungsi ksort ()

Fungsi akan mengurutkan secara menaik array assosiatif berdasarkan nilai indeks elemen array.

Contoh:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
ksort($umur);
foreach ($umur as $x=>$nilaiX) {
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
}
```

Karena elemen kedua memiliki indeks berupa "Indra" yang memiliki urutan alfabetis lebih dahulu ("I" lebih dahulu daripada "J" atau "S") maka akan dicetak sebagai yang pertama.

Fungsi arsort ()

Fungsi ini merupakan gabungan sifat dari fungsi asort() dan rsort() yang akan menghasilkan pengurutan array assosiatif secara menurun (descending).

Contoh:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
arsort($umur);
foreach ($umur as $x=>$nilaiX) [
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
]
```

Akan menghasilkan elemen kedua ("18") dicetak sebagai yang pertama.

Fungsi krsort ()

Fungsi ini gabungan sifat dari fungsi ksort() dan rsort() yang akan menghasilkan pengurutan array assosiatif secara menurun (descending) berdasarkan nilai indeks elemen array.

Contoh:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
arsort($umur);
foreach ($umur as $x=>$nilaiX) {
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
}
```

Akan menghasilkan elemen ketiga (“Susi”) dicetak sebagai yang pertama.

2.3.6.7 Mengelola Data Waktu dan Tanggal

a. Nilai Timestamp

Sebelum mempelajari fungsi-fungsi yang dapat digunakan, pertama kali yang harus kita ketahui adalah informasi waktu dasar yang digunakan dalam memanipulasi data waktu dan tanggal. Informasi ini disebut epoch timestamp atau UNIX timestamp atau timestamp saja. Istilah ini merujuk pada jumlah detik yang dihitung mulai pukul 0:0:0 tanggal 1 Januari 1970 GMT sampai dengan waktu yang kita tentukan. Nilai inilah yang menjadi dasar perhitungan informasi waktu dan tanggal dalam fungsi-fungsi terkait di PHP. Misalnya ingin mencari nilai timestamp pada tanggal 2 Januari 1970 maka nilai timestamp yang kita peroleh adalah 86400 yang berasal dari perhitungan $60 * 60 * 24$. Bagaimana dengan tanggal saat ini, berapa nilai timestampnya?

b. Fungsi-fungsi Waktu dan Tanggal

Nilai timestamp tersebut dapat dimanipulasi untuk menghasilkan informasi tertentu dari waktu dan tanggal. PHP menyediakan banyak fungsi untuk hal ini. Salah satu fungsi dasar yang perlu diketahui adalah date () .

c. Fungsi date ()

Fungsi date () ini akan mengembalikan nilai waktu dan tanggal saat ini dari komputer yang digunakan. Sintaks fungsi: date (format, timestamp) Parameter format wajib digunakan yang akan menentukan format waktu yang ingin dihasilkan. Parameter timestamp bersifat opsional yang menentukan waktu timestamp yang akan dimanipulasi. Nilai default parameter ini adalah waktu timestamp saat ini. Karakter yang dapat digunakan dalam parameter format untuk menghasilkan informasi waktu dan tanggal yang kita inginkan antara lain:

- **d** : menghasilkan nilai hari dalam sebulan (01 s/d 31)
- **m** : menghasilkan nilai bulan dalam setahun (01 s/d 12)
- **Y** : menghasilkan nilai tahun dengan panjang 4 digit.

Karakter lain yang dapat digunakan dalam parameter format antara lain “/”, “.”, atau “-” .

Contoh:

```
echo date("Y/m/d");
echo date("d-m-Y");
✓ echo date("m.d.Y");
```

d. Fungsi mktime ()

Nilai timestamp dapat diperoleh dengan fungsi mktime (). Sintaks fungsi:

Mktime (jam, menit, detik, bulan, tanggal, tahun);

Contoh penggunaan dari fungsi mktime() ini misalkan kita ingin menampilkan informasi tanggal untuk hari besok. Maka kodennya akan seperti berikut ini:

```
$tomorrow = mktime(0, 0, 0, date("m"), date("d")+1, date("Y"));
echo "Besok adalah tanggal " . date("Y/m/d", $tomorrow);
```

Dalam kode diatas pertama ditentukan terlebih dahulu nilai timestamp untuk hari besok. Pada baris berikutnya, nilai tersebut digunakan sebagai parameter kedua dari fungsi date (). Sebagai catatan, untuk menggunakan fungsi ini Anda perlu melihat versi PHP yang Anda gunakan sebab fungsi ini telah dihapus pada PHP versi 5.1.0 keatas. Sebagai gantinya, gunakan fungsi time () berikut ini.

e. Fungsi time ()

Fungsi ini menggantikan fungsi mktime() untuk PHP versi 5.1.0 keatas. Seperti fungsi mktime (), fungsi ini juga akan mengembalikan nilai timestamp. Fungsi time tidak memiliki parameter. Sebagai contoh, dimisalkan akan ditampilkan informasi tanggal seminggu dari saat ini, maka kode programnya:

```
$mgdpn= time() + (7 * 24 * 60 * 60);
echo "Hari ini tanggal : " . date('d-m-Y') . "<br />";
echo "Seminggu lagi tanggal : " . date('d-m-Y', $mgdpn);
```

f. Fungsi getdate ()

Fungsi ini akan mengembalikan nilai dalam bentuk array assosiatif berisi informasi tanggal dari nilai timestamp yang diberikan. Jika tidak menyertakan nilai timestamp, maka waktu lokal (waktu komputer atau server) yang akan digunakan. Array assosiatif yang dihasilkan oleh fungsi berisi sekumpulan informasi tentang waktu yang tersimpan didalam masing-masing elemen array:

"seconds"	berisi nilai numeris menunjukkan detik	0 s/d 59
"minutes"	berisi nilai numeris menunjukkan menit	0 s/d 59
"mday"	berisi nilai numeris menunjukkan hari dalam sebulan	1 s/d 31
"wday"	berisi nilai numeris menunjukkan hari dalam seminggu	0 (sunday) s/d 6 (saturday)

"mon"	berisi nilai numeris menunjukkan bulan	1 through 12
"year"	berisi nilai numeris menunjukkan tahun dengan panjang 4 digit	contoh: 1999 atau 2014
"yday"	berisi nilai numeris menunjukkan hari dalam setahun	0 s/d 365
"weekday"	berisi teks menunjukkan nama hari	Sunday s/d Saturday
"month"	berisi teks menunjukkan nama bulan	January s/d December
0	Jumlah detik sejak dari UNIX epoch, sama seperti nilai yang dikembalikan oleh fungsi time () dan yang digunakan dalam fungsi date ()	Tergantung sistem yang digunakan, biasanya 2147483648 s/d 2147483647.

Elemen array pertama berisi informasi tentang detik dari waktu saat ini yang indeksnya bernama “seconds”, lalu informasi menit pada elemen array kedua dengan indeks “minutes”, demikian dan seterusnya. Jadi, untuk menampilkan informasi waktu dan tanggal dapat menggunakan kode seperti ini:

```
$tgl = getdate();
echo "Waktu saat ini menunjukkan pukul " . $tgl['hours'] . ":" . $tgl['minutes'] . ":" .
| | | | $tgl['seconds'];
```

Hasilnya jika ditampilkan pada layar browser akan menampilkan teks ‘Waktu saat ini menunjukkan pukul 14:7:14’.

g. Fungsi checkdate()

Fungsi ini dapat digunakan untuk memeriksa kebenaran dari data waktu yang diberikan. Sintaks fungsi:

checkdate(bulan, tanggal, tahun)

Parameter bulan dapat berisi nilai 1-12 yang menunjukkan bulan, parameter tanggal dapat berisi 1-31 tergantung bulannya, sedangkan parameter tahun dapat berisi nilai 1-32767. Fungsi akan mengembalikan nilai boolean true (angka 1) jika tanggal valid dan false jika ternyata tanggal tidak valid (misalnya tanggal 30 Pebruari jelas invalid).

Contoh:

```
if (checkdate(2,29,2014)){
    echo "Tanggal valid";
} else {
    echo "Tanggal invalid";
}
```

Kode diatas akan menghasilkan teks ‘Tanggal invalid’ karena tanggal 29 Pebruari 2014 tidak ada.

h. Fungsi date_default_timezone_set ()

Seperti telah dijelaskan diatas, informasi tanggal dan waktu berhubungan dengan berbagai zona waktu dan format tanggal yang berbeda. Jika program manipulasi tanggal yang kita buat tidak menghasilkan informasi waktu yang tepat, misalnya menghasilkan informasi waktu pukul 1 padahal waktu saat itu menunjukkan pukul 9, maka kita perlu mengatur zona waktu yang akan digunakan terlebih dahulu agar sesuai dengan zona waktu tempat kita berada. Fungsi yang dapat digunakan untuk mengatur zona waktu lokal ini adalah date_default_timezone_set (). Sintaks fungsi ini:

```
date_default_timezone_set('nama_zona_waktu')
```

Parameter nama zona waktu merupakan data string yang berbentuk ‘nama kawasan/nama ibukota’. Untuk Indonesia nama zona waktunya adalah ‘Asia/Jakarta’. Contoh penggunaan:

```
$today = date('h:i:s, d-m-Y');
echo $today;
```

Contoh diatas akan menghasilkan informasi waktu ‘03:28:20 18-02-2014’ sementara pada saat program ini dijalankan waktu menunjukkan pukul 10:30:00 dan tanggal 18-02-2014. Disini ada perbedaan informasi waktu yang dihasilkan. Maka agar informasi waktu sesuai dengan waktu saat ini dimana program ini dijalankan (dijalankan di Indonesia), fungsi date_default_timezone_set() perlu digunakan, sehingga kode sebelumnya akan menjadi:

```
date_default_timezone_set('Asia/Jakarta');
$today = date('h:i:s, d-m-Y');
echo $today;|
```

Kode sekarang akan menghasilkan informasi waktu yang sesuai dengan waktu dimana program dijalankan (Dalam hal ini waktu yang sesuai dengan waktu komputer atau server yang digunakan).

Variabel Superglobal

Variabel super global adalah variabel terpasang siap pakai yang selalu tersedia dalam semua lingkup aplikasi web yang dibuat. Tidak seperti variabel biasa yang hanya bisa digunakan pada sebuah file program (misalnya program tambah data mahasiswa) dimana variabel tersebut dibuat yang artinya lingkupnya hanya sebatas program tersebut. Beberapa variabel yang telah ditetapkan dalam PHP adalah superglobal, yang berarti bahwa variabel tersebut selalu dapat diakses, terlepas dari lingkupnya, dan kita dapat mengaksesnya dari fungsi, kelas atau file program tanpa harus melakukan sesuatu yang khusus.

Variabel-variaabel superglobal adalah:

- **`$_GLOBALS`**
- **`$_SERVER`**
- **`$_REQUEST`**
- **`$_POST`**
- **`$_GET`**
- **`$_FILES`**

- **`$_ENV`**
- **`$_COOKIE`**
- **`$_SESSION`**

Variabel **`$GLOBAL`**

`$GLOBAL` adalah variabel super global PHP yang digunakan untuk mengakses variabel global dari mana saja didalam script PHP (juga dari dalam fungsi atau metode). PHP menyimpan semua variabel global dalam sebuah array yang disebut `$GLOBALS` [index]. Indeks berupa nama variabel. Contoh berikut memperlihatkan bagaimana penggunaan `$GLOBAL` ini:

```
?php
$x = 75;
$y = 25;
function addition()
{
    $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}
41
addition();
echo $z;
?>
```

Dalam contoh di atas, karena `z` adalah variabel dalam array `$GLOBALS`, maka variabel ini dapat diakses dari luar fungsi.

Variabel **`$_SERVER`**

`$_SERVER` adalah variabel super global PHP yang menyimpan informasi tentang header, path, dan lokasi file program atau script. Berikut contoh penggunaan `$_SERVER`:

Berikut contoh penggunaan `$_SERVER`:

```

?>php
echo $_SERVER['PHP_SELF'];
echo "<br>";
echo $_SERVER['SERVER_NAME'];
echo "<br>";
echo $_SERVER['HTTP_HOST'];
echo "<br>";
echo $_SERVER['HTTP_REFERER'];
echo "<br>";
echo $_SERVER['HTTP_USER_AGENT'];
echo "<br>";
echo $_SERVER['SCRIPT_NAME'];
?>

```

Tabel dibawah ini berisi daftar elemen-elemen yang paling sering digunakan didalam `$_SERVER`

Elemen	Deskripsi
<code>\$_SERVER['PHP_SELF']</code>	mengembalikan nama file script yang sedang berjalan
<code>\$_SERVER['GATEWAY_INTERFACE']</code>	mengembalikan versi CGI yang digunakan server
<code>\$_SERVER['SERVER_ADDR']</code>	mengembalikan alamat IP server host
<code>\$_SERVER['SERVER_NAME']</code>	mengembalikan nama dari server host (misalnya www.w3schools.com)
<code>\$_SERVER['SERVER_SOFTWARE']</code>	mengembalikan string yang mengidentifikasi server (misalnya Apache/2.2.24)
<code>\$_SERVER['SERVER_PROTOCOL']</code>	mengembalikan nama dan versi dari protokol informasi (misalnya HTTP/1.1)
<code>\$_SERVER['REQUEST_METHOD']</code>	mengembalikan metode request/permintaan yang digunakan untuk mengakses halaman (misalnya POST)
<code>\$_SERVER['REQUEST_TIME']</code>	mengembalikan timestamp dari awal permintaan (misalnya 1377687496)

<code>\$_SERVER['QUERY_STRING']</code>	mengembalikan string query jika halaman diakses melalui sebuah string query
<code>\$_SERVER['HTTP_ACCEPT']</code>	mengembalikan header Accept dari request saat ini
<code>\$_SERVER['HTTP_ACCEPT_CHARSET']</code>	mengembalikan header Accept_Charset dari request saat ini (misalnya utf-8, ISO-8859-1)
<code>\$_SERVER['HTTP_HOST']</code>	mengembalikan header Host dari request saat ini
<code>\$_SERVER['HTTP_REFERER']</code>	mengembalikan URL lengkap dari halaman saat ini (tidak dapat diandalkan karena tidak semua user agent (browser) mendukungnya)
<code>\$_SERVER['HTTPS']</code>	adalah script yang dikirimkan melalui protokol HTTP yang aman
<code>\$_SERVER['REMOTE_ADDR']</code>	mengembalikan alamat IP dari perangkat yang digunakan pengguna untuk mengakses halaman saat ini
<code>\$_SERVER['REMOTE_HOST']</code>	mengembalikan nama host yang digunakan pengguna untuk mengakses halaman saat ini
<code>\$_SERVER['REMOTE_PORT']</code>	mengembalikan port yang digunakan pada perangkat pengguna untuk berkomunikasi dengan server web
<code>\$_SERVER['SCRIPT_FILENAME']</code>	mengembalikan nama path absolut dari script yang dijalankan saat ini
<code>\$_SERVER['SERVER_ADMIN']</code>	mengembalikan nilai yang diberikan ke direktif SERVER_ADMIN dalam

	file konfigurasi server web (jika script anda berjalan diatas host virtual, maka akan berisi nilai yang didefinisikan untuk virtual host tersebut.
<code>\$_SERVER['SERVER_PORT']</code>	mengembalikan port pada komputer server yang digunakan oleh server web untuk berkomunikasi (misalnya 80)
<code>\$_SERVER['SERVER_SIGNATURE']</code>	mengembalikan versi server dan nama host virtual yang ditambahkan ke halaman yang digenerate oleh server
<code>\$_SERVER['PATH_TRANSLATED']</code>	mengembalikan path berbasis file sistem yang mengarah ke script saat ini
<code>\$_SERVER['SCRIPT_NAME']</code>	mengembalikan path script saat ini
<code>\$_SERVER['SCRIPT_URI']</code>	mengembalikan URI dari halaman web saat ini

Variabel `$_REQUEST`

`$_REQUEST` digunakan untuk mengumpulkan data setelah pengiriman (submitting) sebuah formHTML terjadi. Contoh di bawah ini memperlihatkan sebuah form dengan field input dan tombol submit. Ketika pengguna mengirimkan data dengan mengklik "Submit", maka data form dikirim ke file yang ditentukan dalam atribut action dari tag `<form>`. Dalam contoh ini, kami arahkan pengiriman ke file ini sendiri untuk mengolah data form. Jika Anda ingin menggunakan file PHP lain untuk memproses data form, ganti nilai atribut action dengan nama file pilihan Anda. Setelah pengiriman, maka kita bisa menggunakan variabel super global `$_REQUEST` untuk mengumpulkan nilai dari field input:

```
<html>
<body>
<form method="post" action=<?php echo $_SERVER['PHP_SELF'];?>>
Name: <input type="text" name="fname">
<input type="submit">
</form>
<?php
$name = $_REQUEST['fname'];
echo $name;
44
?>
</body>
</html>
```

Variabel \$_POST

\$_POST secara luas digunakan untuk mengumpulkan data form setelah proses pengiriman sebuah form HTML dengan method="post". \$_POST juga banyak digunakan untuk melewatkkan variabel. Contoh di bawah ini memperlihatkan sebuah form dengan field input dan tombol submit. Ketika pengguna mengirimkan data dengan mengklik "Submit", maka data form dikirim ke file yang ditentukan dalam atribut action dari tag <form>. Dalam contoh ini, kami arahkan pengiriman ke file ini sendiri untuk mengolah data form. Jika Anda ingin menggunakan file PHP lain untuk memproses data form, ganti nilai atribut action dengan nama file pilihan Anda. Setelah pengiriman, maka kita bisa menggunakan variabel super global \$_POST untuk mengumpulkan nilai dari field input:

```
<html>
<body>
<form method="post" action=<?php echo $_SERVER['PHP_SELF'];?>>
Name: <input type="text" name="fname">
<input type="submit">
</form>
<?php
$name = $_POST['fname'];
echo $name;
?>
</body>
</html>
```

Variabel \$_GET

\$_GET juga dapat digunakan untuk mengumpulkan data form yang dikirimkan oleh form HTML dengan method="get". \$_GET Juga dapat digunakan untuk mengumpulkan data yang dikirim melalui sebuah URL. Anggaplah kita memiliki halaman HTML yang berisi hyperlink dengan parameter.

```
<html>
<body>
<a href="test_get.php?subject=PHP&web=W3schools.com">Test $GET</a>
</body>
</html>
```

Ketika pengguna mengklik link "Test \$GET", maka parameter subject dan web dikirim ke file test_get.php, kemudian nilai-nilai parameter tersebut di file test_get.php dapat diakses dengan \$_GET.

Berikut ini kode file test_get.php yang digunakan:

```
<html>
<body>
<?php
echo "Belajar " . $_GET['subject'] . " di " . $_GET['web'];
?>
</body></html>
```

Form

Seandainya saja tidak ada form HTML, tentulah programmer web akan kelimpungan ketika berusaha mengumpulkan data dari pemakai/pengunjung melalui aplikasinya. Form adalah alat pengumpul data yang utama bagi seorang programmer web. Inilah alasannya mengapa Anda harus menguasai alat ini dengan baik jika ingin menjadi seorang programmer web. Script PHP bekerjasama secara baik dengan form HTML. Elemen-elemen form pada sebuah halaman web secara otomatis akan tersedia pada script PHP.

Berikut contoh sebuah form (form.php):

```
<html>
<body>
<form action="welcome.php" method="post">
Name: <input type="text" name="nama" />
Age: <input type="text" name="usia" />
<input type="submit" />
</form>
</body>
</html>
```

Ketika seorang user mengisi isian data pada form diatas lalu menekan tombol submit, maka data yang diisikan kedalam form akan dikirim ke file PHP (welcome.php) yang ditentukan.

Berikut isi file welcome.php

```
<html>
<body>
Welcome <?php echo $_POST["nama"]; ?>!<br />
You are <?php echo $_POST["usia"]; ?> years old.
</body>
</html>
```

Variabel `$_POST` digunakan untuk mengambil data dari form. Ada satu variabel jenis ini, yaitu `$_GET`, yang akan digunakan jika method form menggunakan get. Catatan: Variabel-variabel `$_POST` dan `$_GET` adalah variabel super global yang disediakan oleh PHP untuk keperluan pengiriman data form. Artinya, ketika tombol submit diklik maka data form akan diletakkan dalam variabel ini secara otomatis. Maka ketika PHP akan mengambil data yang dikirim oleh form tersebut, akan menggunakan variabel-variabel ini untuk mengaksesnya.

Validasi FORM

Input user sebaiknya divalidasi pada sisi browser menggunakan script klien, misalnya javascript, bila memungkinkan. Ini akan mempercepat

proses validasi dan mengurangi komunikasi dengan server. Jika input user akan disimpan kedalam database, perlu dipertimbangkan melakukan proses validasi di server. Cara yang disarankan dalam melakukan proses validasi di server adalah mengirimkan data form ke form itu sendiri. Dalam contoh diatas, isi dari atribut action, yaitu berisi welcome.php, diganti form.php. Kemudian isi file welcome.php ditulis dibawah kode yang ada pada file form.php sehingga isi dari file form.php akan menjadi:

```
<html>
<body>
<form action="welcome.php" method="post">
Name: <input type="text" name="nama" />
Age: <input type="text" name="usia" />
<input type="submit" />
</form>
Welcome <?php echo $_POST["nama"]; ?>!<br />
You are <?php echo $_POST["usia"]; ?> years old.
</body>
</html>
```

Bagian yang dicetak tebal adalah kode yang semula ditulis di file welcome.php

2.4 Hubungan Antara PHP dan HTML

Halaman web biasanya disusun dari kode-kode html yang disimpan dalam sebuah file berekstensi .html. File html ini dikirimkan oleh server (atau file) ke browser, Kemudian browser menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program php, program ini harus diterjemahkan oleh web-server sehingga menghasilkan kode html yang dikirim ke browser agar dapat ditampilkan. Program ini dapat berdiri sendiri ataupun disisipkan di antara kode-kode html sehingga dapat langsung ditampilkan bersama dengan kode-kode html tersebut. Program php dapat ditambahkan dengan mengapit program tersebut diantara tanda. Tanda-tanda tersebut biasanya disebut tanda untuk escaping (kabur) dari kode html. File html yang telah dibubuhinya program php harus diganti ekstensi-nya menjadi .php3 atau php. Php merupakan bahasa pemograman web yang bersifat server-side HTML=embedded scripting, di mana script-nya menyatu dengan HTML dan berada si server.

Artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di server tetapi disertakan HTML biasa. PHP dikenal sebagai bahasa scripting yang menyatu dengan tag HTML, dieksekusi di server dan digunakan untuk membuat halaman web yang dinamis seperti ASP (Active Server Pages) dan JSP (Java Server Pages).

2.5 Bahasa Pemrograman C



Bahasa Pemrograman C adalah sebuah bahasa pemrograman komputer yang bisa digunakan untuk membuat berbagai aplikasi (general-purpose programming language), mulai dari sistem operasi (seperti Windows atau Linux), antivirus, software pengolah gambar (image processing), hingga compiler untuk bahasa pemrograman, dimana C banyak digunakan untuk membuat bahasa pemrograman lain yang salah satunya adalah PHP.

Meskipun termasuk general-purpose programming language, yakni bahasa pemrograman yang bisa membuat berbagai aplikasi, bahasa pemrograman C paling cocok merancang aplikasi yang berhubungan langsung dengan Sistem Operasi dan hardware. Ini tidak terlepas dari tujuan awal bahasa C dikembangkan.

Bahasa pemrograman C dibuat pertama kali oleh Dennis M. Ritchie pada tahun 1972. Saat itu Ritchie bekerja di Bell Labs, sebuah pusat penelitian yang berlokasi di Murray Hill, New Jersey, Amerika Serikat. Ritchie membuat bahasa pemrograman C untuk mengembangkan sistem operasi UNIX. Sebelumnya, sistem operasi UNIX dibuat menggunakan bahasa assembly (assembly language). Akan tetapi bahasa assembly sendiri sangat rumit dan susah untuk dikembangkan.

Dengan tujuan mengganti bahasa assembly, peneliti di Bell Labs membuat bahasa pemrograman B. Namun bahasa pemrograman B juga memiliki beberapa kekurangan, yang akhirnya dilengkapi oleh bahasa pemrograman C. Dengan bahasa C inilah sistem operasi UNIX ditulis ulang.

Pada gilirannya, UNIX menjadi dasar dari banyak sistem operasi modern saat ini, termasuk Linux, Mac OS (iOS), hingga sistem operasi Android.

2.5.1 Fitur dan Keunggulan Bahasa Pemrograman C

Berikut beberapa fitur serta keunggulan bahasa pemrograman C jika dibandingkan dengan bahasa pemrograman lain:

1. C Sebagai Bahasa pemrograman procedural.

Konsep pemrograman prosedural adalah sebuah metode pemrograman yang setiap baris perintah diproses secara berurutan dari baris paling atas hingga baris paling bawah. Selain itu bisa terdapat fungsi tambahan (*function*) yang digunakan untuk menyelesaikan berbagai tugas. Bahasa pemrograman C termasuk ke dalam kelompok ini.

Selain konsep prosedural, terdapat juga konsep pemrograman object (*object-oriented programming*). Di dalam bahasa pemrograman object, setiap tugas akan dijalankan menggunakan *class* dan *object*. Contoh bahasa pemrograman object adalah JAVA.

Bagi pemula, sangat disarankan untuk mempelajari bahasa pemrograman prosedural terlebih dahulu baru kemudian masuk ke dalam bahasa pemrograman object. Ini juga menjadi alasan untuk belajar bahasa C sebelum masuk ke bahasa pemrograman object seperti JAVA.

Beberapa bahasa pemrograman ada yang mendukung konsep prosedural dan object sekaligus, contohnya bahasa pemrograman C++, Python dan PHP.

2. Bahasa C sangat efisien.

Aplikasi yang dibuat menggunakan bahasa C bisa dieksekusi dengan sangat cepat serta berukuran kecil. Ini karena C bisa langsung berkomunikasi dengan hardware, sebuah fitur yang jarang tersedia di bahasa pemrograman modern seperti JAVA, PHP, maupun Python. Akan tetapi, hal ini juga memiliki kelemahan. Bahasa C relatif sederhana dan tidak memiliki fitur-fitur modern seperti garbage collection dan dynamic typing.

3. C adalah portable language.

Maksudnya, bahasa pemrograman C bisa di-compile ulang supaya berjalan di berbagai sistem operasi tanpa perlu mengubah kode-kode yang ada. Aplikasi yang dibuat di Windows dengan bahasa C, bisa dipindahkan ke Linux dengan sedikit atau tanpa modifikasi.

4. C merupakan induk dari Bahasa pemrograman modern.

Bahasa pemrograman C banyak menginspirasi bahasa pemrograman lain, seperti C++, C#, Objective C, PHP, JAVA, JavaScript dan masih banyak lagi. Dengan mempelajari bahasa C, anda akan familiar dan lebih mudah saat berpindah ke bahasa pemrograman lain yang merupakan turunan dari bahasa C.

2.6 Bahasa Pemrograman Java



Java adalah bahasa pemrograman tingkat tinggi yang berorientasi objek dan program java tersusun dari bagian yang disebut kelas. Kelas terdiri atas metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Para pemrogram Java banyak mengambil keuntungan dari kumpulan kelas di pustaka kelas Java, yang disebut dengan *Java Application Programming Interface* (API). Kelas-kelas ini diorganisasikan menjadi sekelompok yang disebut paket (package). Java API telah menyediakan fungsionalitas yang memadai untuk menciptakan applet dan aplikasi canggih. Jadi ada dua hal yang harus dipelajari dalam Java, yaitu mempelajari bahasa Java dan bagaimana mempergunakan kelas pada Java API. Kelas merupakan satu-satunya cara menyatakan bagian eksekusi program, tidak ada cara lain. Pada Java program javac untuk mengkompilasi file kode sumber Java menjadi kelas-kelas bytecode. File kode sumber mempunyai ekstensi *.java. Kompilator javac menghasilkan file bytecode kelas dengan ekstensi *.class. Interpreter merupakan modul utama sistem Java yang digunakan aplikasi Java dan menjalankan program bytecode Java.

Beberapa keunggulan java yaitu java merupakan bahasa yang sederhana. Java dirancang agar mudah dipelajari dan digunakan secara efektif. Java tidak menyediakan fitur-fitur rumit bahasa pemrograman tingkat tinggi, serta banyak pekerjaan pemrograman yang mulanya harus dilakukan manual, sekarang

digantikan dikerjakan Java secara otomatis seperti dealokasi memori. Bagi pemrogram yang sudah mengenal bahasa C++ akan cepat belajar susunan bahasa Java namun harus waspada karena mungkin Java mengambil arah (semantiks) yang berbeda dibanding C++.

Java merupakan bahasa berorientasi objek (OOP) yaitu cara ampuh dalam pengorganisasian dan pengembangan perangkat lunak. Pada OOP, program komputer sebagai kelompok objek yang saling berinteraksi. Deskripsi ringkas OOP adalah mengorganisasikan program sebagai kumpulan komponen, disebut objek. Objek-objek ini ada secara independen, mempunyai aturan-aturan berkomunikasi dengan objek lain dan untuk memerintahkan objek lain guna meminta informasi tertentu atau meminta objek lain mengerjakan sesuatu. Kelas bertindak sebagai modul sekaligus tipe. Sebagai tipe maka pada saat jalan, program menciptakan objek-objek yang merupakan instan-instan kelas. Kelas dapat mewarisi kelas lain. Java tidak mengijinkan pewarisan jamak namun menyelesaikan kebutuhan pewarisan jamak dengan fasilitas antarmuka yang lebih elegan.

Seluruh objek diprogram harus dideklarasikan lebih dulu sebelum digunakan. Ini merupakan keunggulan Java yaitu Statically Typed. Pemaksaan ini memungkinkan kompilator Java menentukan dan melaporkan terjadinya pertentangan (ketidakkompatibelan) tipe yang merupakan barikade awal untuk mencegah kesalahan yang tidak perlu (seperti mengurangkan variabel bertipe integer dengan variabel bertipe string). Pencegahan sedini mungkin diharapkan menghasilkan program yang bersih. Kebaikan lain fitur ini adalah kode program lebih dapat dioptimasi untuk menghasilkan program berkinerja tinggi.

Java menggunakan model pengamanan tiga lapis (three-layer security model) untuk melindungi sistem dari untrusted Java code. Pertama, bytecode verifier membaca bytecode sebelum dijalankan dan menjamin bytecode memenuhi aturan-aturan dasar bahasa Java. Kedua, class loader menangani pemuatan kelas Java ke runtime interpreter. Ketiga, manajer keamanan menangani keamanan tingkat aplikasi dengan mengendalikan apakah program berhak mengakses sumber daya seperti sistem file, port jaringan, proses eksternal dan sistem window.

Platform independence adalah kemampuan program bekerja di sistem operasi yang berbeda. Bahasa Java merupakan bahasa yang secara sempurna tidak bergantung platform. Tipe variabel Java mempunyai ukuran sama di semua platform sehingga variabel bertipe integer berukuran sama tidak peduli dimana program java dikompilasi. Begitu telah tercipta file .class dengan menggunakan kompilator Java di platform manapun, maka file .class tersebut

dapat dijalankan di platform manapun. Jadi “dimanapun dibuat, dimanapun dapat dijalankan”. Slogan ini biasa diringkas sebagai Write Once, Run Anywhere (WORA).

Java termasuk bahasa Multithreading. Thread adalah untuk menyatakan program komputer melakukan lebih dari satu tugas di satu waktu yang sama. Java menyediakan kelas untuk menulis program multithreaded, program mempunyai lebih dari satu thread eksekusi pada saat yang sama sehingga memungkinkan program menangani beberapa tugas secara konkuren. Program Java melakukan garbage collection yang berarti program tidak perlu menghapus sendiri objek-objek yang tidak digunakan lagi. Fasilitas ini mengurangi beban pengelolaan memori oleh pemrogram dan mengurangi atau mengeliminasi sumber kesalahan terbesar yang terdapat di bahasa yang memungkinkan alokasi dinamis.

Java mempunyai mekanisme exception-handling yang ampuh. Exception-handling menyediakan cara untuk memisahkan antara bagian penanganan kesalahan dengan bagian kode normal sehingga menuntun ke struktur kode program yang lebih bersih dan menjadikan aplikasi lebih tegar. Ketika kesalahan yang serius ditemukan, program Java menciptakan exception. Exception dapat ditangkap dan dikelola program tanpa resiko membuat sistem menjadi turun. Program Java mendukung native method yaitu fungsi ditulis di bahasa lain, biasanya C/C++. Dukungan native method memungkinkan pemrogram menulis fungsi yang dapat dieksekusi lebih cepat dibanding fungsi ekivalen di java. Native method secara dinamis akan di-link ke program java, yaitu diasosiasikan dengan program saat berjalan.

Selain itu keuntungan menggunakan bahasa pemrograman Java antara lain. Memori pada Java secara otomatis dilengkapi garbage collector yang berfungsi mendealokasi memori yang tidak diperlukan. Tidak ada lagi upaya yang dilakukan pemrogram untuk melakukan dispose(). Kita tidak lagi dibebani urusan korupsi memori. Java menerapkan array sebenarnya, menghilangkan keperluan aritmatika pointer yang berbahaya dan mudah menjadi salah. Menghilangkan pewarisan jamak (multiple inheritance) diganti fasilitas antarmuka. Dan mudah dijalankan diberbagai platform.

Grafical User Interface (GUI) adalah salah satu kemampuan Java dalam mendukung dan manajemen antarmuka berbasis grafis. Tampilan grafis yang akan ditampilkan terhubung dengan program serta tempat penyimpanan data. Elemen dasar di Java untuk penciptaan tampilan berbasis grafis adalah dua paket yaitu AWT dan Swing. Abstract Windowing Toolkit (AWT), atau disebut juga “Another Windowing Toolkit”, adalah pustaka windowing bertujuan umum

dan multiplatform serta menyediakan sejumlah kelas untuk membuat GUI di Java. Dengan AWT, dapat membuat window, menggambar, manipulasi gambar, dan komponen seperti Button, Scrollbar, Checkbox, TextField, dan menu pull-down.

Penggunaan komponen AWT ditandai dengan adanya instruksi : import java.awt.*; Swing merupakan perbaikan kelemahan di AWT. Banyak kelas swing menyediakan komponen alternatif terhadap AWT. Contohnya kelas JButton swing menyediakan fungsionalitas lebih banyak dibanding kelas Button. Selain itu komponen swing umumnya diawali dengan huruf "J", misalnya JButton, JTextField, JFrame, JLabel, JTextArea, JPanel, dan sebagainya. Teknologi swing menggunakan dan memperluas gagasan-gagasan AWT. Sementara, penggunaan komponen Swing ditandai dengan adanya instruksi : import javax.swing.*;

Beberapa perbedaan AWT dan Swing, AWT merupakan komponen heavyweight (kelas berat) sedangkan Swing lightweight (kelas ringan). Swing memiliki lebih banyak komponen. Fasilitas Swing Look and Feel : Metal, Windows, Motif. Komponen Swing berdasar model-view, yaitu suatu cara pengembangan komponen dengan pemisahan penyimpanan dan penanganan data dari representasi visual data.

Bahasa pemrograman Java merupakan salah satu bahasa pemrograman yang umum digunakan untuk mengembangkan aplikasi basis data yang dibuat menggunakan MySQL.

2.7 Bahasa Pemrograman PYTHON



Python adalah bahasa pemrograman yang populer. Bahasa pemrograman ini dibuat oleh Guido van Rossum dan dikenalkan sejak tahun 1991. Sebelum memulai untuk belajar Python dasar, akan lebih baik untuk memahami dulu

apa itu Python dan bagaimana cara kerjanya. Python termasuk bahasa pemrograman yang mudah untuk dipelajari. Sampai saat ini bahasa pemrograman Python hampir dipakai di segala bidang seperti game, sistem berbasis web, dan bahkan dapat membuat mesin pencari sendiri. Jadi secara umum, bahasa pemrograman ini dipakai dalam pengembangan website, pengembangan software, matematika, dan system scripting.

2.7.1 Hal-hal yang dapat dilakukan dengan python.

Sebelum belajar Phyton lebih jauh, Anda harus mengetahui apa saja yang bisa dilakukan dengan bahasa pemrograman ini.

Berikut ini beberapa hal yang dapat Anda lakukan menggunakan Python:

1. Python dapat menjadi salah satu bahasa pemrograman untuk membangun server ketika Anda membuat website.
2. Ketika Anda membutuhkan proses pembuatan prototipe atau pengembangan perangkat lunak siap produksi, Python dapat Anda andalkan.
3. Python dapat digunakan untuk membuat workflow di dalam pengembangan perangkat lunak.
4. Python dimanfaatkan untuk membaca dan memodifikasi sebuah file di dalam pembangunan sistem database.
5. Python memungkinkan Anda untuk menangani big data dan menjalankan pemrosesan matematika yang komplek.

2.7.2 Belajar python

Meskipun ada banyak sekali bahasa pemrograman di luar sana, akan tetapi saya merekomendasikan Anda untuk belajar Python. Tentu juga ada beberapa kelebihan yang perlu Anda ketahui. Jika dibandingkan dengan bahasa pemrograman lain, berikut kelebihan bahasa pemrograman Python:

1. Python memiliki sintaksis yang sederhana dan lebih mirip dengan Bahasa Inggris.
2. Python dapat berjalan di berbagai macam sistem operasi.
3. Python berjalan di dalam sistem interpreter, artinya bahasa baris kode bahasa pemrograman ini akan segera dieksekusi setelah ditulis.
4. Python dapat diperlakukan dengan cara prosedural, cara berorientasi objek atau cara fungsional.
5. Python memiliki sintaks yang memungkinkan pengembang untuk menulis program dengan ringkas daripada bahasa pemrograman lain.

2.7.3 Komponen Python

Sesudah Anda memastikan Python sudah terinstall dengan baik di perangkat. Langkah selanjutnya adalah melakukan percobaan beberapa eksekusi program Python. Namun sebelum itu akan lebih baik jika mengetahui terlebih dahulu apa saja komponen yang terdapat di dalam Python.

1. Sintaks

Python sintaks dapat dieksekusi langsung dengan mengetikkannya di Command Line. Selain itu, Anda dapat membuat file Python di dalam server menggunakan ekstensi .py dan menjalankannya menggunakan Command Line.

```
>>> print("Hello, World!")Hello, World!
```

2. Komentar

Sama seperti bahasa pemrograman lainnya, Python juga memiliki kode untuk menjadikan baris program menjadi komentar. Anda dapat menggunakan tanda pagar '#' untuk menjadikan baris kode di Python menjadi komentar.

```
# Ini adalah baris komentar di Python.print("Hello, World!")
```

3. Python Identitations

Berbeda dengan bahasa pemrograman lainnya, jika Anda menulis dalam bahasa Python, indentasi –penempatan kalimat atau baris kode– sangat diperhatikan. Python menggunakan indentasi untuk mengindikasikan baris kode.

```
if 9 > 2:    print("Sembilan lebih besar daripada dua!")
```

Namun ketika baris kode dituliskan menjadi satu kolom atau dalam tab yang sama, maka program akan menjadi error. Di bawah ini adalah contoh penulisan yang menghasilkan error.

```
if 9 > 2:print("Sembilan lebih besar daripada dua!")
```

4. Variable

Python juga memiliki Variabel, tidak berbeda dengan bahasa pemrograman lainnya. Variabel ini digunakan untuk proses penyimpanan dan bekerja dengan berbagai tipe data.

Python sendiri punya standar pendeklarasian variabel. Variabel di Python dapat berupa nama singkat (seperti x dan y tadi) atau nama yang lebih mendeskripsikan seperti umur, nama, alamat, dan lain sebagainya. Aturan penamaan variabel di Python seperti:

- Variabel tidak bisa diawali dengan angka,
- Variabel harus diawali dengan huruf, atau karakter garis bawah (underscore),
- Variabel hanya bisa mengandung karakter alfa-numerik dan karakter garis bawah,
- Variabel di *Python case-sensitive*

Namun berbeda dengan bahasa pemrograman lainnya, Python tidak memerlukan inisiasi variabel untuk mendeklarasikan variabel. Ini berarti sebuah variabel terbuat ketika pertama kali Anda menambahkan nilai ke dalamnya.

Contohnya ketika Anda ingin membuat variabel ‘x’ dan ‘y’, Anda tinggal memasukkan nilainya langsung seperti di bawah ini:

```
x = 10y = "Budi"  
print(x)print(y)
```

Perintah di atas akan mengisi variabel ‘x’ dengan nilai ‘5’ dan ‘y’ dengan nilai ‘Budi’. Jadi proses penyusunan baris kode lebih ringkas.

Kemudahan lainnya, Anda tidak perlu mendefinisikan tipe variabel. Python secara otomatis akan memberikan tipe variabel sesuai dengan nilai yang diberikan pada variabel tersebut. Misalnya pada contoh di bawah ini:

```
x = 10 # x bertipe integer y = "Budi" # y bertipe string
```

5. Booleans

Setelah mempelajari variabel bekerja, di bagian ini Anda akan belajar tentang Booleans. Jika Variabel dapat menyimpan bilangan dengan satu tipe data, booleans juga digunakan untuk menyimpan sebuah tipe data, tapi tipe data yang berbeda.

Tipe data di Booleans hanya ‘benar’ atau ‘salah’. Jadi ini mirip dengan saklar lampu, hanya memiliki dua nilai. Anda dapat menggunakan booleans seperti contoh di bawah ini:

```
a = Trueb = False
```

6. Number

Ketika Anda belajar Python number, ada tiga tipe numerik variabel di Python, yaitu int, float, dan complex. Anda mungkin tidak akan pernah menuliskan tipe variabel di setiap pendeklarasianya, karena (seperti yang sudah dijelaskan di atas) Python sudah menginisiasi tipe variabel ketika Anda menambahkan nilai ke dalamnya.

Int, float, dan complex mempunyai range yang berbeda. Int atau bilangan integer adalah bilangan bulat positif atau negatif, tanpa desimal, dengan panjang tak terbatas. Float atau ‘angka floating point’ adalah angka, positif atau negatif, yang mengandung satu atau lebih desimal. Sedangkan complex adalah bilangan kompleks yang ditulis dengan “j” sebagai bagian dari imajiner.

Sebagai contoh, di bawah ini adalah tiga tipe numerik variabel yang berbeda:

```
x = 1 # inty = 2.8 # floatz = 1j # complex
```

Jadi Python akan mengenali dan membedakan tipe setiap variabel pada saat Anda mengisinya dengan sebuah nilai. Sedangkan untuk mengetahui tipe setiap variabel Anda dapat menggunakan fungsi type():

```
print(type(x))print(type(y))print(type(z))
```

7. String

Ketika ingin belajar Python string, Anda hanya perlu menambahkan tanda kutip tunggal atau tanda kutip ganda di antara nilai variabel yang ingin ditambahkan. Misalnya saja ketika Anda ingin menambahkan string “Budi” ke dalam variabel x maka yang perlu Anda lakukan adalah mendeklarasikannya seperti di bawah ini:

```
x = "Budi"# Atau x = 'Budi'  
# "Budi" sama artinya dengan 'Budi'.
```

Sama seperti bahasa pemrograman lainnya, string dalam Python adalah array byte yang mewakili karakter unicode. Namun, Python tidak

memiliki tipe data karakter sehingga satu karakter hanyalah string dengan panjang 1. Ketika Anda ingin mengakses satu karakter di dalam string, yang perlu Anda gunakan adalah menggunakan tanda kurung kotak.

Di bawah ini adalah contoh mengambil karakter kedua pada sebuah string.

```
a = "Hello, World!"  
print(a[1])
```

8. Operator

Bagian terakhir dari pengenalan komponen Python adalah Operator. Selama melakukan proses coding Anda pasti akan membutuhkan operator untuk membuat sebuah alur logika, penghitungan angka, atau yang lainnya.

Operator ini bekerja untuk melakukan operasi pada variabel dan nilai. Dalam bahasa pemrograman Python, terdapat beberapa grup dari operator, seperti operator aritmatika, penugasan (assignment), pembanding (comparison), logika (logical), identitas (identity), keanggotaan (membership), dan bitwise.

Di antara operator lainnya, operator aritmatika sering digunakan. Operator aritmatika ini mengandung beberapa operator. Di bawah ini adalah daftar operator aritmatika secara lengkapnya:

Operator	Nama	Contoh
+	Penambahan	$x + y$
-	Pengurangan	$x - y$
*	Perkalian	$x * y$
/	Pembagian	x / y
%	Modulus	$x \% y$
**	Exponensian (Pangkat)	$x ** y$
//	Floor division	$x // y$

Selain itu, beberapa proses pengembangan juga sering membutuhkan operator yang dapat menetapkan suatu nilai ke dalam variabel. Berikut ini adalah daftar operator assignment yang dapat Anda gunakan di dalam Python:

Operator	Contoh	Persamaan
=	x = 5	x = 5
+=	x += 3	x = x + 3
-=	x -= 3	x = x - 3
*=	x *= 3	x = x * 3
/=	x /= 3	x = x / 3
%=	x %= 3	x = x % 3
//=	x //= 3	x = x // 3
**=	x **= 3	x = x ** 3
&=	x &= 3	x = x & 3
=	x = 3	x = x 3
^=	x ^= 3	x = x ^ 3
>>=	x >>= 3	x = x >> 3
<<=	x <<= 3	x = x << 3

Sedangkan ketika Anda ingin membandingkan antara satu atau beberapa variabel, biasanya membutuhkan operator perbandingan di antaranya. Operator ini terdiri dari enam jenis. Berikut ini adalah operator perbandingan yang digunakan untuk membandingkan dua nilai:

Operator	Nama	Contoh
==	Sama dengan	x == y
!=	Tidak sama dengan	x != y
>	Lebih besar dari	x > y
<	Kurang dari	x < y
>=	Lebih besar atau sama dengan	x >= y
<=	Lebih kecil atau sama dengan	x <= y

9. Python if else

Ketika Anda membutuhkan perbandingan antara kondisi satu dengan yang lain, Python dapat Anda gunakan untuk mendukung kondisi logis dari matematika. Aturan logika ini biasanya digunakan untuk memberikan syarat sebelum sebuah baris program diambil.

Sesuai dengan tabel perbandingan yang sudah dibahas pada bagian sebelumnya, ada enam kondisi logis yang dapat digunakan di Python; sama dengan ($a == b$), tidak sama dengan ($a != b$), kurang dari ($a < b$), kurang dari atau sama dengan ($a \leq b$), lebih besar dari ($a > b$), lebih besar atau sama dengan ($a \geq b$).

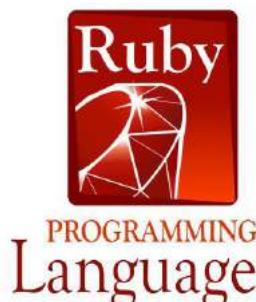
Kondisi ini dapat digunakan dengan beberapa modifikasi, lebih sering digunakan untuk “pernyataan If” dan perulangan. Ketika digunakan untuk “pernyataan If”, contohnya seperti di bawah ini:

```
a = 8  
b = 10  
if a > b:  
    print("b lebih besar dari a")
```

Selain penggunaan “If”, Anda dapat menggunakan “Elif dan Else”. Elif berarti “jika kondisi sebelumnya tidak benar, maka coba kondisi ini. Sedangkan Else menyatakan “lakukan perintah berikut jika semua kondisi tidak sesuai”.

Di bawah ini adalah contoh penggunaan If, Elif, dan Else dalam satu logika.

2.8 Bahasa Pemrograman Ruby



Bahasa Ruby merupakan sebuah bahasa pemrograman yang dinamis, reflektif, general-purposed, dan berbasis objek. Bahasa ini dirancang dan dikembangkan oleh Yukihiro “Matz” Matsumoto di Jepang. Sebelum menentukan namanya, Yukihiro memiliki dua nama untuk digunakan bahasa ini yaitu Rubi dan Coral. Akan tetapi, Yukihiro lebih memilih menggunakan Ruby karena sama dengan batu kelahiran dari salah satu koleganya.

Meskipun mudah dipelajari, bahasa ini memiliki fungsi yang sangat luar biasa diantaranya yaitu dapat digunakan untuk membangun Desktop GUI (Graphic User Interfaces), membuat aplikasi web, atau mengembangkan web itu sendiri. Telah banyak platform yang dibangun menggunakan bahasa ini

mulai dari Hulu, Groupon, Airbnb, dan bahkan Twitter. Hal ini tentunya disebabkan karena kelebihan bahasa pemrograman Ruby tersebut.

Seperti yang dijelaskan sebelumnya, salah satu kelebihan bahasa pemrograman Ruby adalah mudah untuk dipelajari. Bahkan, dalam infografis bahasa pemrograman ini, bahasa pemrograman Ruby dinyatakan sebagai bahasa pemrograman awal terbaik ketika kamu pertama kali belajar pemrograman.

Oleh Yukihiro, bahasa pemrograman ini dikembangkan agar dapat digunakan secara natural dan mudah untuk dimengerti oleh penggunanya. Selain itu, telah terdapat berbagai macam situs yang telah membahas Ruby, yang dapat digunakan sebagai referensi saat mempelajari bahasa ini. Selain mudah untuk dipelajari, kelebihan bahasa pemrograman Ruby lainnya juga memiliki kesamaan dengan kelebihan yang dimiliki bahasa Phyton. Bahasa pemrograman Ruby juga menawarkan berbagai macam library perlengkapan yang menakjubkan serta fungsionalitas yang sangat luas.

Dengan kelebihan ini, para developer dapat menggunakan bahasa pemrograman Ruby dengan sangat mudah dalam membangun berbagai macam hal. Para developer tidak harus membuat segala sesuatunya dari awal, mereka dapat menggunakan beberapa library yang sudah ada sebelumnya dalam membangun sebuah kode.

2.9 Bahasa Pemrograman Cobol



COBOL (Common Business Oriented Language) adalah suatu bahasa computer awam (High Level language) yang berorientasi langsung kepada permasalahan bisnis. Cobol diciptakan pada tahun 1959. pengembangan bahasa cobol selanjutnya dilakukan oleh suatu group yang disebut CODASYL, Singkatan dari Conference On Data System Langguage.

Bahasa Cobol pertama – kali diperkenalkan secara formal pada bulan Januari tahun 1960, versi dari bahasa COBOL ini disebut dengan COBOL -60, dan diperbarui pada tahun 1965 untuk mengatasi hal ini, pada tahun 1968 dan 1974, bahasa cobol dikembangkan dan disempurnakan lebih lanjut dan

distandardisikan dengan nama Ansi Cobol (American National Standards Institute).

Program bahasa COBOL Merupakan Program terstruktur, yaitu program yang strukturnya jelas, mudah dibaca, dan mudah dipelajari, dan baik untuk didokumentasikan , Stuktur Utama dari suatu program Cobol terdiri dari 4 divisi yaitu:

1. IDENTIFICATION DIVISION
2. ENVIRONMENT DIVISION
3. DATA DIVISION
4. PROCEDURE DIVISION

Kalau diinginkan Informal mengenai Identitas program (nama Program, Si pembuat, tanggal dibuat, tanggal dikompilasi, dan lainnya) dapat dilihat pada *identification division* Informasi mengenai Keadaan Komputer dan alat – alat lain yang dipergunakan, dapat dilihat pada *environment division*, Informasi mengenai bentuk, jenis dari data apa saja yang dipergunakan dalam program bersangkutan dapat terlihat pada *data division*, *procedure division* memuat prosedur pemrosesan data yang datanya tampak pada data division untuk dihasilkan outputnya.

Yang dapat dikerjakan COBOL

1. COBOL dibuat untuk operasi yang mencakup langkah dasar Pengolahan
2. data, yaitu membaca data menghasilkan output informasi didalam program COBOL, dua bagian yang utama adalah data division dan Procedure division
3. Dari apa yang dapat dikerjakan oleh cobol, konsep COBOL Orientasinya pada permasalahan yang berhubungan dengan pengolahan data.

Keuntungan Bahasa pemrograman cobol

1. Program COBOL dibuat dalam instruksi bahasa inggris, sehingga lebih mudah dipelajari dan dibuat .
2. program COBOL sesuai untuk pengolahan data, yang banyak diterapkan pada permasalahan bisnis.
3. program COBOL sifatnya standard, sehingga dapat dipergunakan pada computer – computer yang berbeda, tanpa banyak perbedaan.

4. struktur program COBOL jelas, sehingga dapat dimengerti oleh orang seperti akuntan, auditor atau manajer – manajer yang hanya mempunyai pengetahuan pengolahan data yang sedikit.
5. COBOL menyediakan fasilitas listing program , bila mana perlu dapat diperiksa oleh orang lain selain programmernya.
6. mudah didokumentasikan dan dikembangkan bilamana perlu.

2.10 Bahasa Pemrograman Visual Basic



Microsoft Visual Basic adalah suatu paket Bahasa Pemrograman tingkat tinggi yang berbasis Under Windows dengan Orientasi Objek Programming (OOP). Maksudnya, program dapat aktif bila ada respon dari pemakai berupa event atau kejadian tertentu.

Setiap Bahasa Pemrograman yang berbasis objek mempunyai IDE, yaitu suatu tampilan Visual tempat bekerja atau membuat program dalam menyelesaikan suatu proyek.

Visual Basic dikembangkan oleh Microsoft sejak tahun 1991, merupakan pengembangan dari pendahulunya yaitu Bahasa Pemrograman Basic (Beginner of All-Purpose Symbol) sebagai alat bantu untuk membuat berbagai macam program komputer, khususnya menggunakan sistem operasi Windows.

Komponen *Microsoft Visual Basic*

Ketika program Visual Basic telah terbuka, maka akan ditemui banyak komponen yang terdapat di dalamnya, yaitu sebagai berikut :

1. Control Menu

Menu yang digunakan untuk memanipulasi jendela Bahasa Pemrograman, dari menu ini kita bisa mengubah ukuran-ukuran, memudahkan menutup jendela Bahasa Pemrograman.

2. Toolbar

Tombol-tombol yang mewakili suatu perintah tertentu.

3. Toolbox

Komponen-komponen yang akan digunakan dalam perancangan program, terdiri dari :

- a. Label, digunakan untuk menampilkan teks yang tidak dapat diedit oleh user.
- b. Text Box, digunakan untuk memasukan dan mengedit teks.
- c. Option button, merupakan bagian dari Optiongroup, digunakan untuk menampilkan beberapa pilhan dimana user hanya dapat memilih satu.
- d. Image, digunakan untuk menampilkan bitmap, Ikon,file JPG atau GIF.
- e. Chek box, digunakan untuk memilih satu pilhan atau lebih secara bersamaan yang disediakan pada program.
- f. ComboBox, digunakan untuk memilih satu pilhan yang disediakan pada program yang menampilkan seluruh pilhan yang tersedia saat objek tersebut diklik.
- g. Command Button, digunakan untuk membuat suatu aksi saat objek tersebut diklik.
- h. List box,digunakan untuk menampilkan dafatar item dimana user dapat memilih dirinya.
- i. Frame, digunakan sebagai wadah Control atau Toolbox yang lain.
- j. Timer, digunakan untuk mengeksekusi event timer dalam interval waktu tertentu.
- k. Data, digunakan untuk menghubungkan dengan sebuah database dan menampilkan informasinya pada form.

4. Form Windows

Daerah kerja utama pada saat membuat suatu program aplikasi akan otomatis tersedia.

5. Properties Windows

Jendela yang digunakan untuk melihat semua informasi objek yang terdapat pada aplikasi Bahasa Pemrograman Berbasis Objek. Properties adalah sifat dari sebuah objek.

6. Project Explorer

Jendela yang digunakan untuk melihat sebuah informasi tentang tampilan form atau tampilan jendela kode.

7. Code Windows

Jendela tempat menulis perintah yang akan dilaksanakan jika suatu objek dijalankan, jendela berbasis kode-kode yang merupakan instruksi Bahasa Pemrograman Visual Basic.

BAB III

INSTALASI TOOLS YANG DIGUNAKAN

Pada bab ini menjelaskan bagaimana melakukan instalasi tools (alat) yang digunakan untuk membuat sistem.

3.1 XAMPP



XAMPP adalah sebuah paket perangkat lunak (software) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL (dulu) / MariaDB (sekarang), PHP, dan Perl. Sementara imbuhan huruf “X” yang terdapat pada awal kata berasal dari istilah cross platform sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti OS Linux, OS Windows, Mac OS, dan juga Solaris.

Sejarah mencatat, software XAMPP pertama kali dikembangkan oleh tim proyek bernama Apache Friends dan sampai saat ini sudah masuk dalam rilis versi 7.3.9 yang bisa didapatkan secara gratis dengan label GNU (General Public License).

Jika dijabarkan secara gamblang, masing-masing huruf yang ada di dalam nama XAMPP memiliki arti sebagai berikut ini:

X = Cross Platform

Merupakan kode penanda untuk software cross platform atau yang bisa berjalan di banyak sistem operasi.

A = Apache

Apache adalah aplikasi web server yang bersifat gratis dan bisa dikembangkan oleh banyak orang (open source).

M = MySQL / MariaDB

MySQL atau MariaDB merupakan aplikasi database server yang dikembangkan oleh orang yang sama. MySQL berperan dalam mengolah, mengedit, dan menghapus daftar melalui database.

3.1.1 Cara Untuk Instalasi Xampp

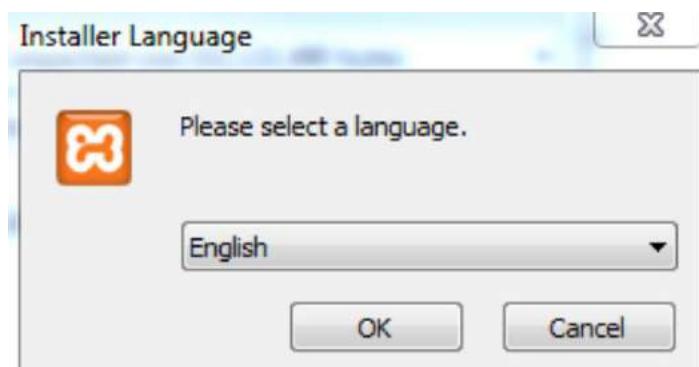
1. Download terlebih dahulu di. Pilihlah salah satu yang paling sesuai dengan laptop atau computer yang kalian miliki

The screenshot shows the XAMPP download page for Windows. At the top, there's a logo of four overlapping windows and the text "XAMPP for Windows 7.2.26, 7.3.13 & 7.4.1". Below this is a table with three rows, each representing a different version of XAMPP for Windows:

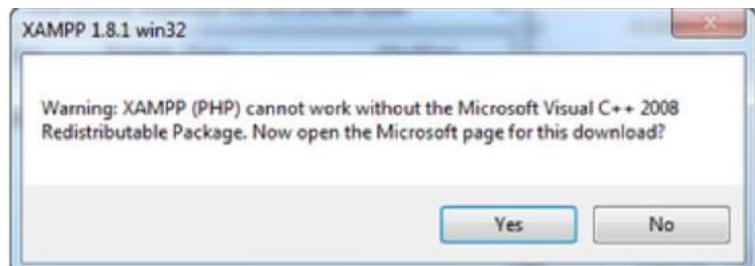
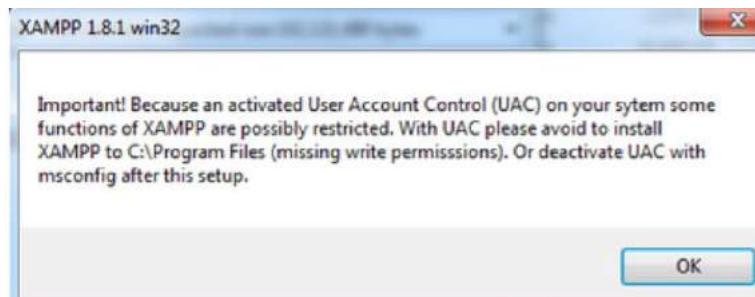
Version	Checksum	Size
7.2.26 / PHP 7.2.26	What's Included? md5 sha1	Download (64 bit) 145 Mb
7.3.13 / PHP 7.3.13	What's Included? md5 sha1	Download (64 bit) 146 Mb
7.4.1 / PHP 7.4.1	What's Included? md5 sha1	Download (64 bit) 146 Mb

Below the table, there are links for "Requirements", "Add-ons", and "More Downloads ». Further down, a note states: "Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms here."

2. Dobel klik file XAMPP yang baru saja Anda download, nanti selanjutnya akan muncul jendela “installer language” seperti di bawah ini:



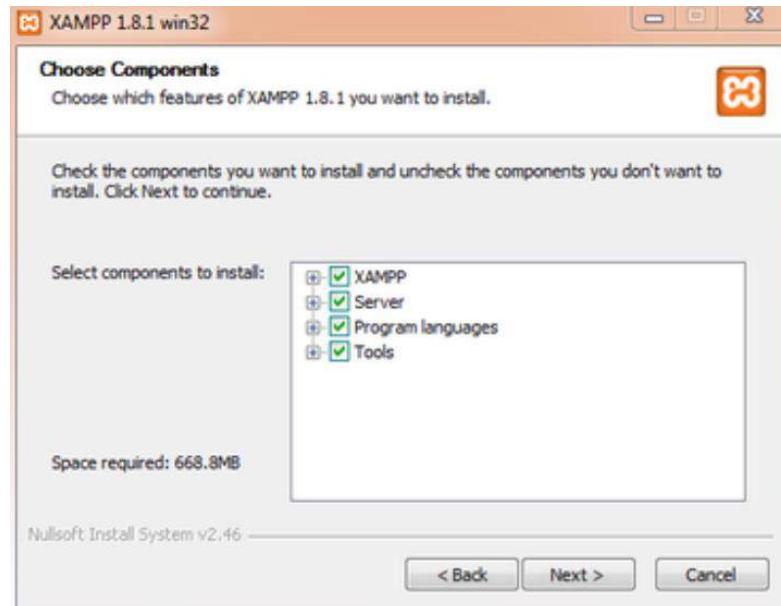
3. Selanjutnya pilih bahasa. Pilih yang Bahasa Inggris (English). Klik OK
4. Kadang pada poses ini muncul error. Jika muncul error tersebut abaikan saja dan lanjutkan dengan klik OK dan YES.



5. Berikutnya akan muncul jendela yang isinya meminta Anda menutup semua aplikasi yang sedang berjalan. Jika semua aplikasi sudah ditutup, maka klik tombol Next.

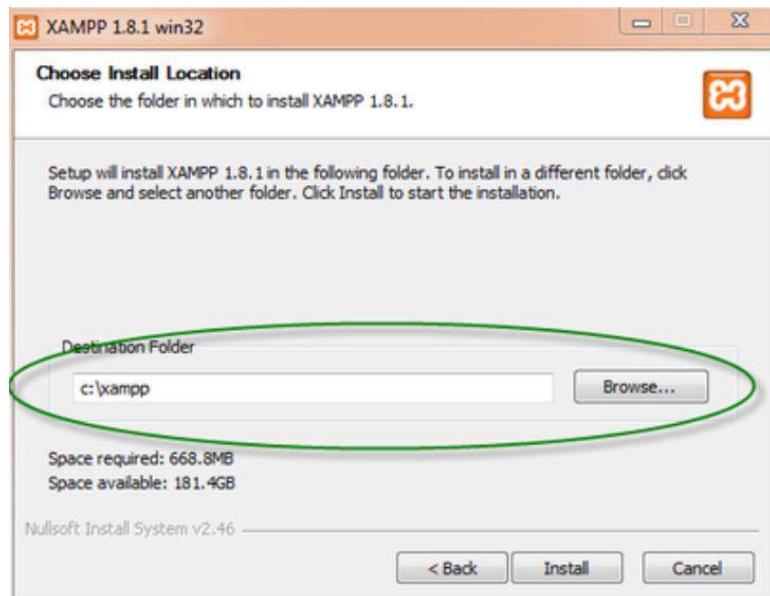


6. Selanjutnya Anda akan diminta untuk memilih aplikasi yang mau diinstal. Centang saja semua pilihan dan klik tombol Next.

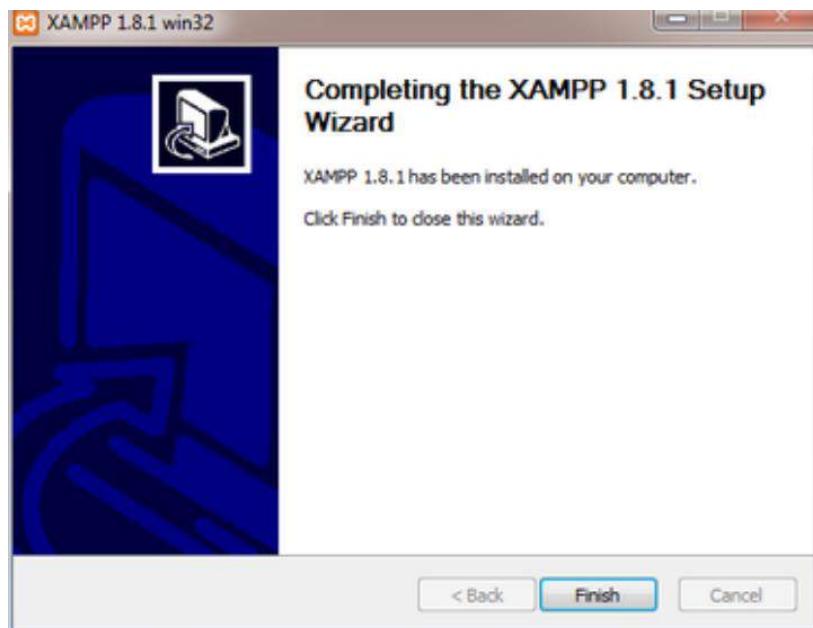


7. Kemudian Anda akan diminta untuk menentukan lokasi folder penyimpanan file-file dan folder XAMPP. Secara default akan diarahkan

ke lokasi c:\xampp. Namun jika Anda ingin menyimpannya di folder lain bisa klik browse dan tentukan secara manual folder yang ingin digunakan. Jika sudah selesai, lanjutkan dan klik tombol Install.



8. Tunggu beberapa menit hingga proses instalasi selesai. Jika sudah muncul jendela seperti di bawah ini, klik tombol Finish untuk menyelesaiakannya.



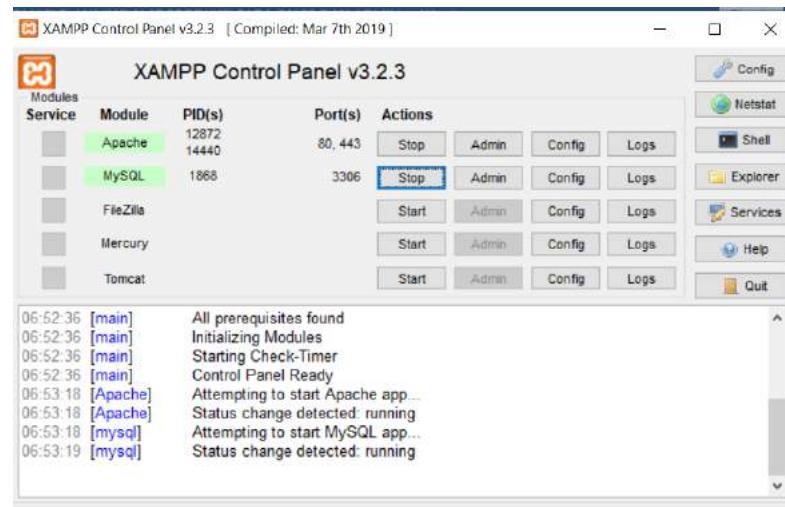
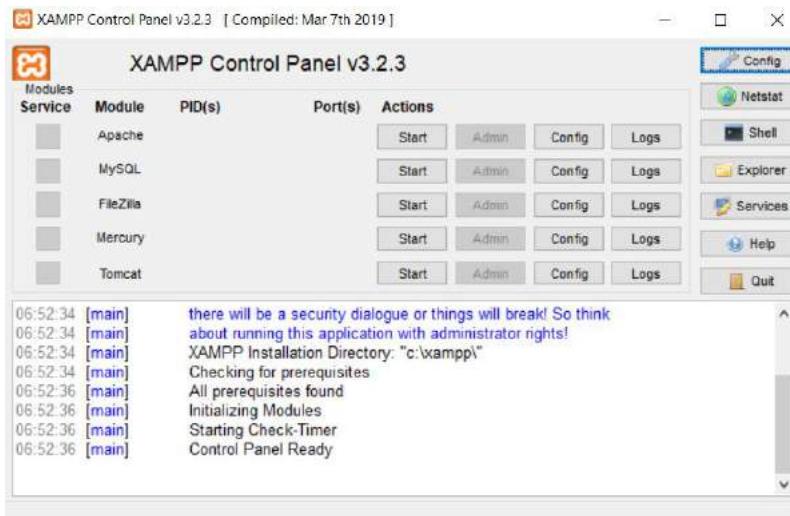
9. Berikutnya, akan muncul jendela dialog seperti gambar di bawah ini yang menanyakan Anda apakah mau langsung menjalankan aplikasi XAMPP atau tidak. Jika ya, maka klik YES.



3.1.2 Cara Menjalankan XAMPP

1. Bukalah aplikasi XAMPP, bisa melalui **Start Menu** atau **Desktop**, dan klik icon XAMPP. Atau, jika Anda membukanya begitu proses instalasi selesai maka klik **Yes** seperti yang terlihat pada gambar di atas.
2. Setelah terbuka, silahkan klik tombol Start pada kolom Action sehingga tombol tersebut berubah menjadi Stop. Dengan mengklik tombol

tersebut, artinya itu adalah aplikasi yang dijalankan. Biasanya jika saya menggunakan XAMPP, yang saya start hanyalah aplikasi Apache dan MySQL, karena saya tidak memerlukan aplikasi seperti Filezilla, dan lain-lain.



3. Sekarang bukalah browser kesukaan Anda, dan coba ketikkan <http://localhost/dashboard> di address bar. Jika muncul tampilan seperti gambar di bawah ini, instalasi telah berhasil.\

Apache Friends

Applications | FAQs | HOW TO Guides | PHP Info | phpBB Index

XAMPP Apache + MariaDB + PHP + Perl

Welcome to XAMPP for Windows 7.3.5

You have successfully installed XAMPP on this system! Now you can start using Apache, MariaDB, PHP and other components. You can find more info in the FAQs section or check the HOW TO Guides for getting started with PHP applications.

XAMPP is meant only for development purposes. It has certain configuration settings that make it easy to develop locally but that are insecure if you want to have your installation accessible to others. If you want have your XAMPP accessible from the internet, make sure you understand the implications and you checked the FAQs to learn how to protect your site. Alternatively you can use WAMP, MAMP or LAMP which are similar packages which are more suitable for production.

Start the XAMPP Control Panel to check the server status.

Community

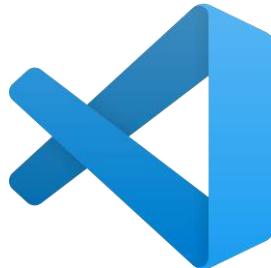
XAMPP has been around for more than 10 years – there is a huge community behind it. You can get involved by joining our forums, adding yourself to the Mailing List, and liking us on Facebook, following our exploits on Twitter, or adding us to your Google+ circles.

Contribute to XAMPP translation at translate.apachefriends.org.

Can you help translate XAMPP for other community members? We need your help to translate XAMPP into different languages. We have set up a site, translate.apachefriends.org, where users can contribute translations.

Install applications on XAMPP using Bitnami

3.2 Visual Studio Code



Microsoft Visual Studio Code adalah one-stop shop yang memungkinkan kita fokus pada proses pengembangan dan melupakan tools baru. Aplikasi Visual Studio Code Text juga merupakan aplikasi editor yang digunakan untuk kode dan teks yang dapat berjalan diberbagai platform operating system dengan menggunakan teknologi Phyton API. Aplikasi ni diciptakan karena terinspirasi dari aplikasiVim, Aplikasi ini sangatlah fleksibel dan powerfull. Aplikasi ini dapat dikembangkan dengan menggunakan Visual Studio Code-packages.

Visual Studio Code Text bukanlah aplikasi opensource dan juga aplikasi yang dapat digunakan dan didapatkan secara gratis, namun beberapa fitur pengembangan fungsionalitas (packages) dari aplikasi ini merupakan hasil dari temuan dan memperoleh dukungan penuh dari komunitas serta memiliki linsensi aplikasi gratis.

Aplikasi ini mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur syntax highlight hampir di semua bahasa pemrogramman yang didukung ataupun dikembangkan oleh komunitas.

Lihat nih beberapa fitur Visual Studio Code:

1. Cross platform

Tersedia di macOS, Linux dan Windows artinya Anda dapat bekerja pada sistem operasi manapun tanpa khawatir belajar coding tools yang sama untuk sistem yang berbeda-beda.

2. Lightweight

Tak perlu menunggu lama untuk memulai. Anda mengontrol sepenuhnya bahasa, tema, debugger, commands dan lain-lainnya sesuai keinginan. Ini dapat dilakukan melalui extention untuk bahasa populer seperti python, node.js, java dan lain-lainnya di Visual Studio Code Marketplace.

3. Powerful editor

Memfungsikan fitur untuk source code editing yang sangat produktif, seperti membuat code snippets, IntelliSense, auto correct, dan formatting.

4. Code Debugging

Salah satu fitur terkeren yang ditawarkan Visual Studio Code adalah membantu Anda melakukan debug pada kode dengan cara mengawasi kode, variabel, call stack dan expression yang mana saja.

5. Source control

Visual Studio Code memiliki integrated source control termasuk Git support in-the-box dan penyedia source code control lainnya di pasaran. Ini meningkatkan siklus rilis proyek Anda secara signifikan.

6. Integrated terminal

Tiada lagi multiple windows dan alt-tabs. Anda dapat melakukan command-line task sekejap dan membuat banyak terminal di dalam editor.

3.2.1 Cara Menginstal Visual Studio Code

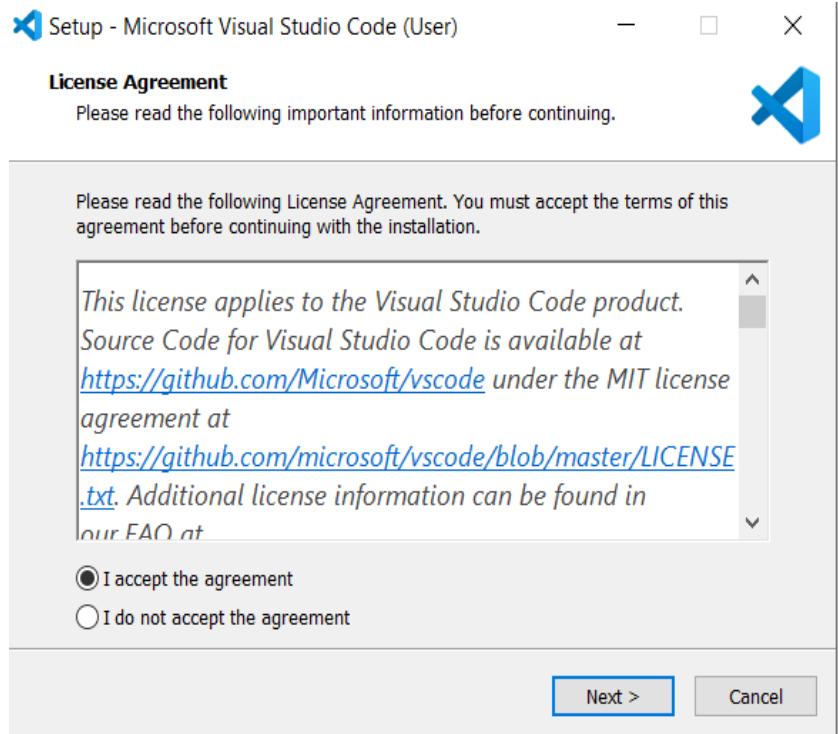
1. Tahapan yang pertama untuk melakukan instalasi maka download dahulu file .exe pada link <https://code.visualstudio.com/download>. Sesuai dengan sistem prosesor apa yang digunakan.



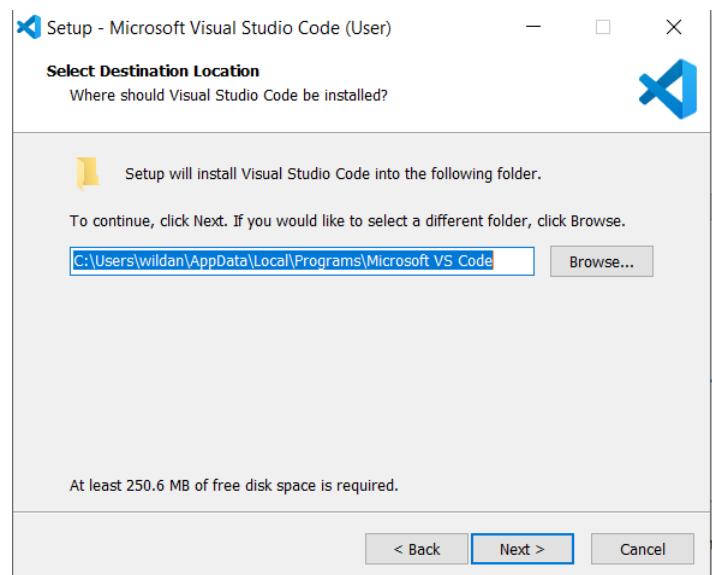
2. Tahapan yang kedua setelah selesai mendownload maka buka directory dimana file .exe tersimpan, kemudian Double klik file .exe hasil download tadi :

VSCodeUserSetup-x64-1.40.1	14/11/2019 0:05	Application	54.575 KB
Wallpaper Engine	04/04/2019 21:43	WinRAR archive	114.056 KB
WebForPC.com-en_visio_professional_20...	13/11/2019 20:28	Application	424.539 KB
WhatsAppSetup	27/10/2018 9:08	Application	79.580 KB
WinRAR.v5.70.kuyhAa.Me	27/02/2019 5:02	WinRAR ZIP archive	4.203 KB
xampp-windows-x64-7.3.5-0-VC15-instal...	06/05/2019 21:37	Application	149.939 KB

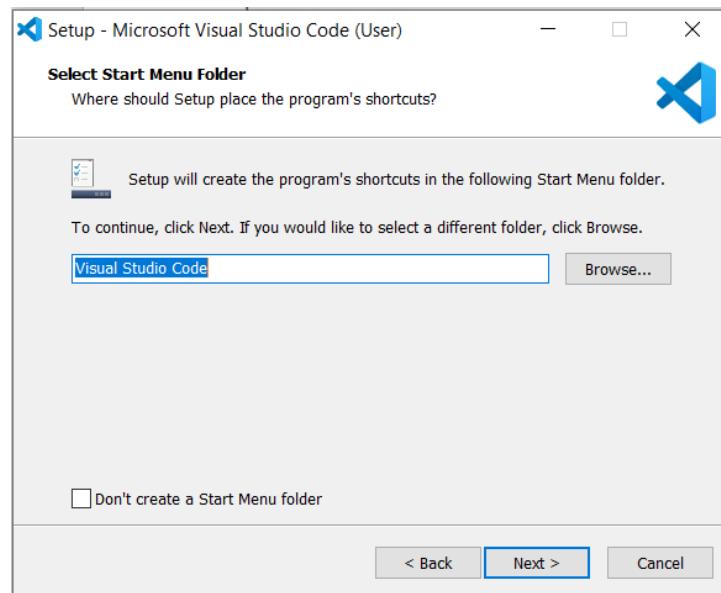
3. Tahapan ketiga jika sudah double klik file .exe maka akan muncul tampilan berikut, kemudian pilih *I accept the agreement* lalu klik next



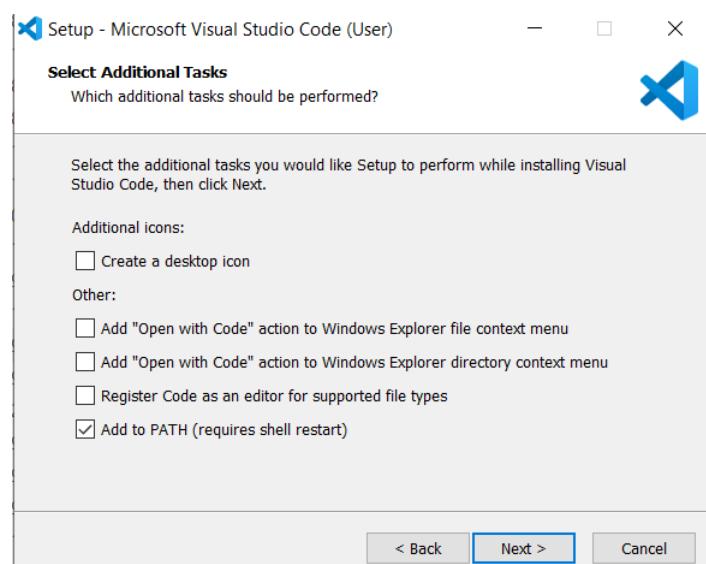
4. Selanjutnya klik next untuk melanjutkan proses instalasi



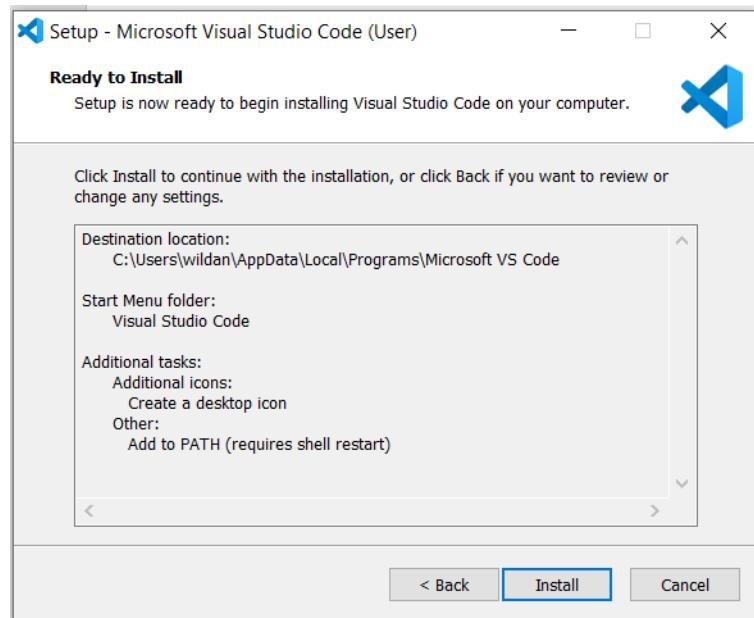
5. Selanjutnya klik next kembali, disini membuat *icon visual studio code* berada pada *start menu folder*



6. Selanjutnya klik next, jika kalian ingin menampilkan di desktop bisa mencentang *create desktop icon*



7. Tekan Install lalu Tunggu sampai proses instalasi selesai, Setelah proses ini selesai maka Visual Studio Code telah bisa digunakan.



8. Ini merupakan tampilan dari Visual studio code yang diamana disini dapat melakukan pembuatan dan pengeditan sintak yang nantinya menjadi aplikasi atau sistem



BAB IV

MENGENAL MYSQL DAN DATABASE

Setelah mempelajari bab ini , pembaca diharapkan dapat membuat program membuat database, dan table.

4.1 Mysql



Mysql adalah sistem manajemen database sql yang bersifat open source dan paling popular saat ini. Sistem Databases Mysql mendukung beberapa fitur seperti multithreaded, multi user, dan sql Database manajemen sistem (DBMS)

4.2 Sejarah Mysql

MySQL:Machine Learning bermula di awal abad 20, seorang penemu Spanyol, Torres y Quevedo, membuat sebuah mesin learning setelah ditemukannya komputer digital. Machine Learning pada dasarnya merupakan proses komputer untuk belajar dari data (Learn from data). Sejak pertama kali komputer diciptakan manusia sudah memikirkan bagaimana caranya agar komputer dapat belajar dari pengalaman. Gagasan ini terbukti yaitu pada tahun 1952, Arthur Samuel menciptakan sebuah program, game of checkers, pada sebuah komputer IBM.

Program tersebut dapat mempelajari gerakan untuk memenangkan permainan checkers dan menyimpan gerakan tersebut kedalam memorinya. Istilah teknologi machine learning pada dasarnya adalah proses komputer untuk belajar dari data (learn from data). Jika tidak ada data, maka komputer tidak akan bisa belajar apa-apa. Maka dari itu ketika kita ingin belajar machine learning, pasti akan terus berinteraksi dengan data. Semua pengetahuan machine learning pasti akan

melibatkan data. Data bisa saja sama, akan tetapi algoritma dan pendekatan nya berbeda-beda untuk mendapatkan hasil yang optimal.

4.3 Kelebihan dari Mysql

Sebagai database server yang memiliki konsep database modern, MySQL memiliki banyak sekali keistimewaan. Berikut ini beberapa keistimewaan yang dimiliki oleh:

1. Portability

MySQL dapat berjalan stabil pada berbagai OS seperti Windows, Linux, Unix, Mac OS, Solaris, Unix, Amiga, HP-UX, Symbian.

2. Open Source "limited"

Dahulu MySQL didistribusikan secara open source (gratis), dibawah lisensi GPL sehingga kita dapat menggunakannya secara cuma-cuma tanpa dipungut biaya. Namun, saat ini karena MySQL telah dibeli oleh SUN, maka kita tidak dapat lagi menikmati fitur-fitur baru yang ada di MySQL, karena SUN akan membatasi fitur-fitur baru ini hanya untuk user yang membeli lisensinya. Sehingga MySQL tidak lagi sebuah opensource yang benar-benar gratis lagi. MySQL sekarang hanya menyediakan fitur-fitur "dasar" saja yang saat ini sudah menggunakan versi 5.1. Untuk mendownloadnya silahkan download di sini dan dicari versi MySQL dengan OS kita.

3. Multiuser

MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami konflik. Hal ini memungkinkan sebuah database server MySQL dapat diakses klien secara bersamaan.

4. Performance Tuning

MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu. Column Types MySQL memiliki tipe kolom yang sangat kompleks, seperti signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year, set serta enum.

5. Command dan Functions

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam query.

6. Security

MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenkripsi.

7. Scalability dan Limits

MySQL mampu menangani database dalam skala besar dengan jumlah records lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu, batas index yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

8. Connectivity

MySQL dapat melakukan koneksi dengan klien menggunakan TCP/IP, Unix soket (Unix), atau Named Pipes (NT).

9. Localisation

MySQL dapat mendetectsi pesan kesalahan (error code) pada klien dengan menggunakan lebih dari dua puluh bahasa

10. Interface

MySQL memiliki interface terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).

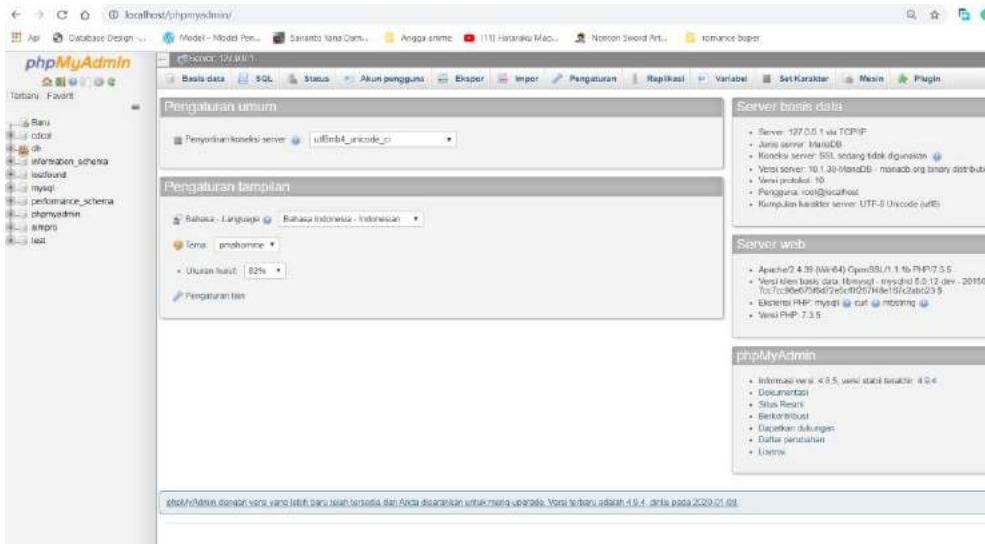
11. Clients dan Tools

MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi database, dan pada setiap tool yang ada disertakan petunjuk online.

12. Struktur Tabel

MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE dibandingkan database lainnya.

Pada saat instalasi aplikasi XAMPP databases Mysql sudah terinstal , jadi kita tinggal menggunakan (tidak install lagi). Untuk mengeceknya , kita menggunakan aplikasi PHPMyAdmin <http://localhost/phpmyadmin>



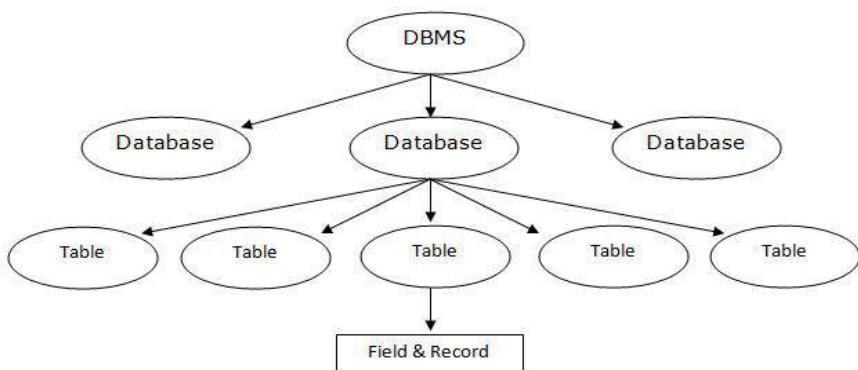
4.4 Pengenalan Basis Data

Basis data adalah atau databases adalah kumpulan informasi yang di simpan dalam computer secara sistematik sehingga dapat diperiksa menggunakan suatu program computer untuk memperoleh informasi

secara sederhana *Database* (basis data) bisa artikan juga sebagai suatu pengorganisasian suatu data dengan bantuan komputer yang membuat data dapat diakses dengan mudah dan cepat. Dalam hal ini, pengertian akses dapat mencakup pemerolehan data maupun pemanipulasi data, seperti menambah dan menghapus data. Manajemen modern mengikutsertakan informasi sebagai sumber daya penting yang setara dengan sumber daya manusia, uang, mesin, dan material. Informasi adalah suatu bentuk penyajian data yang melalui mekanisme pemrosesan, yang berguna bagi pihak tertentu, misalnya manajer. Bagi pihak manajemen, informasi merupakan bahan untuk pengambilan keputusan

4.4.1 Struktur Basis Data

- Database isinya kumpulan tabel yg saling terhubung(dihubungkan oleh kolom yg sama) .
- Tiap tabel berisi baris baris data atau disebut jug record.
- Tiap baris data terdiri dari kolom kolom . Kolom biasanya juga disebut field atau atribut



4.4.2 Tujuan Sistem Basis Data

9. Mencegah data redundancy dan inconsistency.
10. Mempermudah dalam melakukan akses terhadap data.
11. Mempertimbangkan data isolation.
12. Mencegah concurrent access anomaly.
13. Mempertimbangkan masalah keamanan data.
14. Mempertimbangkan masalah integritas.

4.4.3 Keuntungan dan Kerugian Menggunakan Basis Data

Keuntungan Basis Data:

1. Kecepatan dan Kemudahan (Speed)

Mesin mampu mengambil dan mengolah informasi jauh lebih cepat dibandingkan dengan manusia.

2. Efisiensi Ruang Penyimpanan (Space)

Data dikodekan secara elektronik dan disimpan dalam sebuah media.

Misalnya: 1 char = 1 byteHDD

10 GB = ± 10 miliar char

1 hal= 1000 char

Jadi, HDD 10 GB = 10 juta halaman

3. Independensi Program-Data

Struktur data pada database terpisah dengan program. Perubahan struktur data tidak membuat program harus dimodifikasi

4. Meminimalkan Redundansi Data

Redundansi data dapat dikurangi dengan cara data yang sama untuk aplikasi yang berbeda dijadikan satu.

5. Konsistensi Data

Pengurangan redundansi data berimplikasi pada peningkatan konsistensi data (mengurangi kemungkinan untuk tidak konsisten).

6. Berbagi Data

Data dapat diakses oleh banyak pemakai dengan tetap memperhatikan otorisasi. Istilah multiuser menyatakan bahwa sebuah data bisa diakses oleh banyak orang dalam waktu yang bersamaan

7. Standarisasi

Standarisasi seperti nama data, panjang data, kemungkinan nilainya, dan bahkan prosedur untuk mengaksesnya dapat diatur oleh yang berwewenang (DBA)

8. Kualitas Data

Kualitas data sangat berpengaruh terhadap pemerolehan informasi yang berkualitas. Adanya kekangan (constraint) dalam database membuat pelanggaran terhadap isi data oleh pemakai tidak akan ditoleransi oleh sistem dengan sendirinya. Kekangan adalah suatu aturan yang diterapkan pada data dan tidak bisa dilanggar oleh pemakai.

9. Pemeliharaan Program

Perubahan terhadap struktur data dengan berbagai alasan sering kali dilakukan selama tahapan pemeliharaan; Misalnya data baru ditambahkan atau panjang suatu data ditambah. Perubahan seperti ini tidak selalu membuat program-program yang telah jadi harus ikut diubah

Kerugian Basis Data:

1. Mahal
 - Diperlukan hardware tambahan
 - a. CPU yang lebih besar.
 - b. Terminal yang lebih banyak
 - c. Alat untuk komunikasi
 - Biaya performance yang lebih besar
 - a. Listrik
 - b. Personil yang lebih tinggi klasifikasinya
 - c. Biaya telekomunikasi yang antar lokasi / kota2.
2. Kompleks
3. Prosedur backup dan recovery sulit

4.4.4 Konsep dasar Basis Data

Istilah –istilah dasar:

1. Entitas

- Suatu obyek yang dapat dibedakan dari lainnya yang dapat diwujudkan dalam basis data.
- Contoh Entitas dalam lingkungan bank terdiri dari: Nasabah, Simpanan, Hipotik
- Contoh Entitas dalam lingkungan universitas terdiri dari: Mahasiswa, mata kuliahKumpulan dari entitas disebut Himpunan Entitas. Contoh: semua nasabah, semua mahasiswa.

2. Atribut

Atribut adalah sifat atau karakteristik yang melekat dalam sebuah entitas.

Contoh:MAHASISWA = (Nomor_Mhs, Nama_Mhs, Alamat_Mhs, Kota_Mhs)
MOBIL = (Kode_Mobil, Nama_Mobil, Cc)

Nomor_Mhs	Nama_Mhs	Alamat_Mhs	Kota_Mhs
02001	Andi	Jl. Kenaga	yogja
02002	Fikri	Jl. griya Utama	Sleman
02003	Irma	Jl. Kakatua	yogja
02004	Anggi	Jl. Kemenangan	Bantul

3. Record Data

Kumpulan Isi Elemen data yang saling berhubungan.

Contoh: kumpulan atribut npm, nama, alamat, tanggal lahir dari Entitas Mahasiswa berisikan: "10200123", "Sulaeman", "Jl. Sirsak28 Jakarta", "8 Maret 1983".

4. File

Kumpulan dari record yang menggambarkan himpunan Entitas.

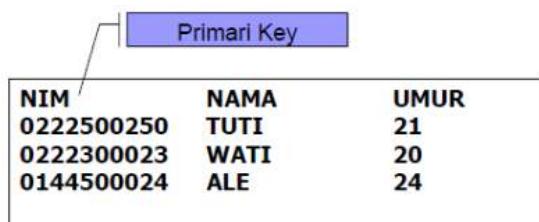
5. Key

Elemen Record yang dipakai untuk menemukan Record tersebut pada waktu akses. Kunci (key) hanya memiliki satu makna dalam teori relasional itu adalah kumpulan dari satu atau lebih kolom yang nilai-nilai gabungan unik di antara semua kejadian dalam tabel tertentu. Utama adalah sarana relasional menetapkan keunikan. Kunci adalah satu atau lebih atribut yang menentukan atribut lainnya atau antara atribut lain saling berkaitan.

Jenis-jenis Key:

a. *Primary Key*

- Field yang mengidentifikasi sebuah record dalam file.
- Bersifat unik



A diagram illustrating a primary key. A blue rectangular box labeled "Primari Key" has a line pointing to the leftmost column of a table. The table has three columns: NIM, NAMA, and UMUR. The data rows are: NIM 0222500250, NAMA TUTI, UMUR 21; NIM 0222300023, NAMA WATI, UMUR 20; and NIM 0144500024, NAMA ALE, UMUR 24.

NIM	NAMA	UMUR
0222500250	TUTI	21
0222300023	WATI	20
0144500024	ALE	24

b. *Secondary key*

- Field yang mengidentifikasi sebuah record dalam file.
- Tidak bersifat unik

NIM	NAMA	UMUR
0222500250	TUTI	21
0222300023	WATI	20
0144500024	ALE	24

Secondary Key

c. *Candidate Key*

- Field-field yang bisa dipilih (dipakai) menjadi primary key

NIM	NAMA	NO_KWIT	JUMLAH
0222500250	TUTI	789	50000
0222300023	WATI	254	60000
0144500024	ALE	365	80000

Candidate key

d. *Alternate Key*

Alternate key adalah candidate key yang tidak terpilih. Misal: dalam suatu entity terdapat dua field yang bisa dijadikan sebagai kunci. Sementara yang boleh dijadikan kunci hanya satu, maka anda harus memilih salah satu. Field yang anda pilih, disebut primary key, sedangkan field yang tidak dipilih disebut dengan alternate key.

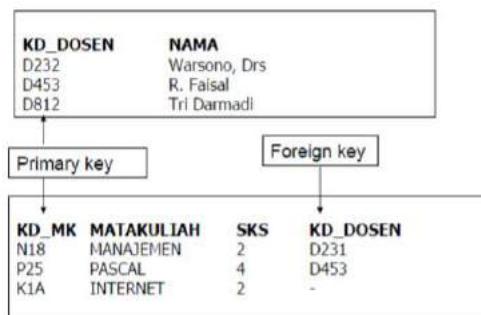
e. *Composite Key*

Composite key adalah kunci yang terdiri dari 2 atau lebih atribut yang secara unik mengidentifikasi suatu kejadian entitas. Setiap atribut yang membentuk kunci senyawa adalah kunci sederhana dalam haknya sendiri.



f. Foreign Key

Field yang bukan key, tetapi adalah key pada file yang lain.



4.4.5 Komponen Basis Data

1. DATA, data tersimpan secara terintegrasi dan dipakai secara Bersama-sama.
2. HARDWARE, perangkat keras yang digunakan dalam mengelola sistem database.
3. SOFTWARE, perangkat lunak perantara pemakai dengan data
4. fisik, perangkat lunak dapat berupa data base management system dan berbagai program aplikasi.
5. USER, sebagai pemakai sistem

a. Data

Ciri-ciri data didalam database:

1. Data disimpan secara terintegrasi (integrated): Database merupakan kumpulan dari berbagai macam file dari aplikasi-aplikasi yang berbeda,yang disusun dengan cara menghilangkan bagian-bagian yang rangkap(redundant).
2. Data dapat dipakai secara bersama-sama (shared): Masing-masing bagian dari database dapat diakses oleh pemakai dalam waktu yang bersamaan, untuk aplikasi yang berbeda.

b. Hardware

Terdiri dari semua peralatan perangkat keras komputer yang digunakan untuk pengelolaan sistem database berupa:

1. Peralatan untuk penyimpanan database, yaitu secondary storage (harddisk, disket, flash disk, CD).
2. Peralatan input (keyboard, scanner, kamera digital) dan output (printer, layar monitor).
3. Peralatan komunikasi data (ethernet card, modem)

c. Software

1. Perangkat lunak dalam hal ini DBMS berkedudukan antara basis data dengan pengguna. perangkat inilah yang berperan melayani kebutuhan-kebutuhan pengguna.
2. Software:
 - Sistem Operasi

Sistem Operasi adalah perangkat lunak sistem yang mengatur sumber daya dari perangkat keras(Hardware) dan perangkat lunak(Software), serta sebagai jurik (daemon) untuk program komputer. Tanpa sistem operasi, pengguna tidak dapat menjalankan program aplikasi pada komputer mereka, kecuali program booting.

Sistem operasi mempunyai penjadwalan yang sistematis mencakup perhitungan penggunaan memori, pemrosesan data, penyimpanan data, dan sumber daya lainnya.

Contoh sistem operasi modern adalah Linux, Android, iOS, Mac OS X, dan Microsoft Windows.

- DMBS
 - ✓ Suatu koleksi data yang saling berelasi dan satu set program untuk mengakses data tersebut.
 - ✓ jd DBMS terdiri dari “Database” dan “set program”. Pengolah untuk menambah data, menghapus data, mengambil dan membaca data. Contoh: File-file mdb yang telah kita buat di MS. Access ditambah dengan program MS. Access sebagai tools, telah menjadikannya sebuah DBMS
- Pogram Aplikasi

d. User

User dapat diklasifikasikan menjadi 3 kategori:

1. Pemrogram Aplikasi adalah orang yang membuat program aplikasi yang sesuai dengan kebutuhan pengguna menggunakan basis data.
2. Administrator Basis Data (DBA) adalah orang yang bertanggung jawab terhadap pengelolaan basis data yang meliputi:
 - Mendefinisikan basis data
 - Menentukan isi basis data
 - Menetukan sekuritas basis data
3. Pengguna Akhir dibagi menjadi dua:

- Pengguna aplikatif yaitu orang yang mengoperasikan program aplikasi yang dibuatoleh pemrogram aplikasi.
- Pengguna Interaktif orang yang dapat menggunakan perintah-perintah beraras tinggi pada antar muka basis data yang tersedia(misalnya SELECT, INSERT) atau melakukanperintah-perintah melalui antar muka basis data.

➤ **Data Independence**

Kemampuan untuk melakukan modifikasi suatu definisi skema pada suatu level tanpa memberikan efek pada skema yang lebih tinggi dikatakan sebagai data independence (ketidaktergantungan data). Dua level dari data independence:

- Physical data independence adalah kemampuan untuk melakukan modifikasi pada level fisik, dengan tujuan untuk meningkatkan performa.
- Logical data independence adalah kemampuan untuk melakukan modifikasi dari skema konsep tanpa menyebabkan perubahan pada program aplikasi.

➤ **Perlunya Prinsip Data Independence**

1. Database administrator dapat merubah isi, lokasi dan organisasi database tanpa mengganggu program aplikasi yang sudah ada.
2. Vendor hardware & software pengelolaan data bisa memperkenalkan produk-produk baru tanpa mengganggu program-program aplikasi yang sudah ada.
3. Untuk memudahkan perkembangan program aplikasi.
4. Memberikan fasilitas pengontrolan terpusat oleh DBA demi security dan integritas data, dengan memperhatikan perubahan-perubahan kebutuhan user.

➤ **Transaction Management**

1. Sebuah transaction adalah sekumpulan operasi yang ditunjukkan sebagai sebuah fungsi logik tunggal dalam sebuah aplikasi basis data.
2. Transaction-management component menjamin bahwa basis data dalam keadaan konsisten (correct), memberitahukan kegagalan system (cth. power failures and operating system crashes) dan kesalahan transaksi.

3. Concurrency-control manager mengontrol interaksi yang berada pada concurrent transaction, supaya menjamin konsistensi dari basis data.

➤ **Database Users**

Users dibedakan dalam caranya berinteraksi dengan sistem:

1. Application programmers (Programer Aplikasi) – berinteraksi dengan sistem melalui pemanggilan DML, yang disertakan (embedded) dalam program yang ditulis dalam bahasa pemrograman induk (C, C++, Java, VB, php).
2. Sophisticated users (Casual user / User Mahir) – berinteraksi dengan sistem tanpa menulis modul program. Mereka menyertakan query (untuk akses data) dengan bahasa query yang telah disediakan oleh suatu DBMS.
3. Specialized users (User Khusus) – menulis aplikasi basis data non konvensional, tetapi untuk keperluan-keperluan khusus, seperti untuk aplikasi AI, Sistem Pakar, Pengolahan Citra, yang bisa saja mengakses basis data dengan / tanpa DBMS yang bersangkutan.
4. Naïve users (End User / User Umum) – berinteraksi dengan sistem basisdata melalui pemanggilan satu program permanen (executable program) yang telah ditulis / disediakan sebelumnya.
5. Contoh, orang yang mengakses basis data dari web, teller bank, staf kasir.

➤ **Database Administrator**

1. Database Administrator mengkoordinasi semua aktifitas basis data.
2. Database Administrator memiliki pengertian yang baik tentang sumber daya informasi perusahaan dan kebutuhannya
3. Satu alasan utama untuk memiliki suatu DBMS adalah untuk memiliki suatu pusat pengendalian terhadap baik data maupun program dalam mengakses data.

4. Orang yang melakukan pengawasan terhadap sistem ini dinamakan Database Administrator.
5. Fungsi dari Database Administrator adalah:
 - a. Schema definition (mendefinisikan skema).
 - b. Storage structure and access method definition (mendefinisikan struktur penyimpanan dan metode akses).
 - c. Schema and physical organization modification (melakukan modifikasi organisasi skema dan fisik).
 - d. Granting user authority to access the database (Memberikan jaminan hak untuk mengakses data).
 - e. Specifying integrity constraints (Membuat spesifikasi untuk kendala integritas).
 - f. Monitoring performance and responding to changes in requirements (melakukan monitor kinerja dan bertanggung jawab dalam perubahan kebutuhan)

4.5 Bahasa Basis Data (*Database Language*)

Bahasa database merupakan bahasa data yang dapat ditempelkan kedalam bahasa pemrograman yang lain, sebut saja Java, Pascal, Fortran dst. Bahasa dimana instruksi data base menempel disebut inang. Beberapa komponen Bahasa data base menurut fungsinya dibagi menjadi:

a. *Data Definition Language*

Data definition language adalah sekumpulan definisi yang disimpan di dalam data dictionary.

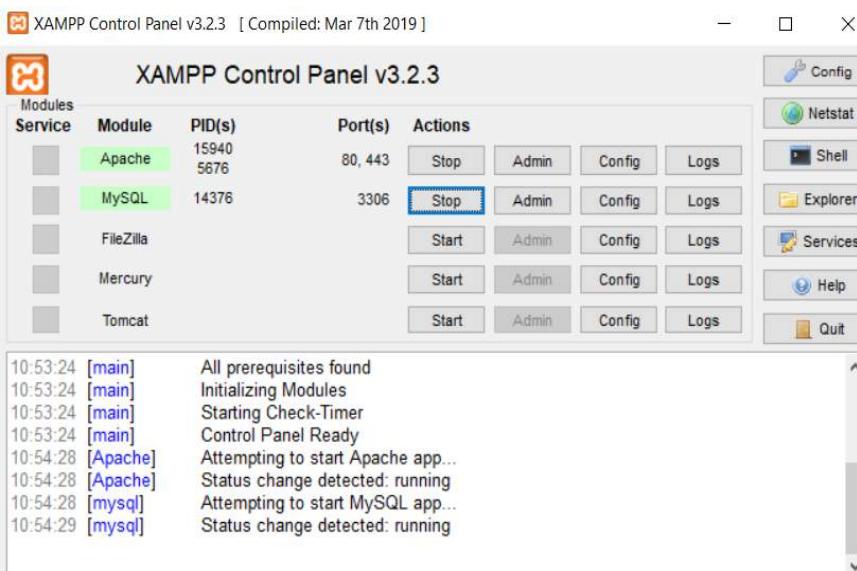
b. *Data Manipulation Language*

Data Manipulation Language berisi akumulasi dari operasi manipulasi basis data yang dilakukan. Ini biasa disebut dengan bahasa query sebab biasanya digunakan untuk meminta informasi yang ada dari basis data tersebut.

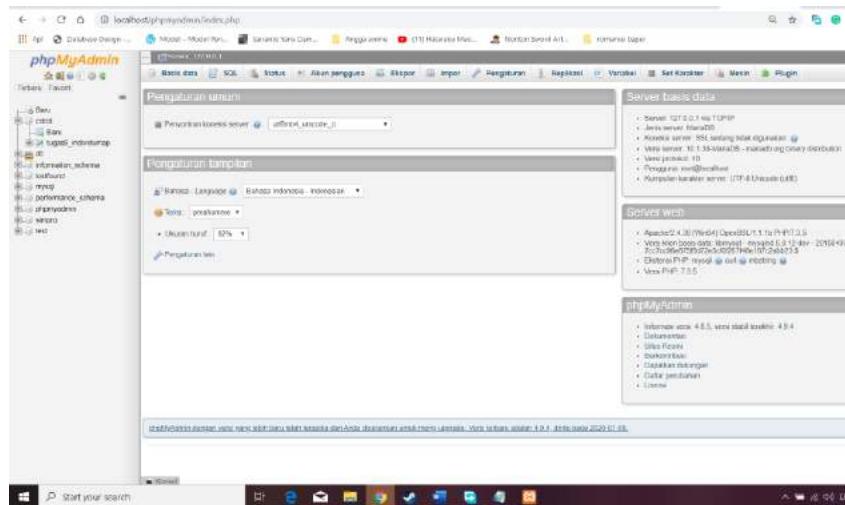
4.6 Cara Membuat Database dan Tabel

4.6.1 Cara Membuat Database di Xampp

1. Instalasi terlebih dahulu xampp (jika ingin mengetahui cara instalasi bisa dilihat di bab sebelumnya)
2. Selanjutnya, buka Xampp control panel dan start apache dan Mysqlnya itu digunakan agar dapat mengaktifkan dan dapat menjalankan xampp pada browser



3. Lalu kita harus masuk ke phpmyadmin di XAMPP. Ketik di browser alamat <http://localhost/phpmyadmin> sehingga muncul tampilan seperti ini.



4. Pada menu Basis data, silahkan masukkan nama database yang diinginkan, kemudian klik Create. Ingat-ingat nama database tersebut ya, karena kita akan menggunakannya nanti. Jika perlu, tuliskan di aplikasi notepad supaya mudah bagi kita untuk mengingatnya.

Catatan : database yang saya buat diberi nama db_mahasiswa

Basis data	Penyortiran	Tindakan
cdcol	latin1_swedish_ci	Check privileges
db_malaria	latin1_swedish_ci	Check privileges
db_prospek	latin1_swedish_ci	Check privileges
db_sales	latin1_swedish_ci	Check privileges
db_simpro	latin1_swedish_ci	Check privileges
information_schema	latin1_swedish_ci	Check privileges

5. Hasil database yang dibuat akan berada di tampilan awal seperti gambar di bawah ini



4.6.2 Cara *Drop* (Menghapus) Database yang dibuat

1. Pilih Menu Basis Data, Lalu pilih database yang ingin dihapus
Catatan; jika ingin menghapus / drop databases harus dipikirkan terlebih dahulu karena menghapus databases berarti menghapus fungsi dan table yang dibuat di databases yang telah dibuat

Basis data

Basis data	Penyortiran	Tindakan
<input type="checkbox"/> edcol	latin1_swedish_ci	=> Check privileges
<input checked="" type="checkbox"/> db_mahasiswa	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> db_malaria	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> db_prospek	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> db_sales	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> db_simpro	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> information_schema	utf8_general_ci	=> Check privileges
<input type="checkbox"/> lostfound	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> mysql	utf8_general_ci	=> Check privileges
<input type="checkbox"/> performance_schema	utf8_general_ci	=> Check privileges
<input type="checkbox"/> phpmyadmin	utf8_general_ci	=> Check privileges
<input type="checkbox"/> simpro	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> test	latin1_swedish_ci	=> Check privileges

Jumlah: 13

Pilih Semua Dengan pilihan:

2. Lalu klik hapus, nanti akan muncul pop up seperti ini

Basis data

Basis data	Penyortiran	Tindakan
<input type="checkbox"/> edcol	latin1_swedish_ci	=> Check privileges
<input checked="" type="checkbox"/> db_mahasiswa	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> db_malaria	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> db_prospek	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> db_sales	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> db_simpro	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> information_schema	utf8_general_ci	=> Check privileges
<input type="checkbox"/> lostfound	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> mysql	utf8_general_ci	=> Check privileges
<input type="checkbox"/> performance_schema	utf8_general_ci	=> Check privileges
<input type="checkbox"/> phpmyadmin	utf8_general_ci	=> Check privileges
<input type="checkbox"/> simpro	latin1_swedish_ci	=> Check privileges
<input type="checkbox"/> test	latin1_swedish_ci	=> Check privileges

Konfirmasi

Anda akan MENGHANCURKAN sebuah database lengkap! Anda yakin akan menjalankan "DROP DATABASE 'db_mahasiswa';"?

Oke Batal

Klik oke jika kalian sudah yakin untuk menghapus database dan batal jika tidak yakin untuk menghapus databases

- Databases berhasil dihapus dan tampilan seperti gambar dibawah

The screenshot shows the phpMyAdmin interface with the 'Basis data' tab selected. On the left, a tree view lists various databases including 'Baru', 'cdcol', 'db', 'db_malaria', 'db_prospek', 'db_sales', 'db_simpro', 'information_schema', 'lostfound', 'mysql', 'performance_schema', 'phpmyadmin', 'simpro', and 'test'. In the main area, a table lists databases with their character sets and collations. One row for 'latin1_swedish_ci' is highlighted. A button labeled 'Hapus' (Delete) is visible. A yellow message box at the top right of the table area states 'Basis data 1 berhasil dihapus.' (Database 1 deleted successfully).

4.6.3 Cara Membuat Tabel Di Xampp

- Membuat terlebih dahulu field dari table yang akan dibuat seperti dibawah ini

Name	Type	Length	Null	Index
Id	Integer	12	no	Primary key
Nama	Varchar	35	no	-
Fakultas	Varchar	35	no	-
Jurusan	Varchar	35	no	-
status	Enum(aktif, tidak aktif)	-	no	-

- Create atau membuat terlebih dahulu database seperti contoh yang di atas (saya menggunakan database db_mahasiswa yang dibuat seperti di atas)
- Selanjutnya kita akan buat table di dalam databasenya, tinggal klik saja database yang sudah dibuat tadi kemudian pilih Structure, maka tampilan akan kosong dan kita akan disuguhkan dengan nama tabel baru. Buatlah

sesuai kebutuhan database tabel yang dibuat, disini saya dengan nama mahasiswa_tb dengan jumlah kolom 5. dapat dilihat digambar:



4. Baiklah Selanjutnya tinggal klik kirim , maka akan muncul tampilan kolom seperti dibawah ini, dikolom pertama saya isikan Id, dengan type data Int dan jangan lupa centang A.I atau auto_increment untuk memudahkan ketika ada data baru dia akan berurutan nambah dengan otomatis. untuk lebih jelasnya dapat dilihat dibawah ini :

The screenshot shows the MySQL Workbench interface. It displays the 'Struktur' (Structure) tab for the 'mahasiswa_tb' table. The table has five columns: 'id', 'nama', 'alamat', 'jurusan', and 'studi'. The 'id' column is highlighted with a red box. Its properties are: 'Jenis' (Type) set to 'INT', 'PanjangNilai' (Length) set to '12', and 'A_I' (Auto-Increment) checked. Other columns have their properties set to 'Tidak ada' (None). The 'PRIMARY' key is also checked for the 'id' column. At the bottom right of the table structure, there are 'Pratinjau SQL' (Preview SQL) and 'Simpan' (Save) buttons, with the 'Simpan' button highlighted by a red box.

5. Jika sudah langkah diatas sekarang kita tinggal melihat tabel yang ada di database, dapat dilihat dengan cara klik Structure, jika berhasil maka akan tampil seperti berikut ini:

#	Nama	Jenis	Penyortiran	Atribut	Tak Termilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(12)			Tidak	Tidak ada	AUTO_INCREMENT		Ubah Hapus Lainnya
2	nama	varchar(35)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
3	fakultas	varchar(35)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
4	jurusan	varchar(35)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
5	status	enum('Aktif', 'Tidak Aktif', '')		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya

4.6.4 Input Data Pada Tabel di Database

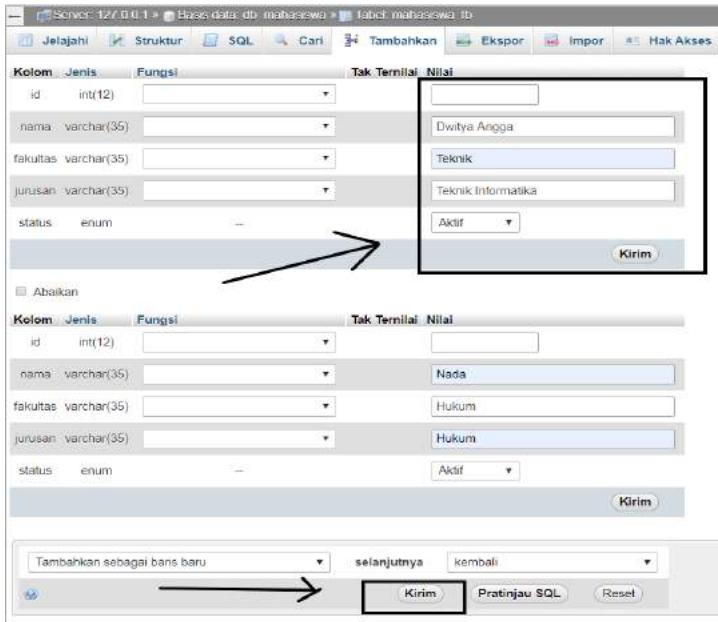
Setelah membuat Database dan table di atas , sekarang menginputkan data di table yang telah dibuat

1. Buka terlebih dahulu tabel mahasiswa_tb yang telah dibuat tadi , lalu pilih menu struktur Tambahkan seperti di bawah:

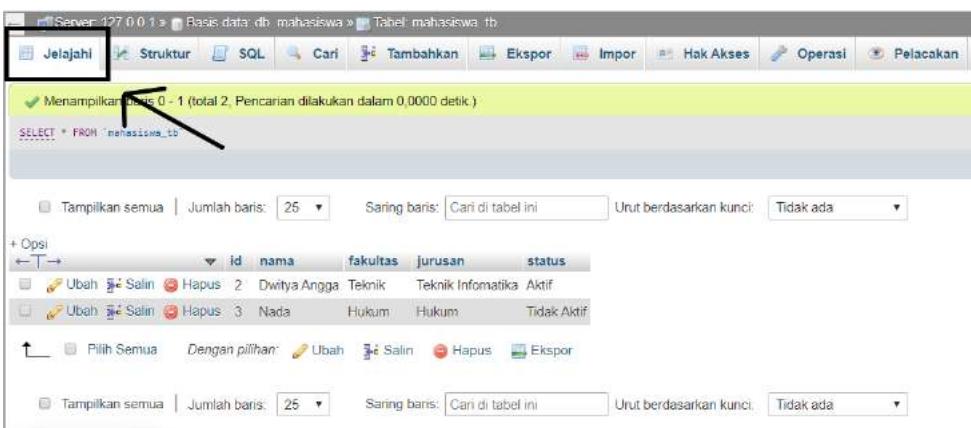
Kolom	Jenis	Fungsi	Tak Termilai	Nilai
id	int(12)			
nama	varchar(35)			
fakultas	varchar(35)			
jurusan	varchar(35)			
status	enum			

Kolom	Jenis	Fungsi	Tak Termilai	Nilai
id	int(12)			
nama	varchar(35)			
fakultas	varchar(35)			
jurusan	varchar(35)			
status	enum			

2. Selanjutnya isi data sesuai dengan field yang dibuat seperti dibawah ini:



- Klik Kirim, selanjutnya buka menu Jelajahi untuk melihat data yang telah diisi seperti gambar ini:



4.6.5 Menghapus data (row) pada Tabel

- Buka Menu Jelajahi yang seperti melihat data table seperti gambar sebelumnya
- Selanjutnya pilih row yang ingin dihapus seperti gambar ini

Jelajahi Struktur SQL Cari Tambahkan Eksport Impor Hak Akses Operasi Pelacakan

Menampilkan baris 0 - 1 (total 2). Pencarian dilakukan dalam 0.0000 detik.

```
SELECT * FROM `mahasiswa_tb`
```

	Tampilkan semua	Jumlah baris:	25	Saring baris:	Cari di tabel ini	Urut berdasarkan kunci:	Tidak ada				
+ Opsi	<input type="button" value="Ubah"/> <input type="button" value="Salin"/> <input type="button" value="Hapus"/>	<input checked="" type="checkbox"/> id	2	nama	Dwitya Angga	fakultas	Teknik	jurusan	Teknik Informatika	status	Aktif
	<input type="button" value="Ubah"/> <input type="button" value="Salin"/> <input type="button" value="Hapus"/>	<input type="checkbox"/> id	3	Nada	Hukum	Hukum	Hukum	Tidak Aktif			
	<input type="button" value="Pilih Semua"/> Dengan pilihan: <input type="button" value="Ubah"/> <input type="button" value="Salin"/> <input type="button" value="Hapus"/> <input type="button" value="Eksport"/>										
	<input type="button" value="Tampilkan semua"/> Jumlah baris: 25 Saring baris: Cari di tabel ini Urut berdasarkan kunci: Tidak ada										

3. Selanjutnya klikhapus , lalu muncul konfirmasi seperti di bawah ini :

Jelajahi Struktur SQL Cari Tambahkan Eksport Impor Hak Akses Operasi Pelacakan Trigger Profil [Edit diconfig]

Menampilkan baris 0 - 2 (total 3). Pencarian dilakukan dalam 0.0000 detik.

```
SELECT * FROM `mahasiswa_tb`
```

	Tampilkan semua	Jumlah baris:	25	Saring baris:	Cari di tabel ini	Urut berdasarkan kunci:	Tidak ada
+ Opsi	<input type="button" value="Ubah"/> <input type="button" value="Salin"/> <input type="button" value="Hapus"/>	<input checked="" type="checkbox"/> id	2	Dwitya Angga	Teknik	Teknik Informatika	Aktif
	<input type="button" value="Ubah"/> <input type="button" value="Salin"/> <input type="button" value="Hapus"/>	<input type="checkbox"/> id	3	Nada	Hukum	Hukum	Tidak Aktif
	<input type="button" value="Pilih Semua"/> Dengan pilihan: <input type="button" value="Ubah"/> <input type="button" value="Salin"/> <input type="button" value="Hapus"/> <input type="button" value="Eksport"/> Konfirmasi						
	<input type="button" value="Tampilkan semua"/> Jumlah baris: 25 Saring baris: Cari di tabel ini Urut berdasarkan kunci: Tidak ada						

Anda yakin akan menjalankan "DELETE FROM `mahasiswa_tb` WHERE `mahasiswa_tb`.`id` = 2"?

Operasi hasil kuery
Markah kuery SQL ini
Judul: Izinkan sementara pengguna untuk mengakses markah ini

Oke Batal

4. Klik oke jika sudah yakin untuk menghapus row (datanya). Hasilnya seperti gambar dibawah ini

Menampilkan baris 0 - 2 (total 3. Pencarian dilakukan dalam 0.0000 detik.)

```
SELECT * FROM `mahasiswa_tb`
```

	Tampilan semua	Jumlah baris:	25	Saring baris:	Cari di tabel ini	Urut berdasarkan kunci:	Tidak ada
+ Opsi	<input type="checkbox"/> Ubah <input checked="" type="checkbox"/> Salin <input type="checkbox"/> Hapus <input type="checkbox"/> 3	<input type="checkbox"/> Pilih Semua	<input type="checkbox"/> Dengan pilihan:	<input type="checkbox"/> Usah <input checked="" type="checkbox"/> Salin <input type="checkbox"/> Hapus <input type="checkbox"/> Eksport			
	Tampilan semua	Jumlah baris:	25	Saring baris:	Cari di tabel ini	Urut berdasarkan kunci:	Tidak ada
+ Operasi hasil kueri	<input type="checkbox"/> Cetak <input checked="" type="checkbox"/> Salin ke clipboard <input type="checkbox"/> Eksport <input type="checkbox"/> Tampilkan bagian <input type="checkbox"/> Buat tampilan						
+ Markahi kueri SQL ini	Judul: <input type="text"/>	<input type="checkbox"/> Izinkan semua pengguna untuk mengakses markah ini					

5. Data Berhasil dihapus 😊

4.6.6 Menghapus Tabel Yang dibuat

1. Buka terlebih dahulu database yang dibuat (disini menggunakan database db_mahasiswa) , lalu pilih menu struktur seperti gambar ini :

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
mahasiswa_tb	<input type="checkbox"/> Jelajahi <input type="checkbox"/> Struktur <input type="checkbox"/> Cari <input type="checkbox"/> Tambahkan <input type="checkbox"/> Kosongkan <input checked="" type="checkbox"/> Hapus	1	InnoDB	latin1_swedish_ci	25 KB	-
1 tabel						
Jumlah						

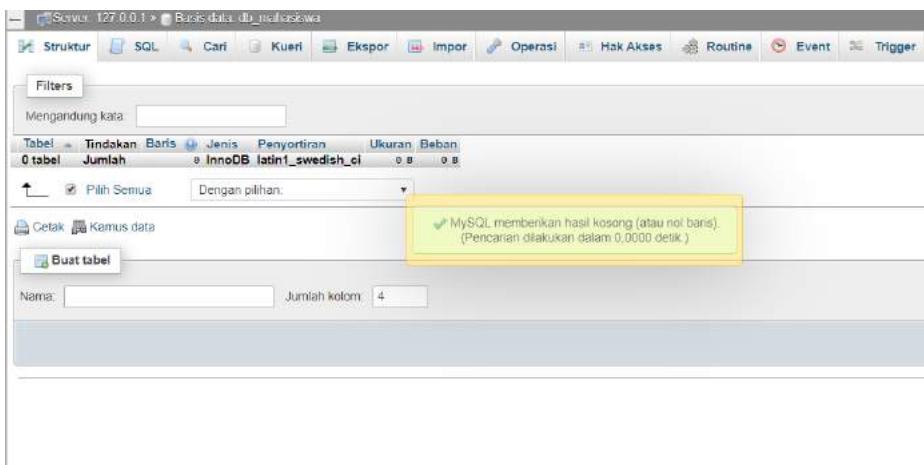
2. Pilih table yang dihapus lalu klik hapus seperti gambar ini :

The screenshot shows the phpMyAdmin interface. In the top navigation bar, there are tabs for Struktur, SQL, Caril, Kueri, Eksport, Impor, Operasi, Hak Akses, Routine, Event, Trigger, and Pelacakan. Below the tabs, there is a 'Filters' section with a search input field labeled 'Mengandung kata:'. A red arrow points to the 'Tindakan' dropdown menu. Another red arrow points to the 'Hapus' (Delete) button in the toolbar above the table list. The main area displays a table with columns: Tabel, Tindakan, Baris, Jenis, Penyortiran, Ukuran, and Beban. One row for 'mahasiswa_tb' is selected, and its details are shown in the right panel: Baris 1, Jenis InnoDB, Penyortiran latin1_swedish_ci, Ukuran 16 KB, and Beban 0 B.

3. Selanjutnya akan muncul konfirmasi dan klik Oke jika seperti ini :

The screenshot shows the phpMyAdmin interface with a confirmation dialog box titled 'Konfirmasi'. The dialog contains the message: 'Anda akan MENGHANCURKAN sebuah tabel lengkap! Anda yakin akan melanjutkan "DROP TABLE" mahasiswa_tb?'. There is a checked checkbox for 'Perbolehan cek foreign key'. At the bottom of the dialog are two buttons: 'Oke' and 'Batal'. The background shows the database structure on the left and the table list on the right, similar to the previous screenshot.

4. Tabel Berhasil dihapus

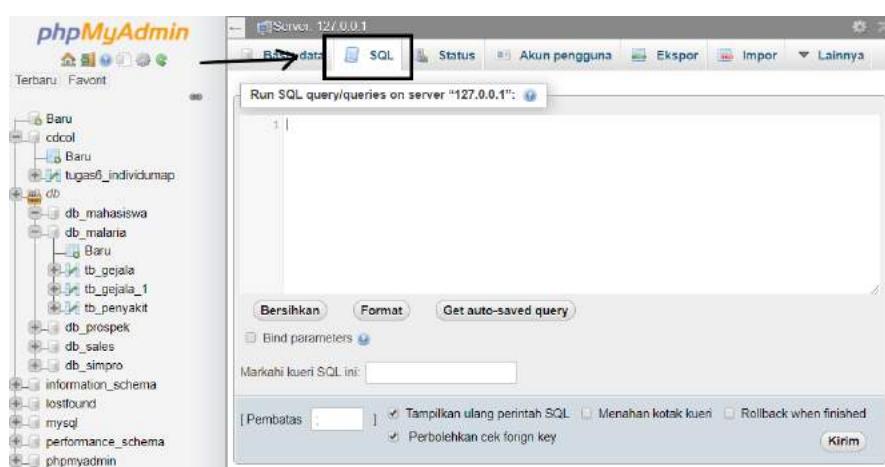


Membuat Tabel Dan Database Dengan Query (Perintah) Di Sql

Ada 2 cara dalam melakukan pembuatan table dan database yaitu secara manual seperti yang diajarkan di halaman sebelumnya dan satunya lagi dibuat dengan perintah atau query yang dituliskan di sqlnya berikut ini penjelasannya:

Membuat Database dengan Query Sql

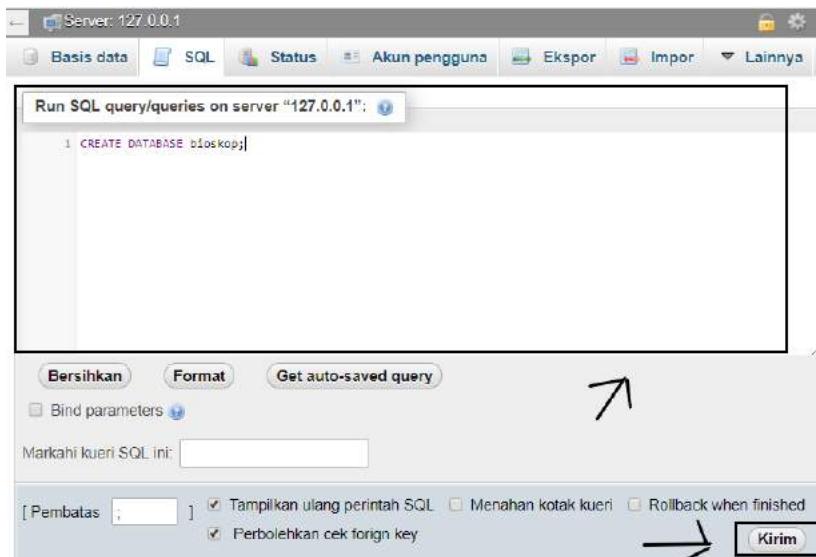
1. Pertama masuk phpmyadmin di XAMPP. Ketik di browser alamat <http://localhost/phpmyadmin> (sama seperti cara yang diatas)
2. Lalu pilih menu Sql yang dimana digunakan sebagai tempat untuk menulis perintahnya



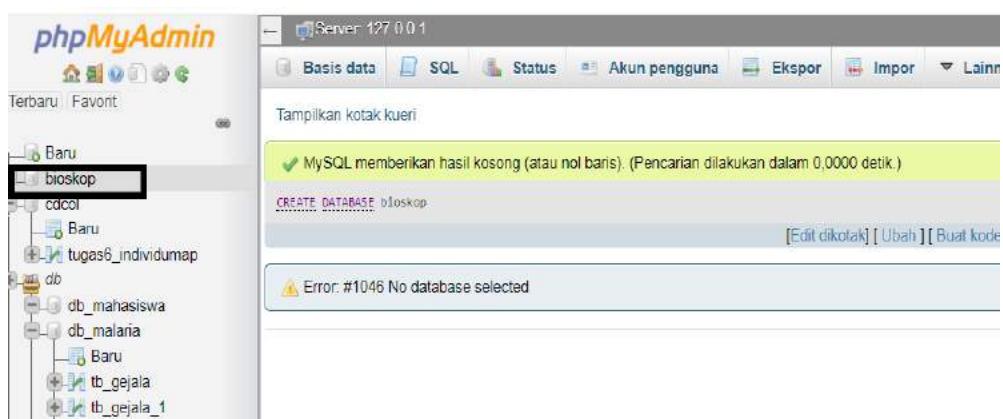
3. Berikut ini adalah perintah untuk membuat database (catatan: database yang dibuat diiberi nama bioskop)

```
CREATE DATABASE bioskop;
```

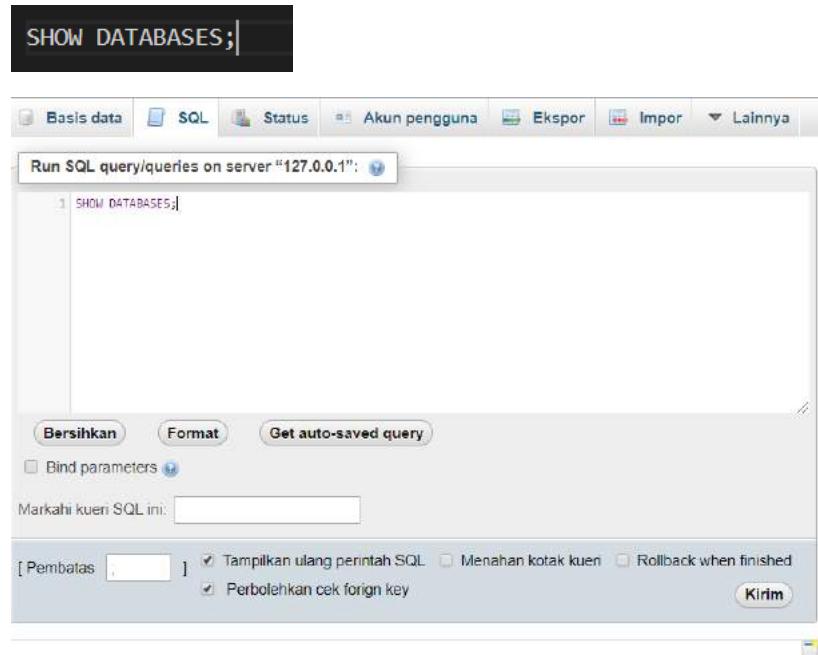
4. Masukan query tersebut di menu sql seperti ini:



5. Lalu klik kirim dan hasilnya terbuatlah database yang diberi nama bioskpo seperti ini



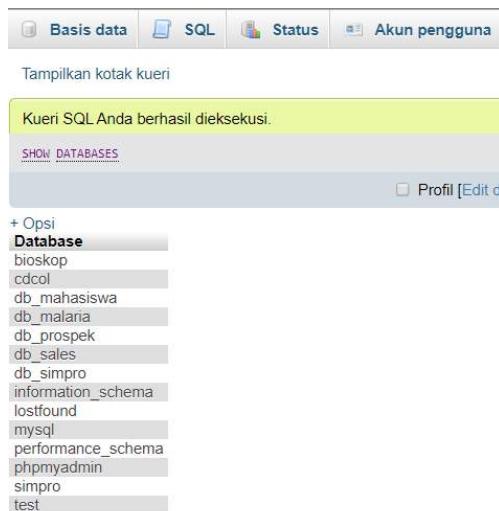
6. Untuk melihat database berhasil dibuat tadi dapat mengetikan di sql dengan perintah



The screenshot shows the phpMyAdmin interface. The top navigation bar has tabs for Basis data, SQL, Status, Akun pengguna, Eksport, Impor, and Lainnya. The SQL tab is active. Below the tabs is a toolbar with icons for Basis data, SQL, Status, Akun pengguna, Eksport, Impor, and Lainnya. A sub-toolbar below the main toolbar includes 'Bersihkan', 'Format', 'Get auto-saved query', and 'Bind parameters'. There is also a field to 'Markahi kueri SQL ini:' and a checkbox for 'Pembatas'. At the bottom of the toolbar are checkboxes for 'Tampilkan ulang perintah SQL', 'Menahan kotak kueri', 'Rollback when finished', and 'Perbolehkan cek foreign key', followed by a 'Kirim' button.

```
SHOW DATABASES;
```

7. Hasilnya seperti berikut:



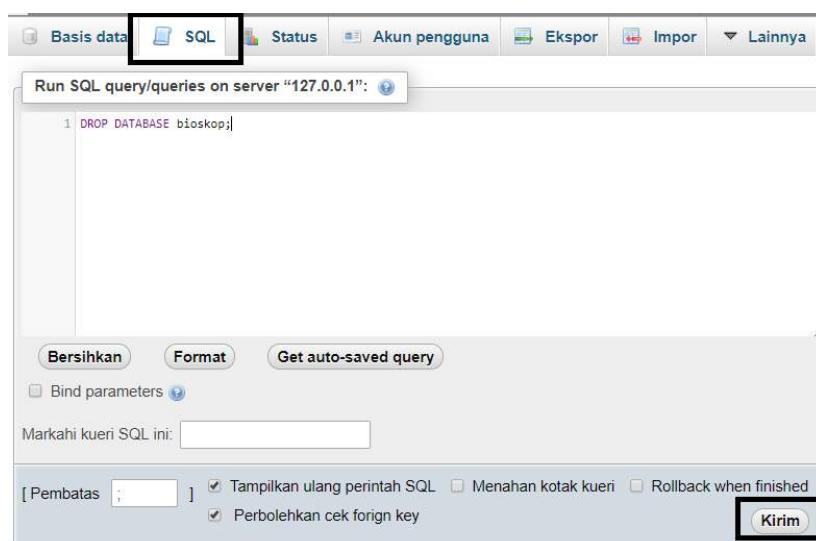
The screenshot shows the phpMyAdmin results page. The top navigation bar has tabs for Basis data, SQL, Status, and Akun pengguna. The SQL tab is active. The results area shows a green header bar with the message 'Kueri SQL Anda berhasil dieksekusi.' Below it is a table with a single row containing the text 'SHOW DATABASES'. To the right of the table is a 'Profil [Edit di]' link. On the left side, there is a sidebar with a '+ Opsi' section and a 'Database' section listing various databases: bioskop, cdccl, db_mahasiswa, db_malana, db_prospek, db_sales, db_simpro, information_schema, lostfound, mysql, performance_schema, phpmyadmin, simpro, and test.

Menghapus Database dengan perintah di sql

1. Sama seperti diatas pilih menu sql terlebih dahulu. Berikut ini perintah untuk menghapus Database yang dibuat

```
DROP DATABASE bioskop;
```

2. Selanjutnya query tersebut dituliskan di sql seperti ini



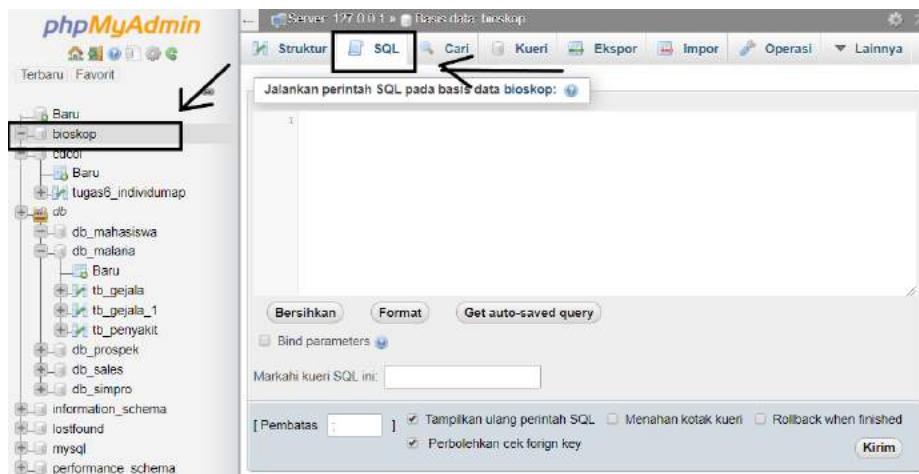
3. Lalu klik kirim. Database dengan nama bioskop berhasil dihapus

Basis data	Penyortiran	Tindakan	Basis data	Penyortiran	Tindakan
bioskop	latin1_swedish_ci	<input type="checkbox"/> Check privileges	cdcol	latin1_swedish_ci	<input type="checkbox"/> Check privileges
cdccl	latin1_swedish_ci	<input type="checkbox"/> Check privileges	db_mahasiswa	latin1_swedish_ci	<input type="checkbox"/> Check privileges
db_mahasiswa	latin1_swedish_ci	<input type="checkbox"/> Check privileges	db_malaria	latin1_swedish_ci	<input type="checkbox"/> Check privileges
db_malaria	latin1_swedish_ci	<input type="checkbox"/> Check privileges	db_prospek	latin1_swedish_ci	<input type="checkbox"/> Check privileges
db_prospek	latin1_swedish_ci	<input type="checkbox"/> Check privileges	db_sales	latin1_swedish_ci	<input type="checkbox"/> Check privileges
db_sales	latin1_swedish_ci	<input type="checkbox"/> Check privileges	db_simpro	latin1_swedish_ci	<input type="checkbox"/> Check privileges
db_simpro	latin1_swedish_ci	<input type="checkbox"/> Check privileges	information_schema	utf8_general_ci	<input type="checkbox"/> Check privileges
information_schema	utf8_general_ci	<input type="checkbox"/> Check privileges	lostfound	latin1_swedish_ci	<input type="checkbox"/> Check privileges
lostfound	latin1_swedish_ci	<input type="checkbox"/> Check privileges	mysql	latin1_swedish_ci	<input type="checkbox"/> Check privileges
mysql	latin1_swedish_ci	<input type="checkbox"/> Check privileges	performance_schema	utf8_general_ci	<input type="checkbox"/> Check privileges
performance_schema	utf8_general_ci	<input type="checkbox"/> Check privileges	phpmyadmin	utf8_bin	<input type="checkbox"/> Check privileges
phpmyadmin	utf8_bin	<input type="checkbox"/> Check privileges	simpro	latin1_swedish_ci	<input type="checkbox"/> Check privileges
simpro	latin1_swedish_ci	<input type="checkbox"/> Check privileges	test	latin1_swedish_ci	<input type="checkbox"/> Check privileges
test	latin1_swedish_ci	<input type="checkbox"/> Check privileges			

Gambar sebelum dan sesudah penghapusan database

Membuat Tabel Dengan Perintah Di Sql

1. Pilih database yang akan dibuat table (di sini saya menggunakan database bioskop yang telah dibuat). Selanjutnya pilih Menu sql seperti pembuatan database



2. Berikut ini perintah untuk membuat table

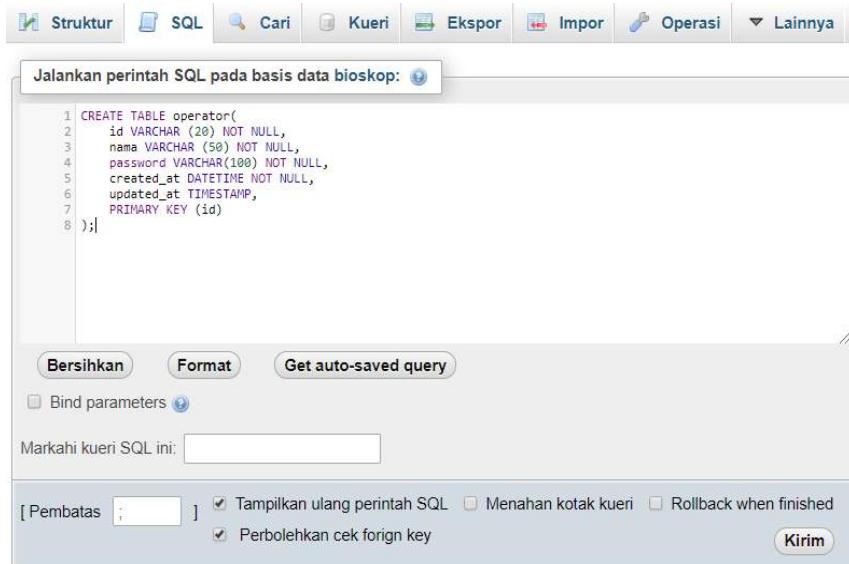
```
CREATE TABLE operator(
    id VARCHAR (20) NOT NULL,
    nama VARCHAR (50) NOT NULL,
    password VARCHAR(100) NOT NULL,
    created_at DATETIME NOT NULL,
    updated_at TIMESTAMP,
    PRIMARY KEY (id)
);
```

Penjelasan:

Pada kode diatas kamu dapat membuat sebuah tabel dengan perintah **CREATE TABLE**. Kemudian ada tipe data berupa **INT**, **VARCHAR**, **TEXT**, **DATETIME** dan **TIMESTAMP**. Untuk tipe data **VARCHAR** kamu harus menentukan berapa panjang maksimal dari kolom tersebut. Tipe **INT** dapat kamu tentukan panjang angka yang akan digunakan. Sedangkan **TIMESTAMP** akan selalu diisi secara otomatis oleh MySQL saat baris baru dibuat. Kemudian ada juga atribut tambahan **NOT NULL** dimana kolom tersebut tidak boleh kosong saat proses insert. Kemudian ada penentuan

PRIMARY KEY dimana kolom tersebut akan menjadi pembeda antar kolom agar mencegah data dengan id sama memiliki dua baris yang sama.

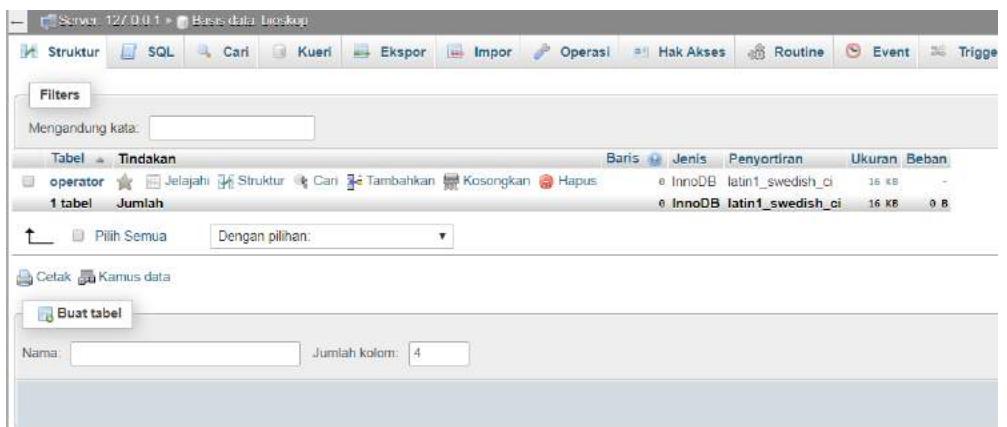
3. Selanjutnya Menuliskan perintah tersebut di sql seperti ini



```
1 CREATE TABLE operator(
2     id VARCHAR (20) NOT NULL,
3     nama VARCHAR (50) NOT NULL,
4     password VARCHAR(100) NOT NULL,
5     created_at DATETIME NOT NULL,
6     updated_at TIMESTAMP,
7     PRIMARY KEY (id)
8 );
```

Bersihkan Format Get auto-saved query
Bind parameters Markahi kueri SQL ini:
[Pembatas ;] Tampilkan ulang perintah SQL Menahan kotak kueri Rollback when finished
 Perbolehkan cek forign key Kirim

4. Lalu klik kirim. Tabel operator pada database bioskop telah dibuat



Mengandung kata:

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
operator	★ ⌂ Jalajahi ⌂ Struktur ⌂ Cari ⌂ Tambahkan ⌂ Kosongkan ⌂ Hapus	6	InnoDB	latin1_swedish_ci	16 KB	-
1 tabel	Jumlah	6	InnoDB	latin1_swedish_ci	16 KB	0 B

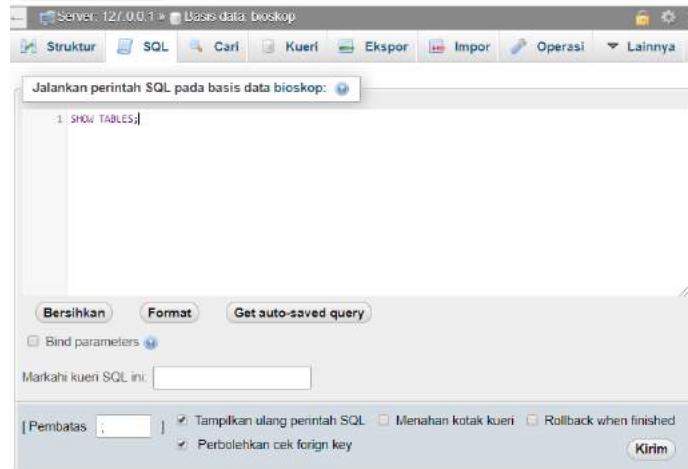
Cetak Kamus data

Buat tabel

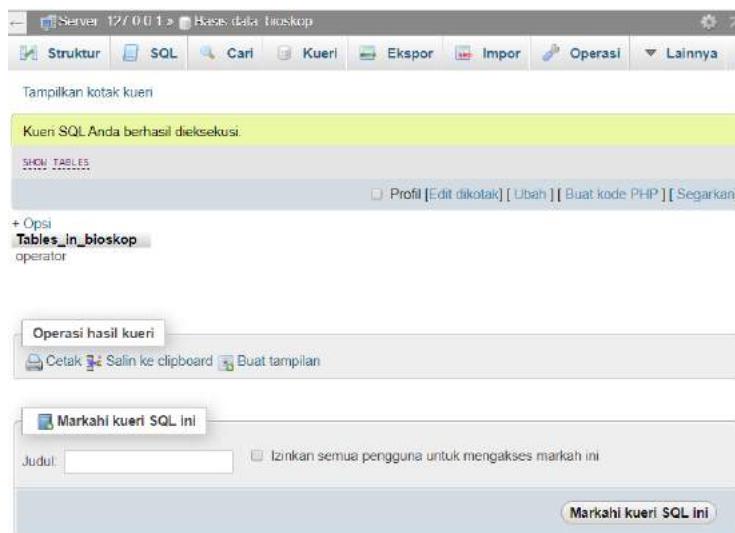
Nama: Jumlah kolom:

5. Untuk melihat table yang tadi dibuat bisa menggunakan perintah seperti ini

```
SHOW TABLES;
```



6. Hasilnya seperti ini :



Menginputkan data di Table

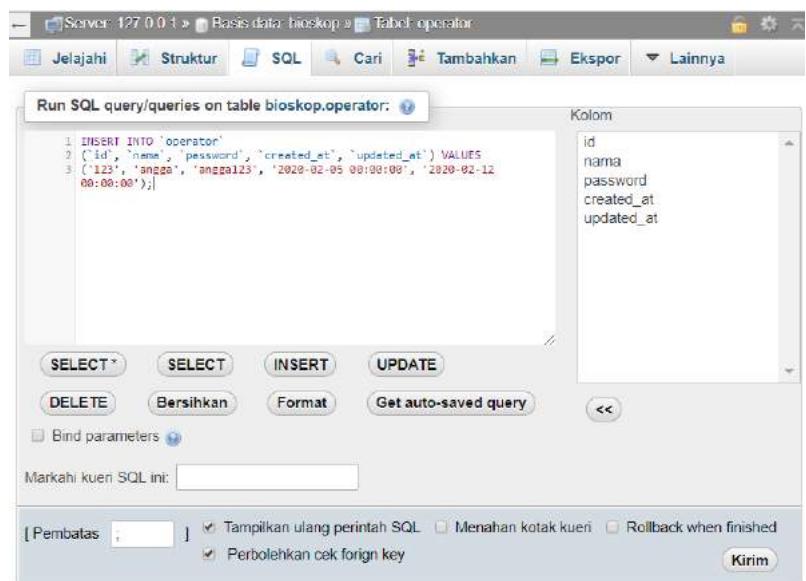
1. Pilih table yang di buat (table operator)
2. Berikut ini query untuk menginputkan data di table

```

INSERT INTO `operator`
(`id`, `nama`, `password`, `created_at`, `updated_at`) VALUES
(['123', 'angga', 'angga123', '2020-02-05 00:00:00', '2020-02-12 00:00:00']);

```

3. Selanjutnya Ketika query atau kodingan tersebut di sql



4. Data berhasil diinputkan , hasilnya seperti ini

The screenshot shows the MySQL Workbench interface after running the query. The results are displayed in a green bar at the top:

✓ Menampilkan baris 0 - 0 (total 1, Pencarian dilakukan dalam 0,0000 detik.)

The SQL query shown is:

```

SELECT * FROM `operator`

```

Below the results, there is a table with the following data:

	+ Opsi	↓	id	nama	password	created_at	updated_at
	<input type="button" value="Ubah"/> <input type="button" value="Salin"/> <input type="button" value="Hapus"/>		123	angga	angga123	2020-02-05 00:00:00	2020-02-12 00:00:00

At the bottom of the interface, there are two sets of buttons and input fields:

- Tampilan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini
- Tampilan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini

Menghapus Data Di Table

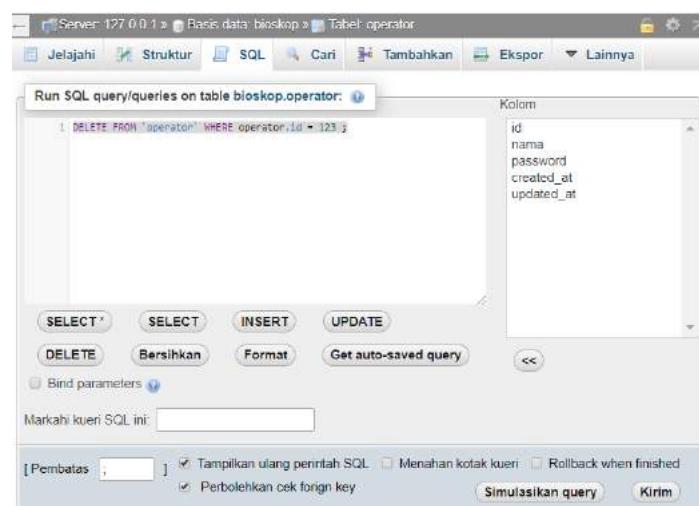
1. Pilih Table seperti yang dilakukan di halaman sebelum
2. Berikut ini perintah menghapus data di table

```
DELETE FROM `operator` WHERE operator.id = 123 ;
```

Penjelasan:

Untuk menghapus digunakan perintah delete. Yang dimana Kodingan ini berarti menghapus data di table operator dengan id = 123

3. Selanjutnya menulis query di sql



4. Data berhasil dihapus



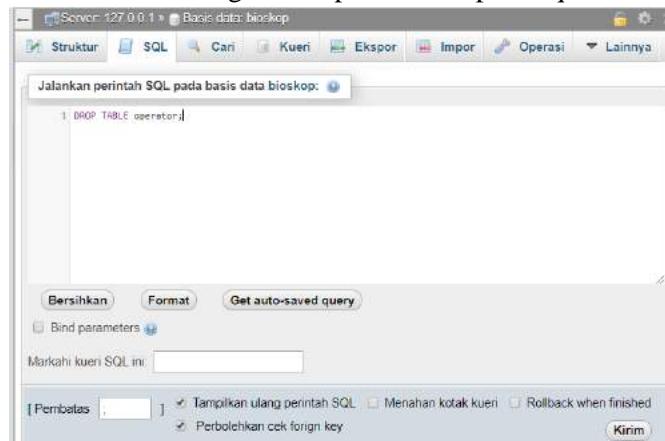
Menghapus Tabel di Sql

1. Menghapus tabel memiliki perintah sama seperti menghapus Database di atas yaitu:

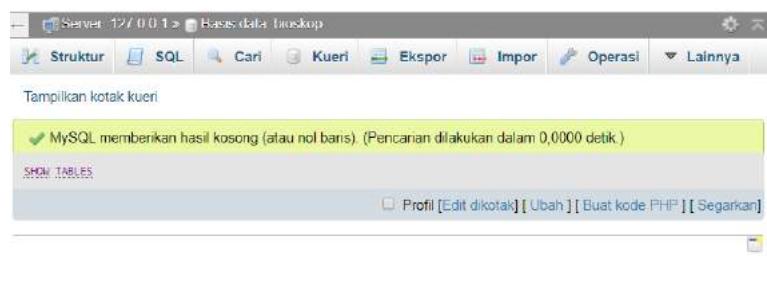
```
DROP TABLE operator;
```

Penjelasan : Perintah Drop merupakan perintah untuk menghapus, yang dimana kodingan ini berarti menghapus table operator pada database tempat table itu dibuat

2. Menuliskan Kodingan atau perintah itu pada sql



3. Hasilnya Tabel berhasil dihapus



BAB V

PENGENALAN FRAMEWORK CODEIGNITER

Bab ini membahas mengenai segala macam mengenai codeigniter

5.1 Pengenalan Framework



Sebelum memahami codeigniter, ada baiknya memahami terlebih dulu framework itu sendiri. Framework adalah sebuah struktur konseptual dasar yang digunakan untuk memecahkan suatu permasalahan, bahkan isu-isu kompleks yang ada. Sebuah framework telah berisi sekumpulan arsitektur/konsep-konsep yang dapat mempermudah dalam pemecahan sebuah permasalahan. Perlu diingat, framework bukanlah peralatan/tools untuk memecahkan sebuah masalah, tetapi sebagai alat bantu. Framework hanya menjadi sebuah konstruksi dasar yang menopang sebuah konsep atau sistem yang bersifat “essential support” atau dengan kata lain “penting tapi bukan komponen utama”.

5.2 Alasan Menggunakan Framework?

Salah satu alasan mengapa orang menggunakan framework untuk membangun website/aplikasi adalah kemudahan yang ditawarkan. Pada framework biasanya sudah tersedia struktur aplikasi yang baik, yaitu: standar coding, best practice, design pattern, dan common function. Dengan menggunakan framework kita dapat langsung fokus kepada business process yang dihadapi tanpa harus berfikir banyak masalah struktur aplikasi, standar coding dan lain-lain.

Dengan memanfaatkan design pattern dan common function yang telah ada di dalam framework maka hal tersebut dapat mempercepat proses pengembangan aplikasi. Kita tidak perlu membuat sesuatu fungsionalitas yang bersifat umum.

Tanpa disadari ketika kita membangun sebuah aplikasi yang banyak melibatkan fungsionalitas yang telah dibangun itu ternyata sama/berulang-ulang. Nah dengan pengelompokan itulah kita dapat mempercepat proses pengembangan aplikasi.

Selain kemudahan dan kecepatan dalam membangun sistem, dengan menggunakan framework kita juga dapat “menyeragamkan” cara kita mengimplementasikan kode program. Dengan framework kita akan “dipaksa” untuk patuh kepada sebuah kesepakatan. Yang pasti dengan framework itu akan memudahkan pengembang lain untuk mempelajari dan mengubah aplikasi yang telah dibuat, dengan catatan kode yang hasilkan konsisten dan patuh pada sebuah aturan tertentu.

5.3 Manfaat Menggunakan Framework

- Dapat membantu developer/programmer dalam membangun aplikasi
- Penerapan Design Patterns memudahkan dalam rancangan Pengembangan dan pemeliharaan sistem.
- Stability dan Reliability aplikasi yang kita bangun lebih stabil dan handal.
- Coding Style Consistence, memudahkan dalam hal membaca source code karena cara pengkodean yang sama/konsisten.
- Security Concern Framework, mengantisipasi dan memasang perisai terhadap adanya berbagai masalah.
- Dokumentasi, framework dapat mendisiplinkan kita menulis dokumentasi untuk sistem apa yang kita bangun.

5.4 Kekurangan Menggunakan Framework

- Programmer mungkin akan menemukan kemungkinan batasan-batasan ketika merancang aplikasi menggunakan framework, contoh : belum tentu fungsi di Framework A ada di Framework B.
- Kemungkinan akan ada penambahan biaya apabila pihak development kurang mendekumentasi dan kurang disupport

5.5 Perbedaan CMS dan Framework



Framework bisa diartikan sebagai kerangka kerja, dalam dunia pemrograman php Framework adalah sekumpulan script yang terstruktur yang dapat membantu seorang develop atau pengembang dalam menangani masalah masalah pemrograman php dengan mudah sehingga mempercepat dalam pembuatan website. Atau bisa dikatakan sebagai sekumpulan fungsi-fungsi php yang terstruktur sehingga programer tidak harus membuat fungsi-fungsi itu dari awal. Framework beda dengan CMS (Content Management System), apa sih CMS itu ? Apa perbedaan CMS dan Framework. Sebagai orang awam atau bisa dikatakan newbie dalam php akan tanya-tanya. CMS sendiri biasanya digunakan oleh orang develop atau newbie yang tidak mau repot masalah perprograman php. Mereka lebih suka menggunakan CMS karena semua sistemnya sudah terbangun dan tinggal merubah beberapa bagian atau menghilangkan bagian-bagian tertentu untuk bisa menjadi website. Banyak sekali CMS yang ada sekarang ini misalnya *Lokomedia, Wordpress, Joomla, Mambo, Drupal* dan masih banyak lagi.

Nah sekarang apa sih CMS itu?? CMS atau dikatakan Sistem Manajemen Konten adalah sebuah perangkat website yang sudah jadi yang memungkinkan seseorang untuk menambahkan dan juga mengubah atau memanipulasi isi dari website cms tersebut. Apa Perbedaannya dengan Framework ? Framework sendiri adalah kerangka kerja website yang terstruktur sedangkan CMS adalah sebuah sistem website yang sudah jadi dan seorang develop tinggal menambahkan atau mengurangi isi konten tersebut, setidaknya untuk membuat website yang diinginkan hanya merubah sedikit, sedangkan pada framework anda harus membuat dulu backend dan juga frontendnya. Lebih gampangnya dalam mencermati ini

semisal anda adalah seorang pengrajin patung, istilah Framework itu adalah anda membuat patung dari nol, yaitu dengan memahat kayu sampai jadi patung. Kalo CMS adalah anda tinggal membeli patung yang sudah jadi , anda tinggal merakitnya dengan memasang bagian- bagiannya dan memberi warna maka sudah manjadi patung.

5.6 Pengertian Codeigniter



CodeIgniter adalah sebuah web application network yang bersifat open source yang digunakan untuk membangun aplikasi php dinamis. CodeIgniter menjadi sebuah framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi web. Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodennya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih CodeIgniter sebagai framework pilihannya. Karena kelebihan-kelebihan yang dimiliki oleh CodeIgniter, pembuat PHP Rasmus Lerdorf memuji CodeIgniter di frOSCon (Agustus 2008) dengan mengatakan bahwa dia menyukai CodeIgniter karena “it is faster, lighter and the least like a framework.”

CodeIgniter pertama kali dikembangkan pada tahun 2006 oleh Rick Ellis. Dengan logo api yang menyala, CodeIgniter dengan cepat “membakar” semangat para web developer untuk mengembangkan web dinamis dengan cepat dan mudah menggunakan framework PHP yang satu ini.

Catatan Penting!

Permulaan dalam belajar *Framework CodeIgniter* ini anda diharusnya mengerti terlebih dahulu konsep php sederhana karena bila anda tidak mengenali terlebih dahulu konsep PHP tersebut, anda akan banyak mengalami kesulitan dalam melakukan pemrograman dengan menggunakan konsep MVC yang ada pada framework CI tersebut. Bukan hanya masalah konsep yang ada pada framework CodeIgniter anda akan kesulitan dan akan terus berikut seputar PHP. Framework CodeIgniter ini sudah menggunakan konsep PHP dan **OOP** (Object Oriented Programming) dan akan lebih bagus lagi bila anda mengenal PHP terlebih dahulu kemudian mengenal konsep **OOP**.

5.7 Pengenalan OOP (*Object Oriented Programming*)

Object Oriented Programming (OOP) merupakan paradigma pemrograman yang berorientasi pada obyek. Semua paradigma ini akan dibungkus dalam beberapa kelas – kelas atau object object yang terlihat terstruktur yang nantinya setiap obyek dapat menerima pesan, memproses pesan dan mengirim pesan ke obyek lain. OOP ini digunakan untuk mengatasi keterbatasan pada bahasa pemrograman tradisional. Dalam konsep OOP, semua masalah dibagi dalam obyek obyek karena konsep data dan fungsi – fungsi yang akan mengoperasikan digabungkan menjadi satu kesatuan yang dapat disebut sebagai obyek.

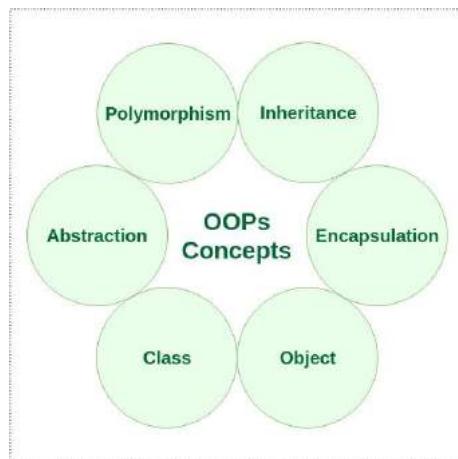
5.8 Pengenalan Object

Secara sederhana obyek sendiri adalah kumpulan variable dan fungsi yang dibungkus menjadi satu tempat atau kesatuan. Dalam sebuah objek diciptakan memalui sebuah kelas atau dikenal dengan istilah instan of class . Didalam obyek sendiri mempunyai 2 elemen utama :

- **Attributes** : merupakan nilai – nilai yang tersimpan dalam obyek tersebut dan secara langsung maupun tidak langsung untuk menentukan karakteristik dari obyek tersebut.
- **Method** : ini merupakan suatu aksi yang akan dijalankan atau dikerjakan obyek tersebut.

Untuk mengenal namanya Attribute / Property dan Method maka kita harus mengenal juga namanya Class dan Inheritance.

5.9 Mengenal Struktur OOP



Pada tutorial kali ini akan dibahas dasar-dasar OOP di dalam bahasa PHP. Versi PHP yang sudah mendukung OOP adalah PHP versi 5 ke atas, sedangkan versi 4 dan sebelumnya belum sepenuhnya mendukung OOP. Tutorial ini akan membahas dasar-dasar OOP sambil langsung mempraktekkan bagaimana OOP tersebut diimplementasikan di PHP.

Encapsulation

Dalam OOP, semua elemen program merupakan objek-objek yang saling berinteraksi satu sama lain. Dalam interaksinya, tiap objek mempunyai informasi yang bisa diakses oleh objek lain ataupun yang disembunyikan dari objek lain. Sebagai contoh, dalam dunia nyata, objek mobil memiliki steer, pedal, yang langsung berinteraksi dengan sopir. Tetapi sopir tidak secara langsung berinteraksi dengan mesin. Sama halnya seperti objek-objek dalam OOP. Tiap objek memiliki informasi yang harus disembunyikan maupun yang tersedia untuk public (objek lainnya). Dalam OOP ini disebut enkapsulasi (encapsulation) yang merupakan salah satu konsep OOP yang sangat penting.

Encapsulation Ringkasan

- Istilah enkapsulasi sebenarnya adalah kombinasi data dan fungsionalitas dalam sebuah unit tunggal sebagai bentuk untuk menyembunyikan detail informasi.
- Proses enkapsulasi memudahkan kita untuk menggunakan sebuah objek dari suatu kelas karena kita tidak perlu mengetahui segala hal secara rinci.
- Enkapsulasi menekankan pada antarmuka suatu kelas, atau dengan kata lain bagaimana menggunakan objek kelas tertentu.
- Contoh: kelas mobil menyediakan antarmuka fungsi untuk menjalankan mobil tersebut, tanpa kita perlu tahu komposisi bahan bakar, udara dan kalor yang diperlukan untuk proses tersebut.

Abstraction (Abstrak Kelas)

- Kelas merupakan deskripsi abstrak informasi dan tingkah laku dari sekumpulan data.
- Kelas dapat diilustrasikan sebagai suatu cetak biru(blueprint) atau prototipe yang digunakan untuk menciptakan objek.
- Kelas merupakan tipe data bagi objek yang mengenkapsulasi data dan operasi pada data dalam suatu unit tunggal.
- Kelas mendefinisikan suatu struktur yang terdiri atas data kelas (data field), prosedur atau fungsi (method), dan sifat kelas (property).

Class

Class adalah merupakan “blueprint” dari objek. Sedangkan objek adalah instance dari class. Class adalah class sendiri adalah struktur dari suatu objek yang didalamnya terdapat Attribute dan juga Method, atau bisa disebut sebuah struktur yang mendefinisikan sebuah variable dan juga method umum pada semua object. class juga merupakan grup suatu object dengan kemiripan attributes/properties, behaviour dan relasi ke object lain.

Sesuai dengan keterangan diatas dapat diartikan bahwa class itu bahan dasar sebelum kita membuat sesuatu, atau juga merupakan sebuah wadah dari apa yang kita buat.

Misalkan kita ingin membuat tempe, tentu saja harus ada bahan dasarnya yaitu kedelai, atau ketika kita mempunyai mangga, apel, dan durian maka ketiga benda tadi masuk dalam kumpulan yang bernama buah. Dari kedua contoh tersebut kita bisa menentukan mana “class” dan mana “object”, pada Tempe merupakan object yang dibuat dari class Kedelai. Dan mangga, apel, durian merupakan kumpulan object dari class Buah. Sebagai contoh di dunia nyata, class seperti cetakan kue, dan objek adalah kuenya sendiri. Dari satu class bisa dibuat berapa pun objek yang diperlukan. Sebuah class dalam php diawali dengan keyword class diikuti dengan nama class. Pada contoh di bawah adalah deklarasi sebuah

```
// class dalam php dengan nama "artikel".
// deklarasi class dalam php
class artikel {
    // di sini isi code dari class akan ditulis
}
```

Sebuah class biasanya memiliki ciri-ciri tertentu, seperti dalam dunia nyata mobil miliki warna, ukuran, jumlah ban, dll. Ciri-ciri ini disebut dengan attribute/property. Selain attribute, class juga memiliki operasi-operasi yang bisa dilakukan oleh class tersebut. Seperti pada mobil memiliki operasi maju, mundur, pindah gigi, dll. Operasi-operasi ini disebut dengan method. Dalam program OOP pun sama halnya seperti di dunia nyata. Tiap class memiliki attribute dan method. Attribute dan method nya pun bisa satu ataupun sebanyak yang diperlukan. Contoh: class artikel, memiliki attribute: judul, nama_penulis, tanggal_terbit, kategori, dll. Artikel juga memiliki method terbitkan(), arsipkan(), hapus(), edit(), dll. Berikut ini contoh class “artikel” dalam php dengan method dan attribute nya

```
✓ class artikel {
    // attribute
    public $tanggal_terbit;
    // method
    function terbitkan() {
        echo 'artikel berhasil diterbitkan';
    }
}
```

Keterangan: Seperti kita lihat pada contoh di atas, attribute adalah variable biasa dan diawali dengan keyword public. Public merupakan access modifier (access modifier memerlukan pembahasan lanjut, yang berkaitan dengan encapsulation). Sedangkan method merupakan fungsi biasa yang memiliki argumen ataupun tidak. Method juga bisa memiliki access modifier sama halnya seperti attribute.

Object

Objek adalah instance dari class, ini berarti class harus di-instantiate (dibuat objeknya) terlebih dahulu agar bisa digunakan dalam program (kecuali static class). Untuk membuat objek baru dalam PHP, digunakan keyword new. Berikut adalah contoh pembuatan objek dalam php:

```
// deklarasi class
class artikel {
    public $tanggal_terbit;
    function terbitkan() {
        echo 'artikel berhasil diterbitkan';
    }
}
// instantiate objek baru
$artikel_a = new artikel();
// instantiate objek lainnya
$artikel_b = new artikel();
```

Keterangan: Pada contoh di atas, \$artikel_a merupakan objek dari class artikel. Objek ini otomatis telah memiliki seluruh atribut dan method yang dimiliki oleh class nya, yaitu: attribute ‘tanggal_terbit’ dan method ‘terbitkan()’. Sama halnya juga dengan \$artikel_b memiliki atribut dan method yang sama. Untuk mengakses suatu atribut/method dari objek, maka kita bisa menggunakan tanda -> (strip, lebih besar dari) Contohnya:

```
// mengakses attribute  
$artikel_a->tanggal_terbit  
// mengakses method  
$artikel_a->terbitkan()
```

Inheritance

Sebuah class dapat di-inherit (diwariskan) ke class turunannya, sehingga class turunannya memiliki attribute dan method yang sama dengan class induknya. Sebagai contoh, class artikel bisa diwariskan menjadi class berita. Class artikel disebut parentclass, dan class berita disebut child class. Dalam inheritance, keyword yang digunakan adalah extends. Berikut contoh inheritance dalam PHP:

```
// parent class  
class artikel {  
    public $tanggal_terbit;  
    function terbitkan() {  
        echo 'artikel berhasil diterbitkan ';  
    }  
}  
// child class  
class berita extends artikel {  
    // attribute tambahan  
    public $nama_wartawan;  
    public $nara_sumber;  
    // method tambahan  
    function cek_nara_sumber() {  
        echo 'nara sumber berhasil dicek ';  
    }  
}
```

Keterangan: Pada contoh di atas, class berita yang merupakan child class akan otomatis memiliki atribut ‘\$tanggal_terbit’ dan method ‘terbitkan()’ sebagai warisan dari parent classnya, yaitu class artikel. Sebagai tambahan, class berita memiliki atribut ‘\$nama_wartawan’, ‘\$nara_sumber’ dan method ‘cek_nara_sumber()’ yang didefinisikan tersendiri pada child class tersebut.

Inheritance Ringkasan

- Kita dapat mendefinisikan suatu kelas baru dengan mewarisi sifat dari kelas lain yang sudah ada.
- Penurunan sifat ini bisa dilakukan secara bertingkattingkat, sehingga semakin ke bawah kelas tersebut menjadi semakin spesifik.
- Sub kelas memungkinkan kita untuk melakukan spesifikasi detail dan perilaku khusus dari kelas supernya.

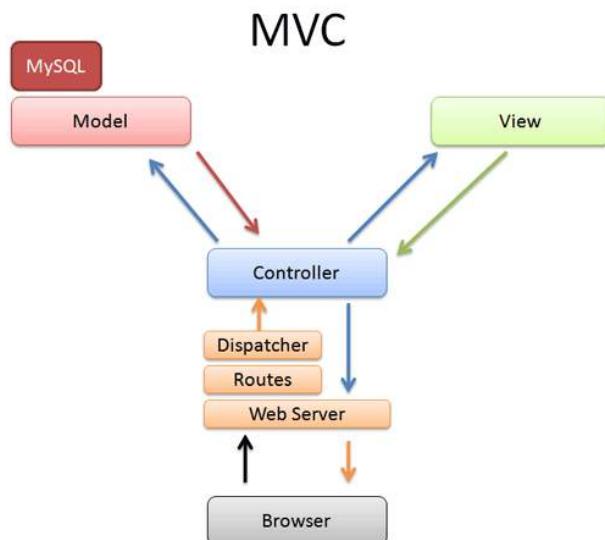
Polymorphism

- Polimorfisme merupakan kemampuan objekobjek yang berbeda kelas namun terkait dalam pewarisan untuk merespon secara berbeda terhadap suatu pesan yang sama.
- Polimorfisme juga dapat dikatakan kemampuan sebuah objek untuk memutuskan method mana yang akan diterapkan padanya, tergantung letak objek tersebut pada jenjang pewarisan.
- Method overriding.
- Method name overloading

5.10 Apa Itu MVC?

MVC adalah konsep dasar yang harus diketahui sebelum mengenal CodeIgniter. MVC (Model, View, Controller) merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman Small Talk, yang memisahkan bisnis logic (alur piker), data logic (penyimpanan data) dan presentation logic (antarmuka aplikasi) atau secara

sederhana adalah memisahkan antara desain, data dan proses. Ada 3 komponen yang membangun suatu MVC yaitu :



1. Model

Biasanya berhubungan dengan data dan interaksi ke database atau webservice. Model juga merepresentasikan struktur data dari aplikasi yang bisa berupa basis data maupun data lain. Biasanya didalam model akan berisi class dan fungsi untuk mengambil, melakukan update dan menghapus data website. Sebuah aplikasi web menggunakan basis data dalam menyimpan data, maka pada bagian Model biasanya akan berhubungan dengan perintah-perintah query SQL.

2. View

Merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data hasil dari model dan controller kepada user. View tidak memiliki akses langsung terhadap bagian model.

3. Controller

merupakan bagian yang mengatur hubungan antara bagian model dan bagian view. Pada controller terdapat class-clas dan fungsi-fungsi yang memproses

permintaan dari View ke dalam struktur data di dalam model. Controller juga tidak boleh berisi kode untuk mengakses basis data Karena tugas megakses data telah diserahkan kepada model. Tugas controller adalah menyediakan berbagai variable yang akan ditampilkan di view, memanggil model untuk melakukan akses ke basis data, menyediakan penanganan kesalahan/error, mengerjakan proses logika dari aplikasi serta melakukan validasi atau cek terhadap input.

5.11 Contoh File Model, Controller, View

1. Model (login.php)

```
<?php

class M_login extends CI_Model
{
    // function __construct()
    //{
    //    parent::__construct();
    //}
    function cek_akun($username,$password)
    {
        $query = $this->db->get_where('tbl_login',array('username'=>$username, 'password'=>$password));

        foreach ($query->result_array() as $qy )
        {
            $this->session->set_userdata('username', $qy['username']);
            $this->session->set_userdata('level', $qy['level']);
        }

        $row = $query->row_array();
        if ($query->num_rows()>0)
        {
            date_default_timezone_set('Asia/Jakarta');

            $date   =  date("Y/m/d H:i:s");
            $data   =  array('tgl'=>$date, 'status'=>1);

            $this->db->where('username', $username);
            $this->db->update('tbl_login',$data);
            return $row['level'];
        } else {
            return 0;
        }
    }

    function logout($username)
    {
        $this->db->set('status', 0);
    }
}
```

2. Controller (sdi.php)

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class sdi extends CI_Controller
5 {
6     public function __construct()
7     {
8         parent::__construct();
9         $this->load->model('m_sdi'); // di jadikan constructor agar langsung di jalankan saat di akses dan bisa di akses
10        $this->load->helper('url');
11    }
12
13
14
15    public function index()
16    {
17        $data['tbl_sdi'] = $this->m_sdi->tampil_data()->result();
18        $dataku['tbl_sdi'] = $this->m_sdi->get();
19        $this->load->view('templates/header');
20        $this->load->view('templates/sidebar');
21        $this->load->view('sdi',$data);
22        $this->load->view('templates/footer');
23    }
24
25    public function tambah_aksi()
26    {
27        $pekerjaan      = $this->input->post('pekerjaan');
28        $sto           = $this->input->post('sto');
29        $lokasi         = $this->input->post('lokasi');
30        $desigmator     = $this->input->post('desigmator');
31        $uraian_pekerjaan = $this->input->post('uraian_pekerjaan');
32        $satuan         = $this->input->post('satuan');
33
34        $data = array(
35            'pekerjaan'   => $pekerjaan,
36            'sto'          => $sto,
37            'lokasi'       => $lokasi,
38            'desigmator'  => $desigmator,

```

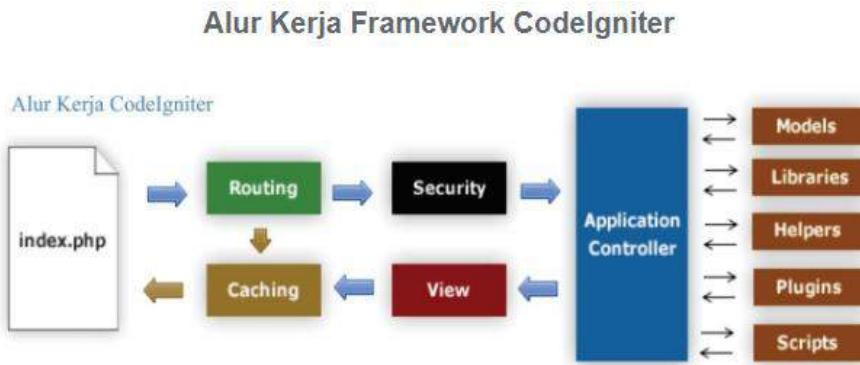
3. View (home.php)

```

1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5
6     <meta charset="utf-8">
7     <meta http-equiv="X-UA-Compatible" content="IE=edge">
8     <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
9     <meta name="description" content="">
10    <meta name="author" content="">
11
12    <title>SIMPRO</title>
13    <!-- Custom fonts for this template-->
14    <link href=<?php echo base_url('assets/vendor/fontawesome-free/css/all.min.css') ?> rel="stylesheet" type="text/css">
15    <!-- Page level plugin CSS-->
16    <link href=<?php echo base_url('assets/vendor/datatables/dataTables.bootstrap4.css') ?> rel="stylesheet">
17    <!-- Custom styles for this template-->
18    <link href=<?php echo base_url('assets/css/sb-admin.css') ?> rel="stylesheet">
19 </head>
20
21 <body id="page-top">
22 <nav class="navbar navbar-expand navbar-dark bg-dark static-top">
23     <a class="navbar-brand mr-1" <?php echo anchor('home','Sales dan Teknisi'); ?>
24         <button class="btn btn-link btn-sm text-white order-1 order-sm-0" id="sidebarToggle" href="#">>
25             <i class="fas fa-bars"></i>
26         </button>
27
28     <!-- Navbar Search -->
29     <form class="d-none d-md-inline-block form-inline ml-auto mr-0 mr-md-3 my-2 my-md-0">
30         <div class="input-group">
31             <input type="text" class="form-control" placeholder="Search for..." aria-label="Search" aria-describedby="basic-addon2">
32             <div class="input-group-append">
33                 <button class="btn btn-primary" type="button">
34                     <i class="fas fa-search"></i>
35                 </button>
36             </div>
37         </div>
38     </form>

```

5.12 Alur Kerja Framework CodeIgniter



1. Index.php: Index.php disini berfungsi sebagai file pertama dalam program yang akan dibaca oleh program.
2. The Router: Router akan memeriksa HTTP request untuk menentukan hal apa yang harus dilakukan oleh program.
3. Cache File: Apabila dalam program sudah terdapat “cache file” maka file tersebut akan langsung dikirim ke browser. File cache inilah yang dapat membuat sebuah website dapat di buka dengan lebih cepat. Cache file dapat melewati proses yang sebenarnya harus dilakukan oleh program codeigniter.
4. Security: Sebelum file controller di load keseluruhan, HTTP request dan data yang disubmit oleh user akan disaring terlebih dahulu melalui fasilitas security yang dimiliki oleh codeigniter.
5. Controller: Controller akan membuka file model, core libraries, helper dan semua resources yang dibutuhkan dalam program tersebut.
6. View: Hal yang terakhir akan dilakukan adalah membaca semua program yang ada dalam view file dan mengirimkannya ke browser supaya dapat dilihat. Apabila file view sudah ada yang di “cache” maka file view baru yang belum ter-cache akan mengupdate file view yang sudah ada.

5.13 Fungsi dari Codeigniter

1. Mempercepat dan mempermudah kita dalam pembuatan website.
2. Menghasilkan struktur pemrograman yang sangat rapi, baik dari segi kode maupun struktur file phpnnya

3. Memberikan standar coding sehingga memudahkan kita atau orang lain untuk mempelajari kembali sistem aplikasi yang dibangun.

5.14 Kelebihan Codigniter

1. Berukuran sangat kecil. File downloadnya hanya sekitar 2MB, itu pun sudah included dokumentasinya yang sangat lengkap.
2. Dokumentasi yang bagus. Saat anda mendownloadnya, telah disertakan dengan dokumentasi yang berisi pengantar, tutorial, bagaimana panduan penggunaan, serta referensi dokumentasi untuk komponen-komponennya.
3. Kompatibilitas dengan Hosting. CodeIgniter mampu berjalan dengan baik pada hampir semua platform hosting. CodeIgniter juga mendukung database-database paling umum, termasuk MySQL.
4. Tidak ada aturan coding yang ketat. Terserah anda jika anda hanya ingin menggunakan Controller, tanpa View, atau tidak menggunakan Model, atau tidak salah satu keduanya. Namun dengan menggunakan ketiga komponennya adalah pilihan lebih bijak.
5. Kinerja yang baik. CodeIgniter sangat cepat bahkan mungkin bisa dibilang merupakan framework yang paling cepat yang ada saat ini.
6. Sangat mudah diintegrasikan. CodeIgniter sangat mengerti tentang pengembangan berbagai library saat ini. Karenanya CodeIgniter memberikan kemudahan untuk diintegrasikan dengan library-library yang tersedia saat ini.
7. Sedikit Konfigurasi. Konfigurasi CodeIgniter terletak di folder application/config. CodeIgniter tidak membutuhkan konfigurasi yang rumit, bahkan untuk mencoba menjalankannya, tanpa melakukan konfigurasi sedikitpun ia sudah bisa berjalan.
8. Mudah dipelajari. Disamping dokumentasi yang lengkap, ia juga memiliki berbagai forum diskusi,

5.15 Kekurangan Codeigniter

1. CodeIgniter tidak ditujukan untuk pembuatan web dengan skala besar.
2. Library yang sangat terbatas. Hal ini dikarenakan sangat sulit mencari plugin tambahan yang terverifikasi secara resmi, karena pada situsnya CodeIgniter tidak menyediakan plugin-plugin tambahan untuk mendukung pengembangan aplikasi dengan CI.

3. Belum adanya editor khusus CodeIgniter, sehingga dalam melakukan create project dan modul-modulnya harus berpindah-pindah folder

5.16 Codeigniter 3

Pada bulan Juli 2013, EllisLab mengumumkan bahwa mereka mencari pemilik baru untuk CodeIgniter karena internal mereka sendiri tidak memiliki cukup fokus untuk terus mengembangkan CodeIgniter. Akhirnya pada bulan Oktober 2014, kepemilikan CodeIgniter berpindah tangan kepada British Columbia Institute of Technology, salah satu sekolah tinggi teknologi di Kanada. Setelah hampir lima bulan lamanya sejak peralihan kepemilikan, BCIT akhirnya merilis CodeIgniter 3.0. Dan berikut adalah perubahan codeigniter 2 menjadi codeigniter 3 :

1. Codeigniter 3 memerlukan PHP versi 5.1.6 atau di atasnya
2. Penamaan model, controller Codeigniter 3 harus di awali huruf besar
3. Driver databasenya kini memiliki refactoring yang lebih luas. Sekarang default database nya menggunakan mysqli, tidak lagi menggunakan mysql
4. Penambahan user agent Windows 7, Windows 8, Windows 8.1, Android, Blackberry, iOS dan PlayStation 3
5. Update perbaikan di mimes.php
6. Update penulisan class dengan PHP 5 style
7. Pindah path halaman error di application/view/errors
8. Pindah Log Class di application/core
9. Update perbaikan di beberapa Library dan Helper
10. Perbaikan file dokumentas

5.17 Perbandingan Dari Versi Framework Codeigniter



Perbedaan CodeIgniter Versi 2 dan Versi 3

Perbedaan Codeigniter v 2.x dengan v 3.x terbilang cukup banyak. Namun yang paling terasa ialah dibagian penamaan model dan controller. Pada versi 2.x tidak diberikan ketentuan penulisannya, namun di versi 3, penulisan ditentukan berupa penggunaan huruf kapital pada bagian controller dan model. Untuk perbedaan lainnya telah admin list pada penjelasan berikut ini :

1. Perbedaan Penulisan Nama File pada Controller dan Model

Seperti yang diulas diawal, penulisan awal nama file di bagian controller dan model pada Codeigniter 3.x diharuskan menggunakan huruf kapital.

2. Versi Minimum PHP

Tak seperti Codeigniter v 2.0 yang menerapkan spesifikasi minimum PHP versi 5.1.6, Codeigniter versi 3.x mengharuskan programmer untuk menerapkan setidaknya PHP v 5.3.7.

3. Perubahan Database Driver

Pada CI versi lama, EllisLab menggunakan MySQL sebagai database driver, namun di versi terbaru yang dirilis oleh MIT, penggunaan database driver sudah diarahkan ke MySQLi yang di klaim lebih lengkap dan efektif.

4. Penambahan User Agent

Versi Codeigner 2.x pertama dirilis di tahun 2006, dimana Operating System saat itu belum sebanyak jaman now ini. Pada versi 3.x, User Agent pada Codeigniter telah ditambahkan. Diantara beberapa user agent yang ditambahkan ialah sebagai berikut :

- iOS,
- Blackberry OS,
- Windows 7,
- Windows 8, serta
- Windows 8.1.

Perbedaan CI V 3.x dengan Codeigniter Versi 4.x

Setelah mengetahui perbedaan CI versi 2 dengan versi 3, kini admin akan mencoba menulis perbedaan versi tersebut dengan versi terbaru, yaitu versi 4.x. Pada saat tulisan ini diterbitkan, versi 4 telah dipublish di github, namun belum disiapkan untuk diproduksi. Sejak awal bulan Juni 2016 hingga saat ini Maret 2018 (hampir 2 tahun) pihak CodeIgniter sedang menjalankan masa percobaan. Terlepas dari lamanya masa launching Codeigniter ini, semoga cepat diresmikan deh. Back to the point lalu apa saja perbedaan versi 3 dan 4 ? Berikut ulasannya

1. Beberapa perubahan dibagian coding (query database, model),
2. Pada v 4.x ditambahkan beberapa library, serta
3. Pemanggilan model dari controller.

5.18 Pembandingan Framework Codeigniter dengan Framework lainnya



Pembanding	Codeigniter	Laravel	Yii	CakePHP
<i>Trend and popularity</i>	CodeIgniter selalu populer. Ini memberikan kemudahan penggunaan dalam 2.x dan karenanya sebagian besar pengembang web lebih suka CodeIgniter. Ada banyak situs web di CodeIgniter dan akan tetap ada.	Laravel sedang dalam puncak popularitasnya saat ini. Dengan gaya yang mendukung pengkodean ekspresif yang disukai oleh pengembang berpengalaman, Laravel berubah dari kekuatan ke kekuatan. Pekerjaan Laravel adalah yang paling banyak muncul di kalangan freelance juga	Perjalanan Yii berjalan lambat dan berat. Popularitas ada di sana beberapa tahun yang lalu dan ada situs web dan aplikasi yang dikembangkan di Yii dari era itu yang membutuhkan pembaruan dan pemeliharaan. Namun, Yii bukanlah pilihan pertama hari ini.	Sebagaimana dibahas sedikit di atas, penggunaan CakePHP telah turun secara perlahan sejak 2009. Pekerjaan baru di CakePHP semakin sedikit karena kerangka kerja lain telah mengambil alih mantel
<i>Structure and Updates</i>	Strukturnya adalah MVC (model-view-controller) dan menyediakan on-boarding yang mudah. Struktur ini secara longgar didasarkan pada pemrograman berorientasi objek tetapi kebanyakan pengembang	Laravel mengikuti struktur pengarsipan MVC. Laravel dilengkapi dengan alat baris perintah yang disebut Artisan. Yang membuat semua pengembang ngiler melihat fitur-fiturnya. Dengan Artisan Anda dapat membuat model dan kontrol, mengatur tugas yang dijadwalkan, antrian,	Yii mengikuti pendekatan MVC yang ketat dan konvensi penamaan yang ketat untuk file. Pembaruan di Yii tidak begitu sering dibandingkan dengan kerangka kerja lainnya. Sudah	Kode CakePHP mengikuti pendekatan metodis yang sangat ketat dan sebenarnya dimodelkan pada Ruby on Rails. Nama file dan database sangat ketat. Kerangka ini

	<p>menggunakannya sesuai dengan kebutuhan mereka.</p> <p>Kerangka kerja yang digunakan adalah case-insensitive.</p> <p>Namun, sejak 3.x ia telah menjadi jauh lebih ketat dalam pendekatannya untuk menghadapi pesaing baru seperti Laravel dan Phalcon.</p> <p>Namun, tidak ada dukungan untuk <i>composer</i> dan <i>Middleware</i></p>	<p>dan perintah khusus. Alat lain Eloquent membantu kita berinteraksi dengan basis data, membuat metode ajaib yang bermanfaat, dan mengambil data.</p> <p>Pembaruan Laravel umum dan berulang</p>	<p>lama diperkenalkan dan masih dalam versi 2.0.x</p>	<p>mengikuti konvensi dalam penamaan dan merupakan sesuatu yang bisa dibanggakan.</p> <p>Laravel Framework mengambil pendekatan ini dengan scaffolding berbasis Artisan.</p> <p>Namun, pengembangannya terus berlanjut dan memiliki pengikut yang sangat aktif mengikuti dan merilis timeline di Github</p>
<i>Libraries and online help</i>	<p>CodeIgniter menyediakan banyak fungsi bawaan dan situs web mereka memiliki panduan praktis untuk digunakan.</p> <p>Namun, ada</p>	<p>Bantuan terbaik laravel adalah dokumentasi resminya sendiri yang sangat rumit dan bermanfaat.</p> <p>Namun, Anda bisa mendapatkan bantuan hebat dari Laracast.com.</p> <p>Jumlah Libraries</p>	<p>Yii adalah kerangka kerja lama dan karenanya memiliki banyak tutorial dan bantuan online yang tersedia untuk pengembang web. Ini</p>	<p>CakePHP memiliki sejarah panjang dan menemukan bantuan dan tutorial CakePHP di internet sangat mudah</p>

	<p>konsensus umum bahwa CodeIgniter tidak menyediakan Libraries pihak ketiga dengan tingkat yang sama seperti yang disediakan kerangka kerja lain seperti Laravel atau Yii. Mereka sekarang telah memperkenalkan caching yang kuat dalam 3x</p>	<p>Laravel standar yang dapat Anda komposer ke dalam proyek Laravel sangat mengejutkan. Anda dapat menemukannya di 100-an Libraries di Packagist. Demikian pula, Anda dapat menemukan bantuan hebat di Stackoverflow.</p>	<p>memiliki dokumentasi yang sangat bagus yang menyediakan sebagian besar jawaban. Ada cukup paket di Packagist.org untuk membuat klaim yang kuat</p>	<p>karena ada banyak sumber daya. CakePHP hadir dengan Komposer dan seharusnya cukup mudah untuk menambahkan Libraries vendor</p>
<i>Databases integration</i>	<p>CodeIgniter menyediakan basis data relasional seperti MySQL atau PostgreSQL secara default. Namun, CodeIgniter 1.6.3arave dengan alat ORM sendiri.</p>	<p>Laravel dilengkapi dengan alat ORM sendiri. Laravel dilengkapi dengan migrasi untuk port perubahan database dan versi yang mudah. Demikian pula, Laravel dilengkapi dengan alat Seeder untuk memperkenalkan data uji. Laravel juga dilengkapi dengan kunci Redis: value store. Istirahat, Laravel memiliki paket untuk setiap database yang tersedia</p>	<p>Yii menyediakan dukungan database yang layak, Anda dapat bekerja dengan berbagai basis data yang mencakup SQLite, PostgreSQL, Oracle, MySQL dan DBMS relasional lainnya. Ini telah dibangun dalam dukungan ActiveRecord ORM. Namun,</p>	<p>Akses basis data sendiri sangat mudah di CakePHP apa dengan semua konvensi penamaan tersebut. Kerangka dilengkapi dengan ORM dan alat pembuat kueri. CakePHP hadir dengan dukungan untuk database relasional</p>

			kami dapat menemukan paket pihak ketiga untuk MongoDB, Microsoft BI.	konvensi termasuk Oracle. Namun, untuk database NoSQL Anda harus mendapatkan pustaka pihak ketiga
<i>API support</i>	Tidak ada dukungan API bawaan di 2.0. CI menyediakan <code>base_controller</code> baru bernama <code>Rest_Controller</code> . Masih belum ada konsep middleware di CodeIgniter maka aplikasi tidak akan didasarkan pada praktik pemrograman terbaru pada framework kerja lain yang dilakukan	Laravel memiliki dukungan terbaik untuk API di luar blok. Laravel menyediakan kerangka kerja kecil yang disebut Lumen serta dukungan untuk REST built-in dengan rute API. Anda dapat menggunakan middleware untuk otentikasi. Laravel juga menyediakan layanan Paspor bawaan untuk fungsionalitas oAuth 2	Seperti Laravel, Yii telah bekerja dengan baik untuk menyesuaikan API ke dalam Framework. Yii dilengkapi dengan rute dan pengaturan API untuk mempercepat dengan API. o Input data 2 dan JSON didukung untuk dukungan yang baik. Meskipun dikembangkan dengan anggun, modul API Yii masih terlihat ditambahkan	API ditangani dengan baik di CakePHP karena struktur kodennya dan sebagian besar hal diimplementasikan dengan cara yang benar. Namun, menerapkan API dan terus terang perutean Kue itu sendiri telah menjadi sulit dan orang-orang cenderung menjauh dari CakePHP karena alasan ini.

			daripada built-in	
<i>Ease of development and templating</i>	CodeIgniter mudah digunakan dan karenanya merupakan kurva belajar yang paling tidak mengkhawatirkan. Seiring dengan tidak adanya mesin template, ada kode PHP murni yang dapat digunakan	Laravel hadir dengan mesin templat Blade yang bagus untuk pengembang frontend yang tidak harus belajar PHP. Laravel membuat pengembangan lebih mudah dengan menyediakan fitur-fitur seperti perutean, migrasi basis data, pembuat fasih dan query, utilitas baris perintah, pagination built-in dll. Fitur lain yang sangat berguna dari laravel adalah mesin template. Dengan menggunakan Blade, kita dapat mewarisi templat serta mendefinisikan bagian.	pengembangannya jauh lebih cepat di Yii, karena fitur pembuatan kode otomatis, yang dapat dicapai dengan menggunakan ekstensi bernama gii. Gii mendukung antarmuka perintah dan gui. Dengan gii kita dapat membuat apa saja dari model sederhana hingga menyelesaikan CRUD. Untuk informasi lebih lanjut tentang gii, silakan periksa dokumentasinya. Yii tidak menggunakan sistem templating apa pun secara default, tetapi memiliki kemampuan untuk	Jika Anda mengikuti pola desain CakePHP maka Anda harus cukup diatur untuk siklus pengembangan yang lebih cepat mengingat kepatuhannya pada praktik standar. Waktu harus dihemat. CakePHP mengikuti struktur PHP mentah dalam templat tetapi memungkinkan templat Twig yang seharusnya bagus untuk pengembang UI

			menggunakan sistem templating pihak ketiga seperti Twig dan Smarty.,	
--	--	--	--	--

5.19 URL Framework Codeigniter

Secara default, URL di CodeIgniter dirancang untuk search-engine dan ramah manusia. Alih-alih menggunakan standar "string kueri" pendekatan ke URL yang identik dengan sistem dinamis, CodeIgniter menggunakan pendekatan segment-based:

example.com/news/article/my_article

Catatan: URL String Query dapat secara optional diaktifkan, seperti yang dijelaskan di bawah ini.

URI Segmen

Segmen dalam URL, dalam mengikuti dengan pendekatan Model-View-Controller, biasanya mewakili:

example.com/class/function/ID

1. Segmen pertama mewakili kelas controller yang harus dipanggil.
2. Segmen kedua merupakan fungsi kelas, atau metode, yang harus dipanggil.
3. Yang ketiga, dan setiap segmen tambahan, mewakili ID dan variabel yang akan diteruskan ke controller.

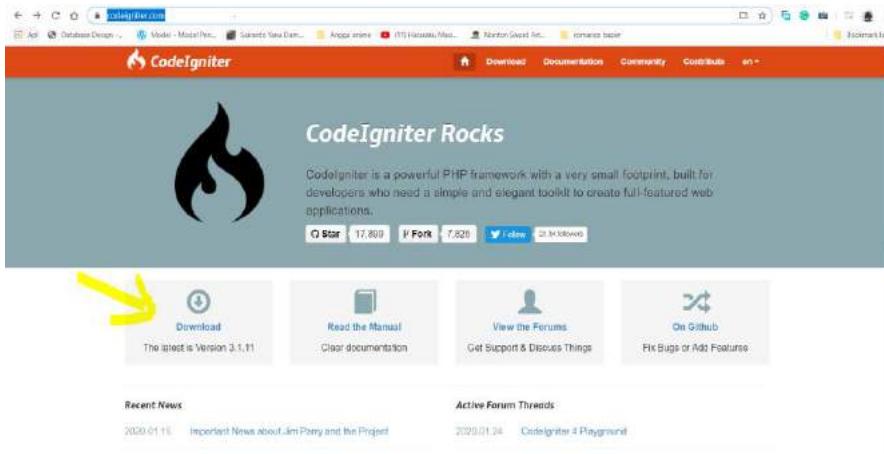
URI Class dan URL Helper mengandung fungsi yang membuatnya mudah untuk bekerja dengan data URI Anda. Selain itu, URL Anda dapat dipetakan dengan menggunakan fitur URI Routing untuk fleksibilitas yang lebih.

5.20 Cara instalasi Codeigniter

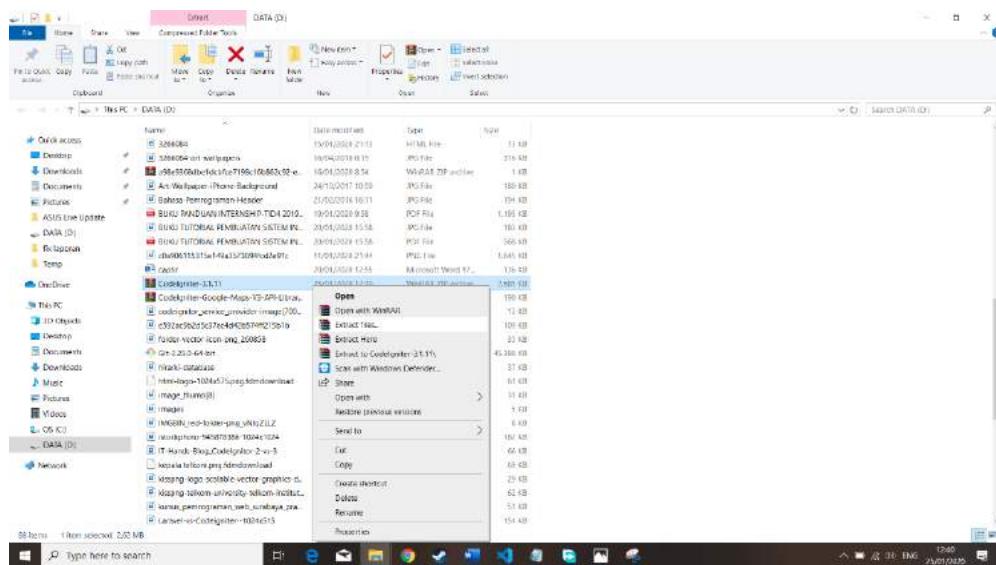
Yang harus dipersiapkan dalam memasang Framework Codeigniter di computer atau laptop anda:

- Web Server (apache,nginx dsb)
- Database (MySQL, postgresql,Maria DB)
- PHP

1. Buka terlebih dahulu link <https://codeigniter.com/>, selanjutnya download codeigniter 3.1.11, Dimana versi yang paling stabil yaitu versi tersebut



2. Setelah mendownload file codeigniter tersebut, extract file tersebut



3. Setelah di extract file hasilnya seperti ini

	Name	Date modified	Type	Size
Quick access				
Desktop	application	19/09/2019 19:08	File folder	
Downloads	system	19/09/2019 19:08	File folder	
Documents	user_guide	19/09/2019 19:08	File folder	
Pictures	editorconfig	19/09/2019 19:08	EDITORCONFIG File	1 KB
ASUS Live Update	.gitignore	19/09/2019 19:08	Text Document	1 KB
DATA (D)	composer.json	19/09/2019 19:08	JSON File	1 KB
fix laporan	contributing.md	19/09/2019 19:08	MD File	7 KB
Temp	index.php	19/09/2019 19:08	PHP File	11 KB
	license	19/09/2019 19:08	Text Document	2 KB
	readme.txt	19/09/2019 19:08	RST File	3 KB
OneDrive				

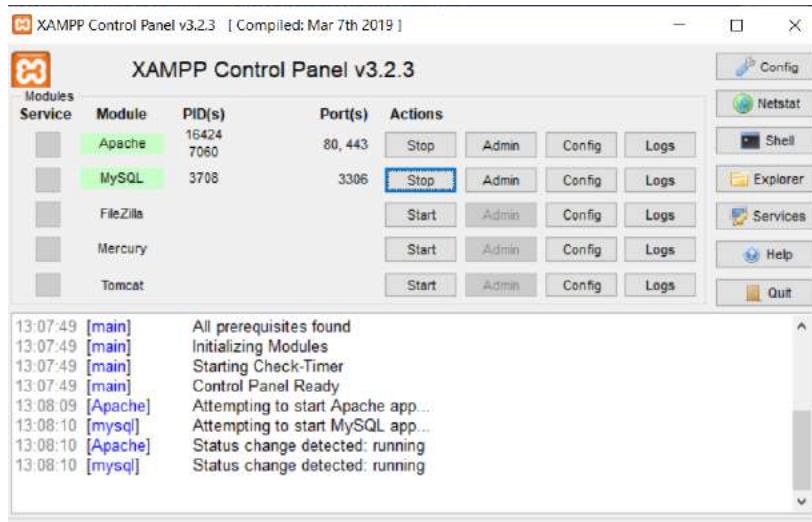
4. Pindahkan file yang sudah di extract ke folder c/xampp/htdocs (dengan catatan sudah menginstall xampp)

	Name	Date modified	Type	Size
OneDrive	application	17/12/2018 22:03	File folder	
This PC	Bot	15/12/2019 23:05	File folder	
3D Objects	CodeIgniter-3.1.11	25/01/2020 13:00	File folder	
Desktop	dashboard	14/05/2019 17:28	File folder	
Documents	images	29/12/2018 10:34	File folder	
Downloads	img	14/05/2019 17:28	File folder	
Music	lostfound	14/05/2019 17:36	File folder	
Pictures	maps	16/01/2019 22:49	File folder	
Videos	resources	17/12/2018 0:53	File folder	
	simon	16/01/2020 22:07	File folder	

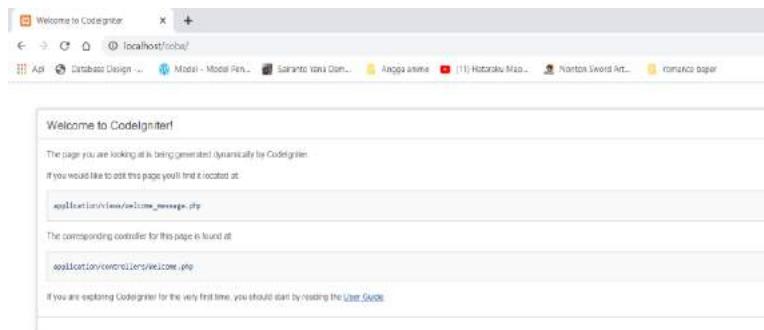
5. Setelah itu ubah nama codeigner-3.1.11 sesuai dengan nama project atau keinginan anda , ini dilakukan agar memudahkan dalam memanggilnya pada saat di browser (disini saya memberi nama dengan “ coba ”)

	Name	Date modified	Type	Size
Quick access	application	17/12/2018 22:03	File folder	
Desktop	Bot	15/12/2019 23:05	File folder	
Downloads	coba	25/01/2020 13:00	File folder	
Documents	dashboard	14/05/2019 17:28	File folder	
Pictures	images	29/12/2018 10:34	File folder	
ASUS Live Update	img	14/05/2019 17:28	File folder	
DATA (D)	lostfound	14/05/2019 17:36	File folder	
fix laporan	maps	16/01/2019 22:49	File folder	
Temp	resources	17/12/2018 0:53	File folder	

6. Selanjutnya buka xampp lalu start Apache dan Mysql



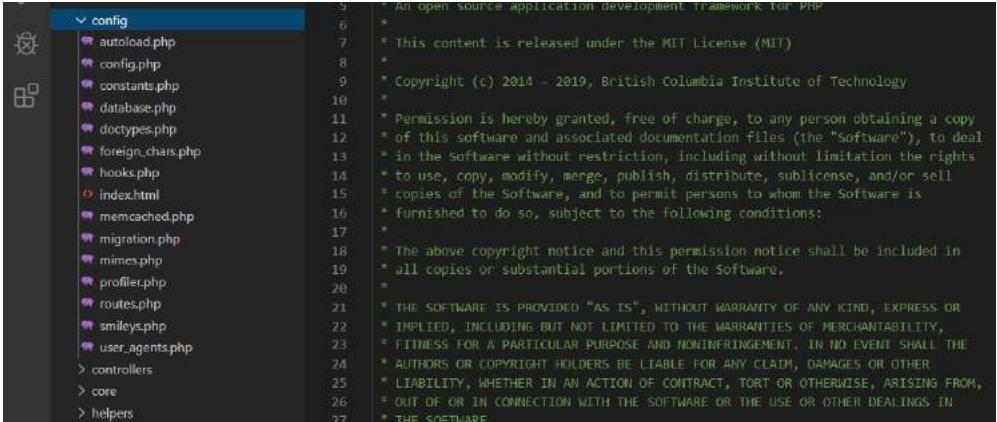
7. Lalu buka browser dan masukan <http://localhost/coba/>



8. Codeigniter siap untuk digunakan

5.21 Konfigurasi Framework Codeigniter

Walaupun CodeIgniter dapat berjalan dengan konfigurasi default, tetapi untuk sebuah aplikasi yang nyata kita harus tetap melakukan konfigurasi, setidaknya pada bagian base_url dan router. Pengaturan base_url dan router sangat berguna ketika proses pengembangan aplikasi yang banyak menggunakan helper dan library. File konfigurasi terletak dalam folder application/config. File yang terdapat pada direktori tersebut yang sering digunakan adalah file autoload.php, config.php, database.php dan routes.php. Semua konfigurasi pada codeigniter, terletak pada satu tempat yaitu di dalam folder **application/config**.



```
5 * An open source application development framework for PHP
6 *
7 * This content is released under the MIT License (MIT)
8 *
9 * Copyright (c) 2014 - 2019, British Columbia Institute of Technology
10 *
11 * Permission is hereby granted, free of charge, to any person obtaining a copy
12 * of this software and associated documentation files (the "Software"), to deal
13 * in the Software without restriction, including without limitation the rights
14 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
15 * copies of the Software, and to permit persons to whom the Software is
16 * furnished to do so, subject to the following conditions:
17 *
18 * The above copyright notice and this permission notice shall be included in
19 * all copies or substantial portions of the Software.
20 *
21 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
22 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
23 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
24 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
25 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
26 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
27 * THE SOFTWARE.
```

1. Autoload.php

File ini digunakan untuk mengatur fungsi-fungsi yang akan dimuat otomatis di awal ketika program dijalankan. Konfigurasi pada file ini bertujuan untuk mementukan sumber daya apa yang akan diload secara otomatis. Untuk melakukan konfigurasi pada file autoload.php, silahkan buka folder:

- a. *application/config/autoload.php*

The screenshot shows the Visual Studio Code interface. In the Explorer sidebar, there is a tree view of files under 'COBA'. One file, 'autoload.php', is highlighted with a blue selection bar at the bottom of the list. An arrow points from the text above to this selection bar. The main editor area displays the contents of 'autoload.php'. The code starts with a check for 'BASEPATH' and exits if it's not defined. It then defines an AUTO-LOADER section, which is described as specifying which systems should be loaded by default. The code is annotated with comments explaining its purpose.

```
<?php  
defined('BASEPATH') OR exit('No direct script access allowed');  
  
/*  
 * AUTO-LOADER  
 *  
 * This file specifies which systems should be loaded by default.  
 *  
 * In order to keep the framework as light-weight as possible only the  
 * absolute minimal resources are loaded by default. For example,  
 * the database is not connected to automatically since no assumption  
 * is made regarding whether you intend to use it. This file lets  
 * you globally define which systems you would like loaded with every  
 * request.  
 */  
  
$autoload['libraries'] = array();  
  
/*  
| -----  
| Auto-load Drivers  
| -----  
| These classes are located in system/libraries/ or in your  
| application/libraries/ directory, but are also placed inside their  
| own subdirectory and they extend the CI_Driver_Library class. They  
| offer multiple interchangeable driver options.  
|  
| Prototype:  
|  
|     $autoload['drivers'] = array('cache');  
|  
| You can also supply an alternative property name to be assigned in  
| the controller:  
|  
|     $autoload['drivers'] = array('cache' => 'cch');
```

Ada beberapa hal yang bisa diload secara otomatis diantaranya: packages, libraries, drivers, helper files, custom config files, language files, dan models. Untuk konfigurasi dasar yang perlu Anda ketahui adalah **libraries** dan **helper files**. Hal ini bertujuan agar beberapa library dan helper tertentu berjalan secara otomatis.

Contoh Libraries

```
/*  
$autoload['libraries'] = array();  
  
/*  
| -----  
| Auto-load Drivers  
| -----  
| These classes are located in system/libraries/ or in your  
| application/libraries/ directory, but are also placed inside their  
| own subdirectory and they extend the CI_Driver_Library class. They  
| offer multiple interchangeable driver options.  
|  
| Prototype:  
|  
|     $autoload['drivers'] = array('cache');  
|  
| You can also supply an alternative property name to be assigned in  
| the controller:  
|  
|     $autoload['drivers'] = array('cache' => 'cch');
```

Contoh Helper

```
$autoload['helper'] = array();  
  
/*  
 | -----  
 | Auto-load Config files  
 | -----  
 | Prototype:  
 |  
 | $autoload['config'] = array('config1', 'config2');  
 |  
 | NOTE: This item is intended for use ONLY if you have created custom  
 | config files. Otherwise, leave it blank.  
 |  
 */  
+-----+-----+-----+-----+-----+-----+
```

- b. Untuk melakukan konfigurasi pada libraries, buka file autoload.php dengan text editor seperti notepad ++, sublime text, atau lainnya.(disini menggunakan visual studio code)
- c. Kemudian temukan kode

```
$autoload['libraries'] = array();  
 ]
```

- d. Ubahlah menjadi seperti ini :

```
$autoload['libraries'] = array(['database']);
```

Keterangan:

Pada kode diatas, artinya kita meload library “database” secara otomatis. Dengan demikian Anda dapat menggunakan fungsi-fungsi database pada codeigniter. Seperti fungsi: Query Builder Class.

- e. Selanjutnya, untuk melakukan konfigurasi pada helper files, buka file autoload.php dengan text editor kemudian temukan kode berikut:

f. Kemudian temukan kode

```
$autoload['helper'] = array();
```

g. Ubahlah menjadi seperti ini :

```
$autoload['helper'] = array('url');
```

Keterangan:

Pada kode diatas, artinya kita meload helper “url” secara otomatis. Dengan demikian Anda dapat menggunakan fungsi-fungsi url pada codeigniter. Seperti fungsi: base_url(), site_url(), URI Segment, dan sebagainya

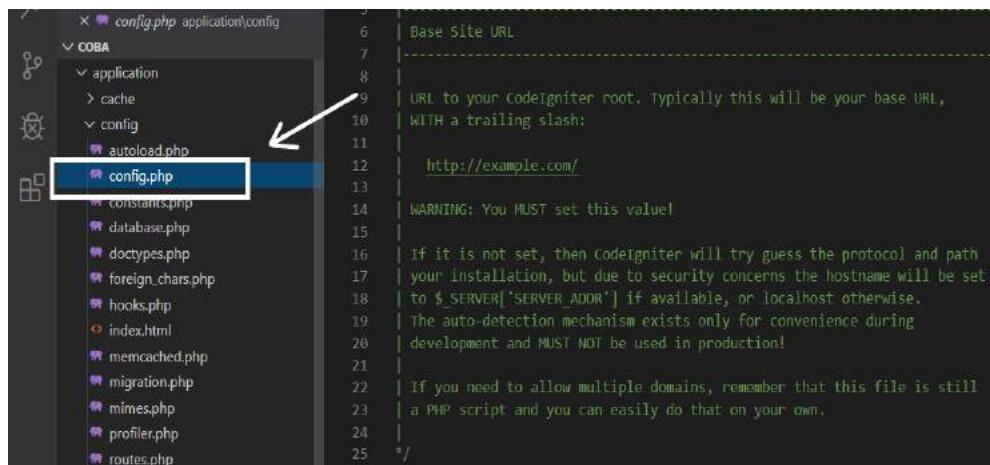
2. Config.php

```
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 /*
5 |-----
6 | Base Site URL
7 |-----
8 |
9 | URL to your CodeIgniter root. Typically this will be your base URL,
10 | WITH a trailing slash:
11 |
12 |     http://example.com/
13 |
14 | WARNING: You MUST set this value!
15 |
16 | If it is not set, then CodeIgniter will try guess the protocol and path
17 | your installation, but due to security concerns the hostname will be set
18 | to $_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
19 | The auto-detection mechanism exists only for convenience during
20 | development and MUST NOT be used in production!
21 |
22 | If you need to allow multiple domains, remember that this file is still
23 | a PHP script and you can easily do that on your own.
24 |
25 */
26 $config['base_url'] = 'http://localhost/coba/';
```

Pada file ini terdapat beberapa konfigurasi yang secara standar sudah terkonfigurasi Namun terdapat beberapa konfigurasi yang perlu diperhatikan yaitu:

```
$config['base_url']  
$config['index_page']  
$config['encryption_key']
```

- a. Untuk konfigurasi dasar, Anda cukup mengetahui konfigurasi base_url. Base_url merupakan url dasar dari project Anda.Untuk mengkonfigurasi base_url, buka file config.php dengan text editor.



```
✓ config.php application\config  
  COBA  
    application  
    cache  
    config  
      autoload.php  
      config.php  
      constants.php  
      database.php  
      doctypes.php  
      foreign_chars.php  
      hooks.php  
      index.html  
      memcached.php  
      migration.php  
      mimes.php  
      profiler.php  
      routes.php
```

5 | Base Site URL
6 | -----
7 |
8 | URL to your CodeIgniter root. Typically this will be your base URL,
9 | WITH a trailing slash:
10|
11| http://example.com/
12|
13| WARNING: You MUST set this value!
14|
15| If it is not set, then CodeIgniter will try guess the protocol and path
16| your installation, but due to security concerns the hostname will be set
17| to \$_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
18|
19| The auto-detection mechanism exists only for convenience during
20| development and MUST NOT be used in production!
21|
22| If you need to allow multiple domains, remember that this file is still
23| a PHP script and you can easily do that on your own.
24|
25|

- b. Kemudian temukan kode

```
$config['base_url'] = '';
```

- c. Ubahlah menjadi seperti ini :

```
$config['base_url'] = 'http://localhost/coba/';
```

3. Database.php

```

application > config > database.php
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  /*
5  |
6  |-----[REDACTED]-----|
7  |-----[REDACTED]-----|
8  |-----[REDACTED]-----|
9  |-----[REDACTED]-----|
10 |-----[REDACTED]-----|
11 |-----[REDACTED]-----|
12 |-----[REDACTED]-----|
13 |-----[REDACTED]-----|
14 |-----[REDACTED]-----|
15 |-----[REDACTED]-----|
16 |-----[REDACTED]-----|
17 |-----[REDACTED]-----|
18 |-----[REDACTED]-----|
19 |-----[REDACTED]-----|
20 |-----[REDACTED]-----|
21 |-----[REDACTED]-----|
22 |-----[REDACTED]-----|
23 |-----[REDACTED]-----|
24 |-----[REDACTED]-----|
25 |-----[REDACTED]-----|
26 |-----[REDACTED]-----|
27 |-----[REDACTED]-----|
28 |-----[REDACTED]-----|
29 |-----[REDACTED]-----|
30 |-----[REDACTED]-----|
31 |-----[REDACTED]-----|
32 |-----[REDACTED]-----|

```

Dilihat dari nama filenya maka Anda sudah dapat menangkap apa fungsi dari file ini. File database.php digunakan untuk melakukan konfigurasi yang berkaitan dengan konfigurasi database dari website yang akan dibuat. Adapun konfigurasi yang perlu diperhatikan yaitu: hostname, username, password, dan database.

- Hostname : menggunakan localhost karena di sini kita menggunakan database mysql dari aplikasi xampp yang terinstal di computer local.
 - Username : secara default username database mysql dari aplikasi xampp adalah root.
 - Password : secara default ini bisa dikosongi.
 - Database : ini berisikan nama database yang dibuat untuk aplikasi.
- a. Untuk melakukan konfigurasi pada database.php. Buka file database.php dengan text editor.

The screenshot shows a file structure for a 'COBA' application. Inside the 'application' folder, there is a 'config' folder containing several files: 'autoload.php', 'config.php', 'constants.php', and 'database.php'. The 'database.php' file is highlighted with a blue rectangle and has a blue arrow pointing from it to a code editor window. The code editor displays the contents of the 'database.php' file, which is a PHP script defining database connectivity settings.

```
/*
-----  
DATABASE CONNECTIVITY SETTINGS  
-----  
This file will contain the settings needed to access your database.  
For complete instructions please consult the 'Database Connection'  
page of the User Guide.  
-----  
EXPLANATION OF VARIABLES  
-----  
['dsn'] The full DSN string describe a connection to the database.  
['hostname'] The hostname of your database server.  
['username'] The username used to connect to the database.
```

b. Kemudian carilah Kode :

```
73 $active_group = 'default';  
74 $query_builder = TRUE;  
75  
76 $db['default'] = array(  
77     'dsn' => '',  
78     'hostname' => 'localhost',  
79     'username' => '',  
80     'password' => '',  
81     'database' => '',  
82     'dbdriver' => 'mysqli',  
83     'dbprefix' => '',  
84     'pconnect' => FALSE,  
85     'db_debug' => (ENVIRONMENT !== 'production'),  
86     'cache_on' => FALSE,  
87     'cachedir' => '',  
88     'char_set' => 'utf8',  
89     'dbcollat' => 'utf8_general_ci',  
90     'swap_pre' => '',  
91     'encrypt' => FALSE,  
92     'compress' => FALSE,  
93     'stricton' => FALSE,  
94     'failover' => array(),  
95     'save_queries' => TRUE  
96 );  
97
```

c. Ubahlah Kode menjadi seperti ini :

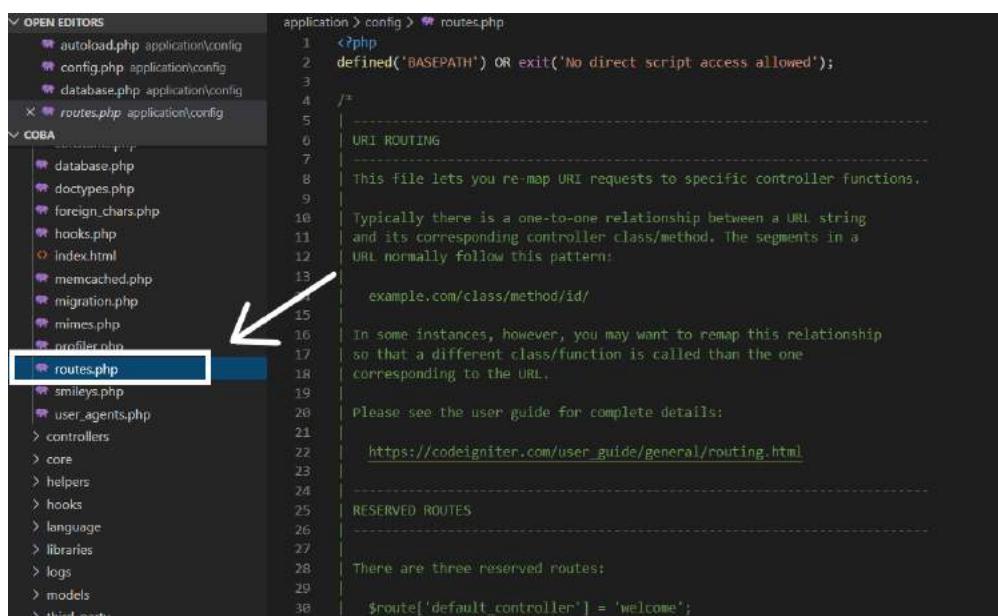
```

$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn'      => '',
    'hostname' => 'localhost', //hostname
    'username' => 'root', //username
    'password' => '', // password boleh kosong
    'database' => 'database_name',// nama database yang dibuat |
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

```

4. Routes.php



The screenshot shows a code editor interface with two panes. On the left, the 'OPEN EDITORS' sidebar lists several files under 'application/config' and 'COBA'. An arrow points from the 'routes.php' file in the sidebar to its content in the main pane. The main pane displays the 'routes.php' file content, which includes PHP code for URI routing and comments explaining the purpose of the file.

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
URI ROUTING

This file lets you remap URI requests to specific controller functions.

Typically there is a one-to-one relationship between a URL string
and its corresponding controller class/method. The segments in a
URL normally follow this pattern:

    example.com/class/method/id

In some instances, however, you may want to remap this relationship
so that a different class/function is called than the one
corresponding to the URL.

Please see the user guide for complete details:
https://codeigniter.com/user_guide/general/routing.html

RESERVED ROUTES

There are three reserved routes:
$route['default_controller'] = 'welcome';

```

Konfigurasi routing digunakan untuk memetakan permintaan atau request kedalam controller didalam website yang dibuat. Misalnya kita membuka alamat <http://www.nama-website.com>, permintaan tersebut tidak menyertakan nama controller yang ingin dibuka tetapi kita bisa secara default mengarahkannya agar secara otomatis akan membuka controller sesuai yang definisikan. Untuk melakukan konfigurasi routing buka file konfigurasi routes.php. Settingan utama yang ada adalah sebagai berikut :

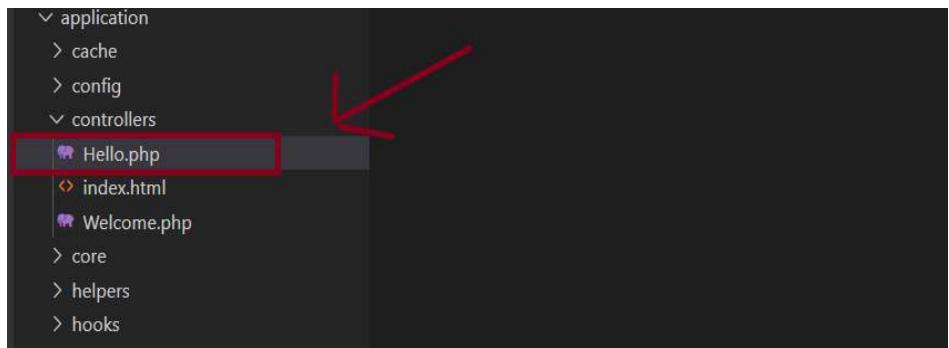
```
52     $route['default_controller'] = 'welcome';
53     $route['404_override'] = '';
54     $route['translate_uri_dashes'] = FALSE;
55
```

Artinya secara default semua permintaan yang tidak menyertakan nama controllernya akan diarahkan untuk membuka controller “welcome”. Sehingga saat alamat <http://www.nama-website.com> dibuka secara otomatis akan membuka <http://www.namawebiste.com/index.php/welcome>. Dan file tersebut dapat di edit sesuai keinginan.

5.22 Membuat ‘Hello world’ Di Codeigniter

Jika serius dengan codeigniter, Anda harus mengerti bagaimana sebuah controller bekerja. Untuk lebih jelasnya, saya akan sharing kasus sederhana agar Anda dapat memahami bagaimana controller bekerja. Disini saya mengangkat kasus yaitu bagaimana menampilkan text “Hello World” pada browser menggunakan controller. Let’s do it 😊

1. Buat sebuah controller dengan nama Hello.php seperti gambar berikut:

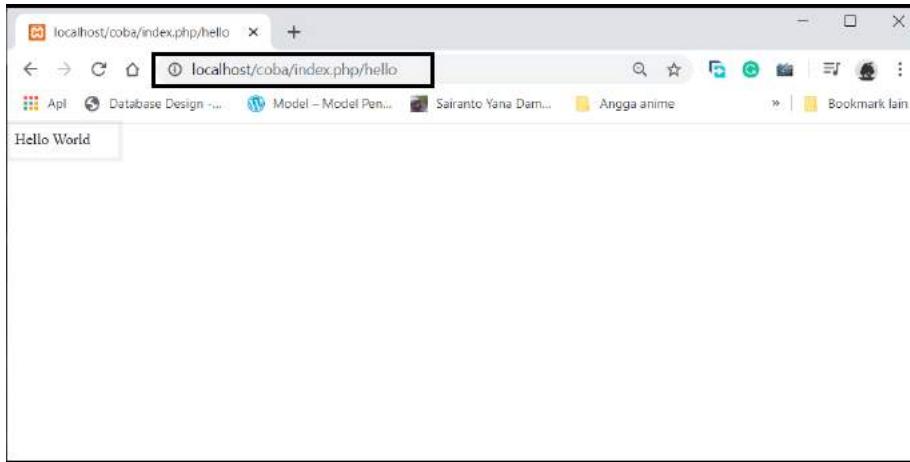


2. Kemudian ketikan kode berikut:

```
application > controllers > Hello.php
1  <?php
2  class Hello extends CI_Controller{
3
4      function index(){
5          echo "Hello World";
6      }
7
8  }
```

NB: Setiap penulisan nama file dan nama class selalu di dahului dengan huruf Capital.

3. Setelah itu save dan buka browser Anda, lalu kunjungi url berikut:
<http://localhost/coba/index.php/hello>



Jika Anda perhatikan dengan seksama, pada dasarnya url pada codeigniter terlihat seperti gambar berikut:

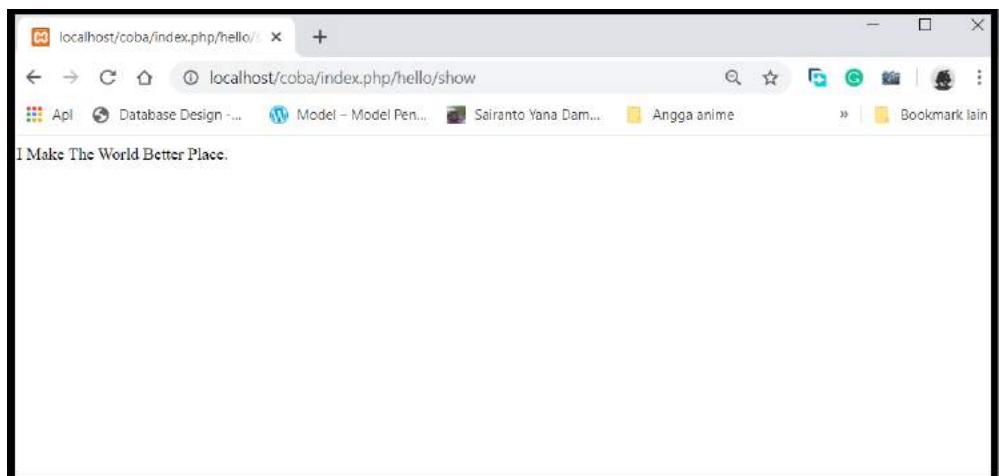


Dimana, terdapat protocol, primary domain, index.php, class name, dan function name. Mungkin terdegar rumit, tapi sebenarnya tidak.Untuk lebih jelasnya silahkan tambahkan satu function lagi pada Controller Hello.php. disini saya beri nama “show”

4. Tambahkan kodingan di Hello.php . Sehingga controller Hello.php menjadi seperti berikut:

```
1 <?php
2 class Hello extends CI_Controller{
3
4     function index(){
5         | echo "Hello World";
6     }
7
8     function show(){
9         | echo "I Make The World Better Place.";
10    }
11
12 }
```

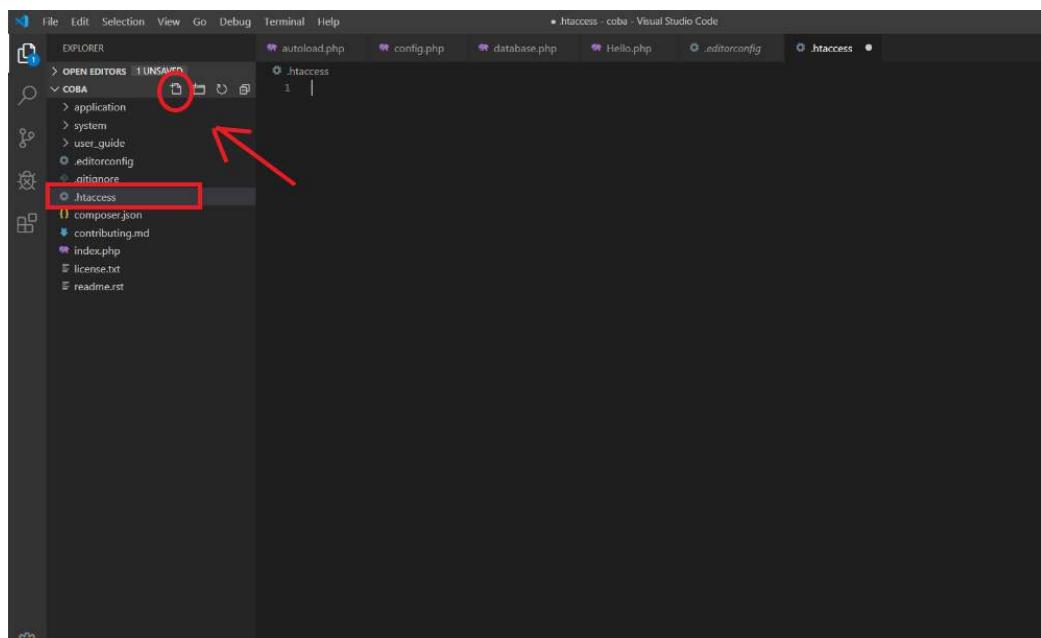
5. Jika Anda jalankan dengan mengunjungi URL berikut:
http://localhost/coba/index.php/hello/show Maka, akan tampil hasilnya seperti berikut:



5.23 Menghilangkan index.php pada URL

Codeigniter merupakan framework php yang mendukung clean URL. Dengan demikian Anda dapat membuat URL yang mudah dibaca dan sekaligus SEO Friendly. Pada URL aplikasi “Hello World” diatas, dapat dilihat bahwa adanya index.php pada url yang terlihat mengganggu. Adakah cara untuk menghilangkan index.php dari URL? Tentu saja, Anda dapat menggunakan file .htaccess untuk menghilangkannya. Bagaimana membuat file .htaccess? Mari kita mulai.

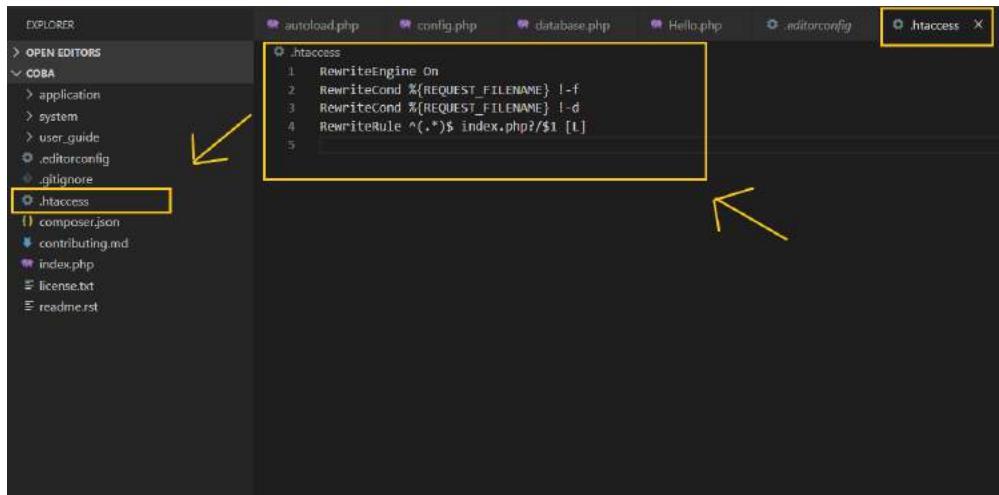
1. Buka terlebih dahulu file codeigniter yang digunakan dengan tex editor yang digunakan (menggunakan visual studio code). Buat sebuah file dengan nama .htaccess pada web root Anda seperti ini



2. Ini merupakan kodingan yang digunakan :

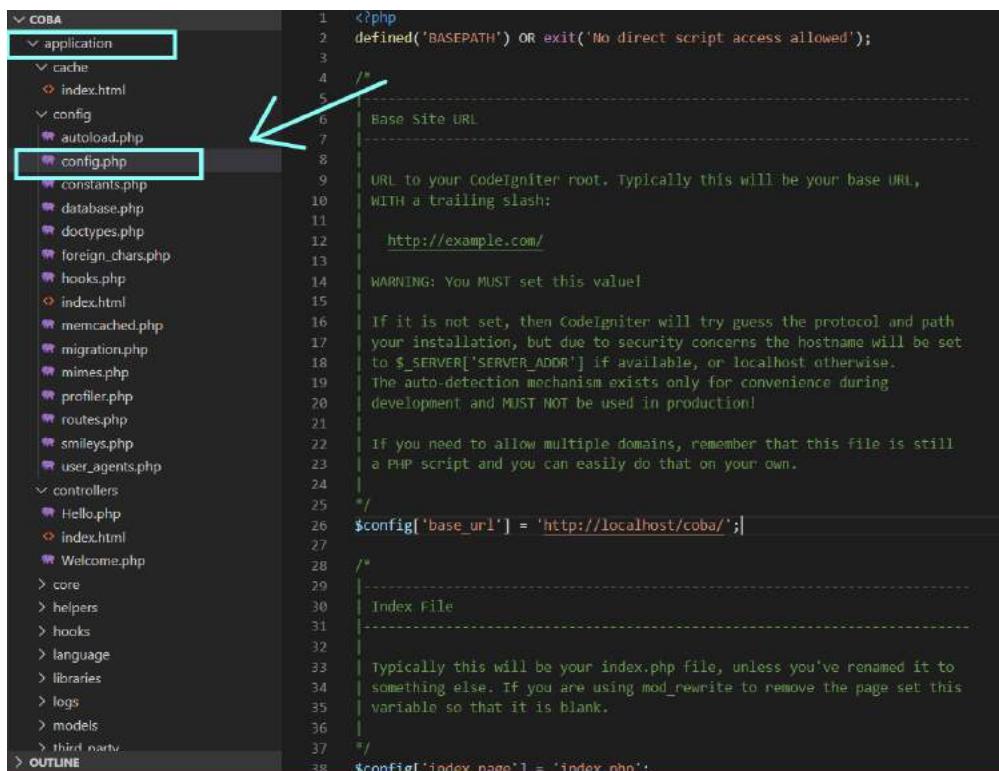
```
RewriteEngine On  
RewriteCond %{REQUEST_FILENAME} !-f  
RewriteCond %{REQUEST_FILENAME} !-d  
RewriteRule ^(.*)$ index.php?/$1 [L]
```

3. Lalu tuliskan kodingan tersebut pada .htaccess di text editor yang digunakan



```
COBA/.htaccess
1 RewriteEngine On
2 RewriteCond %{REQUEST_FILENAME} !-f
3 RewriteCond %{REQUEST_FILENAME} !-d
4 RewriteRule ^(.*)$ index.php/$1 [L]
5
```

4. Kemudian buka folder application/config/config.php dengan text editor.



```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
Base Site URL
URL to your CodeIgniter root. Typically this will be your base URL,
WITH a trailing slash:
http://example.com/
WARNING: You MUST set this value!
If it is not set, then CodeIgniter will try guess the protocol and path
your installation, but due to security concerns the hostname will be set
to $_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
The auto-detection mechanism exists only for convenience during
development and MUST NOT be used in production!
if you need to allow multiple domains, remember that this file is still
a PHP script and you can easily do that on your own.
*/
$config['base_url'] = 'http://localhost/coba/';

/*
Index File
typically this will be your index.php file, unless you've renamed it to
something else. If you are using mod_rewrite to remove the page set this
variable so that it is blank.
*/
$config['index_page'] = 'index.php';
```

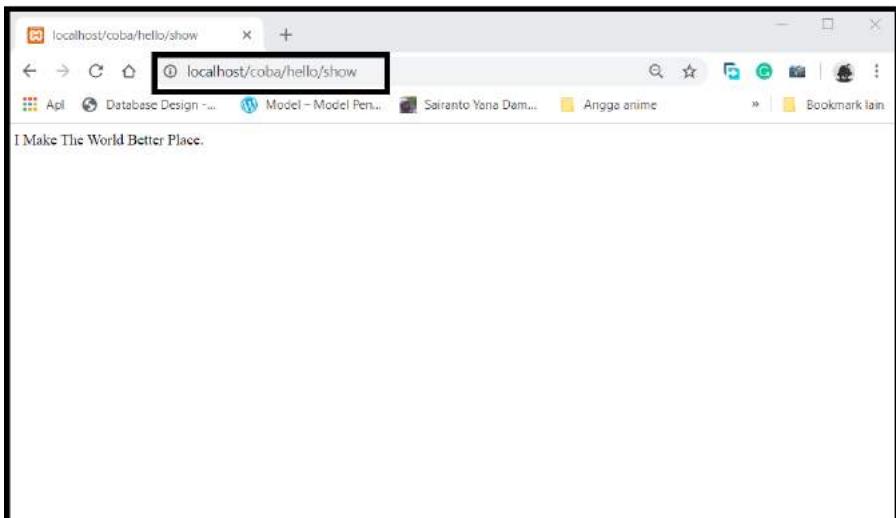
5. Kemudian temukan kode berikut:

```
*/  
$config['index_page'] = 'index.php';
```

6. Kemudian ubah menjadi seperti ini

```
$config['index_page'] = '';
```

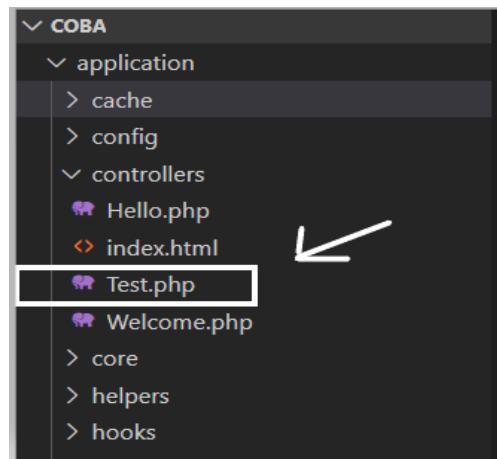
7. Sekarang silahkan kunjung url berikut untuk uji coba:
<http://localhost/coba/hello/show>. Maka akan terlihat hasilnya seperti berikut:



5.24 Controller dan View

Pada masalah sebelumnya, Anda telah mengetahui bagaimana menampilkan text “Hello World” langsung dari controller. Namun, hal tersebut sebaiknya dilakukan di view. Sekarang saya akan menunjukkan bagaimana cara menampilkan view melalui controller.

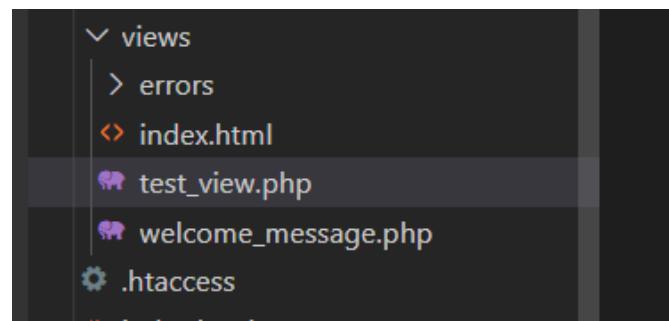
- Pertama, buat sebuah file pada *application/controller* dengan nama Test.php.



- Kemudian ketikan kode berikut:

```
1 <?php
2 class Test extends CI_Controller
3 {
4     function __construct()
5     {
6         parent::__construct();
7     }
8
9     function index(){
10        $this->load->view('test_view');
11    }
12 }
13
14 }
```

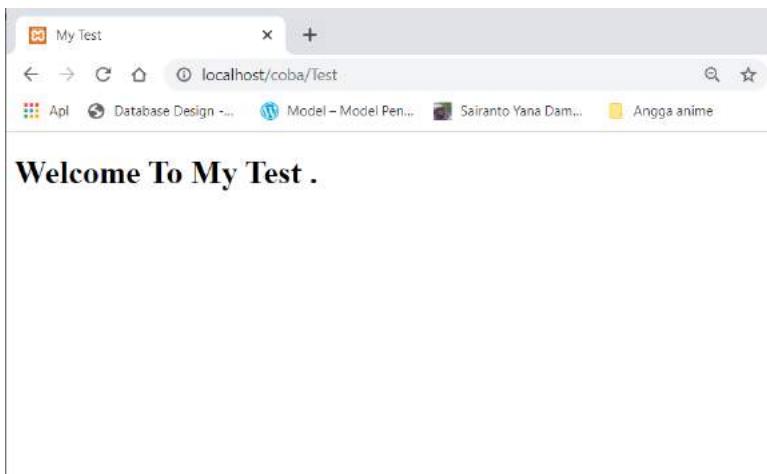
- Kemudian buat view pada *application/view* dengan nama **test_view.php**



4. Kemudian masukan kode pada view

```
application > views > test_view.php
1   <!DOCTYPE html>
2   <html lang="en">
3     <head>
4       <meta charset="utf-8">
5       <title>My Test</title>
6     </head>
7     <body>
8       <h1>Welcome To My Test</h1>
9     </body>
10    </html>
```

5. Lalu buka browser dengan akses controller Test



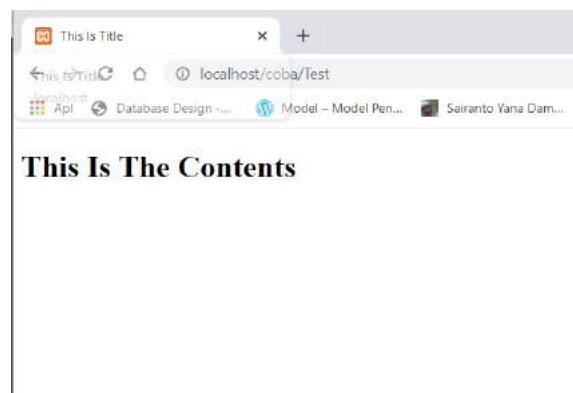
6. Anda juga dapat mengirimkan parameter ke view melalui controller. Sebagai contoh, silahkan ubah controller Test.php menjadi seperti berikut:

```
application > controllers > Test.php
1  <?php
2  class Test extends CI_Controller
3  {
4      function __construct()
5      {
6          parent::__construct();
7      }
8
9      function index(){
10         $data['title'] = "This Is Title";
11         $data['content'] = "This Is The Contents";
12         $this->load->view('test_view',$data);
13     }
14 }
15 }
```

7. Kemudian ubah view test_view.php menjadi seperti berikut:

```
application > views > test_view.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="utf-8">
5      <title><?php echo $title;?></title>
6  </head>
7  <body>
8      <h1><?php echo $content;?></h1>
9  </body>
10 </html>
```

8. Kemudian, buka browser Anda dan akses kembali controller Test. Maka akan terlihat hasilnya seperti berikut:



BAB VI

Contoh Program

Dimana dalam bab ini membahas mengenai tahapan membuat sebuah program yang mudah dengan menggunakan framework codeigniter

6.1 Persiapan Pembuatan Aplikasi

Dalam Membuat sebuah aplikasi atau sistem harus mempersiapkan tools yang digunakan untuk menunjang dalam pembuatan aplikasi tersebut yaitu

Tools	Kegunaan
Xampp	Sebagai tempat pembuatan Database dan segala yang berhubungan dengan pengolahan data
Visual Studio Code	Digunakan Sebagai tempat pengetikan kodingan dalam hal Pembuatan aplikasi
Framework Codeigniter	Framework yang digunakan pada aplikasi yang dibuat

6.2 Memasukan Template ke Codeigniter

6.2.1 Pengenalan Template

Web template terdiri dari kata web dan template. Dalam bidang ilmu komputer, arti kata template adalah : “Sebuah dokumen atau file yang memiliki format preset, digunakan sebagai titik awal untuk aplikasi tertentu sehingga format tidak harus diciptakan kembali setiap kali digunakan”. Sedangkan pengertian web template itu sendiri dalam situs wikipedia disebutkan bahwa :

“Web template merupakan komponen dasar dari suatu sistem template website yang memisahkan antara bagian konten dengan bagian presentasi dari suatu desain web. Web template digunakan untuk memproduksi dokumen web secara massal”.

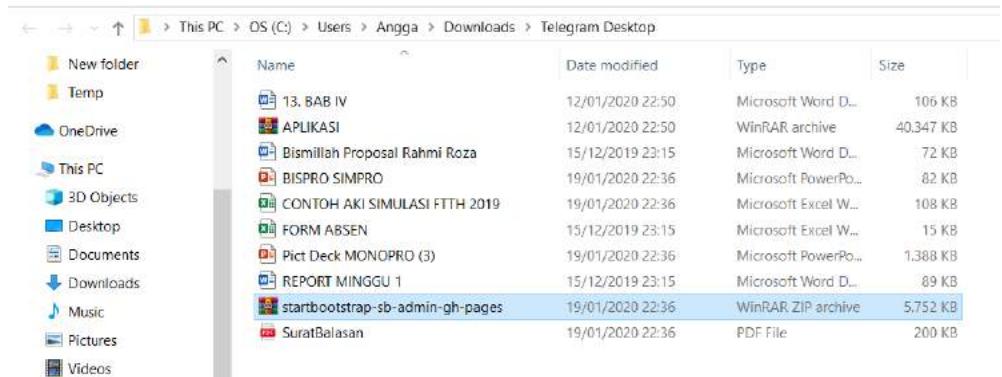
Dan pengertian yang sangat umum dari desain template adalah pola layout dari sebuah website yang ingin didesain dan disiapkan dengan berbagai fitur, content suatu bidang di dalamnya. Kita juga bisa merubah serta mengedit serta mengembangkan kembali dari desain yang di rencanakan.

6.2.2 Mengitergrasikan SB Admin ke Codeigniter

Template SB Admin adalah salah satu template yang paling sering digunakan oleh web developer sebagai backend template pada proyek yang sedang dikerjakan. template ini dibuat menggunakan framework bootstrap yang merupakan framework CSS yang paling banyak digunakan di kalangan web desainer sekarang. Disini kita akan belajar bagaimana cara menjadikan menu pada template SB Admin agar menjadi dinamis dengan data berasal dari database.

Catatan : (proses integrasi ini bisa menggunakan template yang berbeda yang saya gunakan , tetapi tata caranya sama)

1. Download Terlebih template SB Admin hingga selesai (Bisa mendownload template lain sesuai dengan kebutuhan)Link : (getbootstrap.com)



2. Selanjutnya Extract file

📁 startbootstrap-sb-admin-gh-pages	14/06/2019 13:27	File folder
📁 system	19/09/2019 8:55	File folder
📁 Tor	05/01/2020 0:56	File folder
📁 tugasbesar_Ai	23/06/2019 20:56	File folder
📁 uploads	29/12/2018 21:54	File folder
📁 vendor	29/12/2018 11:10	File folder
📁 webalizer	14/05/2019 17:28	File folder

3. Kemudian Download dan Extract Framework Codeigniter (seperti penjelasan di bab sebelumnya) hingga seperti ini gambar di bawah ini.

📁 application	17/12/2018 22:03	File folder
📁 Bot	26/01/2020 8:03	File folder
📁 coba	07/02/2020 19:22	File folder
📁 dashboard	14/05/2019 17:28	File folder
📁 images	29/12/2018 10:34	File folder
📁 img	14/05/2019 17:28	File folder

4. Selanjutnya konfigurasi terlebih dahulu framework codeigniter
5. Buatlah folder baru pada framework Codeigniter yang diberi Assets seperti ini :

📁 application	25/01/2020 13:00	File folder
📁 assets	08/02/2020 10:53	File folder
📁 system	25/01/2020 13:00	File folder
📁 user_guide	07/02/2020 19:20	File folder
📄 .editorconfig	19/09/2019 19:08	EDITORCONFIG File 1 KB
📄 .gitignore	19/09/2019 19:08	Text Document 1 KB
📄 .htaccess	07/02/2020 19:22	HTACCESS File 1 KB
📄 composer.json	19/09/2019 19:08	JSON File 1 KB
📄 contributing.md	19/09/2019 19:08	MD File 7 KB
📄 index.php	19/09/2019 19:08	PHP File 11 KB
📄 license	19/09/2019 19:08	Text Document 2 KB
📄 readme.rst	19/09/2019 19:08	RST File 3 KB

6. Buka kembali folder Template admin tersebut lalu copy folder yang ada di dalam template tersebut

OS (C:) > xampp > htdocs > startbootstrap-sb-admin-gh-pages

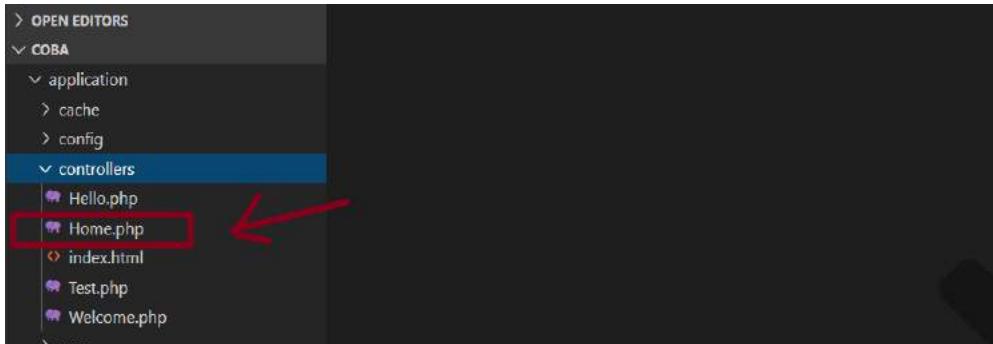
Name	Date modified	Type	Size
css	14/06/2019 13:27	File folder	
js	14/06/2019 13:27	File folder	
scss	14/06/2019 13:27	File folder	
vendor	14/06/2019 13:27	File folder	
.gitignore	14/06/2019 13:27	Text Document	1 KB
.travis	14/06/2019 13:27	YML File	1 KB
404	14/06/2019 13:27	HTML File	8 KB
blank	14/06/2019 13:27	HTML File	8 KB
charts	14/06/2019 13:27	HTML File	9 KB
forgot-password	14/06/2019 13:27	HTML File	2 KB
gulpfile	14/06/2019 13:27	JavaScript File	4 KB
index	14/06/2019 13:27	HTML File	28 KB
LICENSE	14/06/2019 13:27	File	2 KB
login	25/11/2019 14:27	HTML File	3 KB
package.json	14/06/2019 13:27	JSON File	2 KB
package-lock.json	14/06/2019 13:27	JSON File	254 KB
README.md	14/06/2019 13:27	MD File	5 KB
register	25/11/2019 14:26	HTML File	4 KB
tables	14/06/2019 13:27	HTML File	25 KB

7. Pindahkan folder-folder tersebut ke dalam folder Assets yang telah dibuat tadi

OS (C:) > xampp > htdocs > coba > assets

Name	Date modified	Type	Size
css	08/02/2020 11:44	File folder	
js	08/02/2020 11:44	File folder	
scss	08/02/2020 11:44	File folder	
vendor	08/02/2020 11:44	File folder	

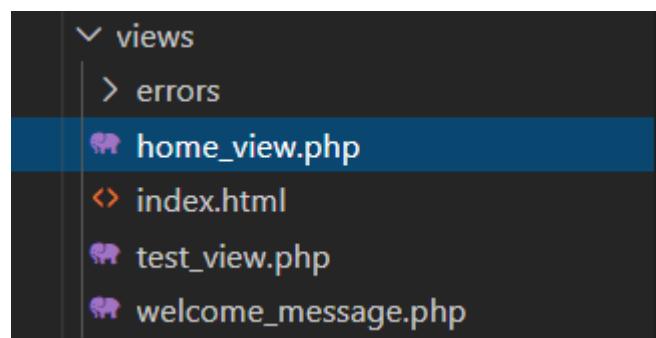
8. Selanjutnya buka folder codeigniter yang dibuat pada text editor
9. Buka Controller lalu buat file baru dengan nama Home.php(pembuatan nama bebas)



10. Lalu ketikan kodingan pada Home.php seperti berikut

```
1 <?php
2 class Home extends CI_Controller
3 {
4     function __construct()
5     {
6         parent::__construct();
7     }
8
9     function index(){
10        $this->load->view('home_view');
11    }
12
13 }
```

11. Lalu buat view dengan nama Home_view.php



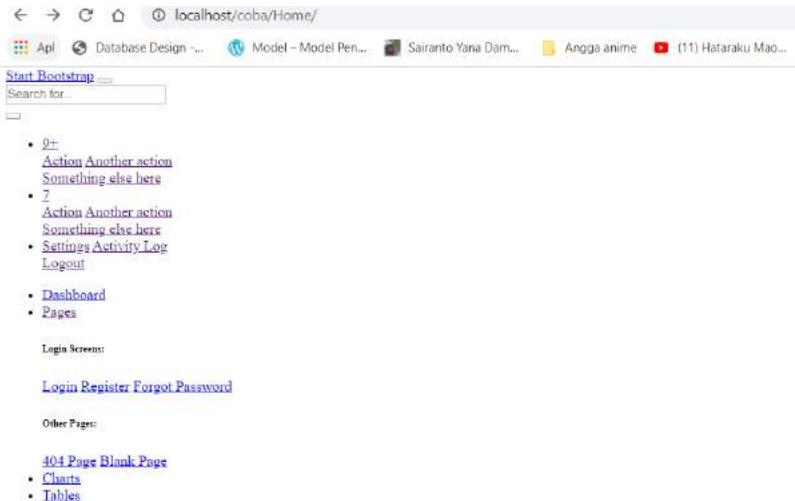
12. Lalu copy kodingan yang terdapat di index.html pada template sb Admin

css	14/06/2019 13:27	File folder
js	14/06/2019 13:27	File folder
scss	14/06/2019 13:27	File folder
vendor	14/06/2019 13:27	File folder
.gitignore	14/06/2019 13:27	Text Document
.travis	14/06/2019 13:27	YML File
404	14/06/2019 13:27	HTML File
blank	14/06/2019 13:27	HTML File
charts	14/06/2019 13:27	HTML File
forgot-password	14/06/2019 13:27	HTML File
gulpfile	14/06/2019 13:27	JavaScript File
index	14/06/2019 13:27	HTML File
LICENSE	14/06/2019 13:27	File
login	25/11/2019 14:27	HTML File

13. Paste Kodingan Index ke home_view.php

```
home_view.php ×
application > views > home_view.php
714     <div class="content my-auto" style="background-color: #f0f0f0;">
715         <div class="copyright text-center my-auto">
716             <span>Copyright © Your Website 2019</span>
717         </div>
718     </div>
719     </footer>
720 
721     </div>
722     <!-- /.content-wrapper -->
723 
724 </div>
725 <!-- /#wrapper -->
726 
727 <!-- Scroll to Top Button-->
728 <a class="scroll-to-top rounded" href="#page-top">
729     <i class="fas fa-angle-up"></i>
730 </a>
731 
732 <!-- Logout Modal-->
733 <div class="modal fade" id="logoutModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
734     <div class="modal-dialog" role="document">
735         <div class="modal-content">
736             <div class="modal-header">
737                 <h5 class="modal-title" id="exampleModalLabel">Ready to Leave?</h5>
738                 <button class="close" type="button" data-dismiss="modal" aria-label="Close">
739                     <span aria-hidden="true">×</span>
740                 </button>
741             </div>
742             <div class="modal-body">
743                 ...
744             </div>
745             <div class="modal-footer">
746                 <button class="btn btn-primary" type="button" data-dismiss="modal" aria-label="Accept">Accept</button>
747                 <button class="btn btn-secondary" type="button" data-dismiss="modal" aria-label="Cancel">Cancel</button>
748             </div>
749         </div>
750     </div>
751 </div>
```

14. Lalu jalankan di browser hasilnya seperti ini



Mengapa template sb tidak terpanggil, karena kita belum memanggil library pendukungnya.

Kuncinya ada di <?php echo base_url('assets/..')?>, kode itu mengarahkan pada library yang digunakan. Jadi, silahkan ganti ../../ dengan <?php echo base_url('assets/..')?>

15. Atau jika Anda masih bingung, silahkan ganti kode home_view.php dengan kode di bawah ini:

```
home_view.php
<!DOCTYPE html>
<html lang="en">

<head>

<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">

<title>SB Admin - Dashboard</title>

<!-- Custom fonts for this template-->
<link href="<?php echo base_url('assets/vendor/fontawesome-free/css/all.min.css') ?>" rel="stylesheet" type="text/css">
```

```

<!-- Page level plugin CSS-->
<link href=<?php echo base_url('assets/vendor/datatables/dataTables.bootstrap4.css') ?>" rel="stylesheet">

<!-- Custom styles for this template-->
<link href=<?php echo base_url('assets/css/sb-admin.css') ?>" rel="stylesheet">

</head>

<body id="page-top">

<nav class="navbar navbar-expand navbar-dark bg-dark static-top">

    <a class="navbar-brand mr-1" href="index.html">Start Bootstrap</a>

    <button class="btn btn-link btn-sm text-white order-1 order-sm-0" id="sidebarToggle" href="#">
        <i class="fas fa-bars"></i>
    </button>

    <!-- Navbar Search -->
    <form class="d-none d-md-inline-block form-inline ml-auto mr-0 mr-md-3 my-2 my-md-0">
        <div class="input-group">
            <input type="text" class="form-control" placeholder="Search for..." aria-label="Search" aria-describedby="basic-addon2">
            <div class="input-group-append">
                <button class="btn btn-primary" type="button">
                    <i class="fas fa-search"></i>
                </button>
            </div>
        </div>
    </form>

    <!-- Navbar -->
    <ul class="navbar-nav ml-auto ml-md-0">
        <li class="nav-item dropdown no-arrow mx-1">
            <a class="nav-link dropdown-toggle" href="#" id="alertsDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                <i class="fas fa-bell fa-fw"></i>
                <span class="badge badge-danger">9+</span>
            </a>
        </li>
    </ul>
</nav>

```

```

<div class="dropdown-menu dropdown-menu-right" aria-
labelledby="alertsDropdown">
    <a class="dropdown-item" href="#">Action</a>
    <a class="dropdown-item" href="#">Another action</a>
    <div class="dropdown-divider"></div>
    <a class="dropdown-item" href="#">Something else here</a>
</div>
</li>
<li class="nav-item dropdown no-arrow mx-1">
    <a class="nav-link dropdown-toggle" href="#" id="messagesDropdown" role="button" data-toggle="dropdown" aria-
haspopup="true" aria-expanded="false">
        <i class="fas fa-envelope fa-fw"></i>
        <span class="badge badge-danger">7</span>
    </a>
    <div class="dropdown-menu dropdown-menu-right" aria-
labelledby="messagesDropdown">
        <a class="dropdown-item" href="#">Action</a>
        <a class="dropdown-item" href="#">Another action</a>
        <div class="dropdown-divider"></div>
        <a class="dropdown-item" href="#">Something else here</a>
    </div>
</li>
<li class="nav-item dropdown no-arrow">
    <a class="nav-link dropdown-toggle" href="#" id="userDropdown" role="button" data-toggle="dropdown" aria-
haspopup="true" aria-
expanded="false">
        <i class="fas fa-user-circle fa-fw"></i>
    </a>
    <div class="dropdown-menu dropdown-menu-right" aria-
labelledby="userDropdown">
        <a class="dropdown-item" href="#">Settings</a>
        <a class="dropdown-item" href="#">Activity Log</a>
        <div class="dropdown-divider"></div>
        <a class="dropdown-item" href="#" data-toggle="modal" data-
target="#logoutModal">Logout</a>
    </div>
</li>
</ul>

</nav>

<div id="wrapper">

    <!-- Sidebar -->
    <ul class="sidebar navbar-nav">

```

```

<li class="nav-item active">
  <a class="nav-link" href="index.html">
    <i class="fas fa-fw fa-tachometer-alt"></i>
    <span>Dashboard</span>
  </a>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" id="pagesDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    <i class="fas fa-fw fa-folder"></i>
    <span>Pages</span>
  </a>
  <div class="dropdown-menu" aria-labelledby="pagesDropdown">
    <h6 class="dropdown-header">Login Screens:</h6>
    <a class="dropdown-item" href="login.html">Login</a>
    <a class="dropdown-item" href="register.html">Register</a>
    <a class="dropdown-item" href="forgot-password.html">Forgot Password</a>
    <div class="dropdown-divider"></div>
    <h6 class="dropdown-header">Other Pages:</h6>
    <a class="dropdown-item" href="404.html">404 Page</a>
    <a class="dropdown-item" href="blank.html">Blank Page</a>
  </div>
</li>
<li class="nav-item">
  <a class="nav-link" href="charts.html">
    <i class="fas fa-fw fa-chart-area"></i>
    <span>Charts</span>
  </a>
</li>
<li class="nav-item">
  <a class="nav-link" href="tables.html">
    <i class="fas fa-fw fa-table"></i>
    <span>Tables</span>
  </a>
</li>
</ul>

<div id="content-wrapper">
  <div class="container-fluid">
    <!-- Breadcrumbs-->
    <ol class="breadcrumb">
      <li class="breadcrumb-item">
        <a href="#">Dashboard</a>
      </li>
    </ol>
  </div>
</div>

```

```

<li class="breadcrumb-item active">Overview</li>
</ol>

<!-- Icon Cards-->
<div class="row">
  <div class="col-xl-3 col-sm-6 mb-3">
    <div class="card text-white bg-primary o-hidden h-100">
      <div class="card-body">
        <div class="card-body-icon">
          <i class="fas fa-fw fa-comments"></i>
        </div>
        <div class="mr-5">26 New Messages!</div>
      </div>
      <a class="card-footer text-white clearfix small z-1" href="#">
        <span class="float-left">View Details</span>
        <span class="float-right">
          <i class="fas fa-angle-right"></i>
        </span>
      </a>
    </div>
  </div>
  <div class="col-xl-3 col-sm-6 mb-3">
    <div class="card text-white bg-warning o-hidden h-100">
      <div class="card-body">
        <div class="card-body-icon">
          <i class="fas fa-fw fa-list"></i>
        </div>
        <div class="mr-5">11 New Tasks!</div>
      </div>
      <a class="card-footer text-white clearfix small z-1" href="#">
        <span class="float-left">View Details</span>
        <span class="float-right">
          <i class="fas fa-angle-right"></i>
        </span>
      </a>
    </div>
  </div>
  <div class="col-xl-3 col-sm-6 mb-3">
    <div class="card text-white bg-success o-hidden h-100">
      <div class="card-body">
        <div class="card-body-icon">
          <i class="fas fa-fw fa-shopping-cart"></i>
        </div>
        <div class="mr-5">123 New Orders!</div>
      </div>
      <a class="card-footer text-white clearfix small z-1" href="#">

```

```

<span class="float-left">View Details</span>
<span class="float-right">
  <i class="fas fa-angle-right"></i>
</span>
</a>
</div>
</div>
<div class="col-xl-3 col-sm-6 mb-3">
  <div class="card text-white bg-danger o-hidden h-100">
    <div class="card-body">
      <div class="card-body-icon">
        <i class="fas fa-fw fa-life-ring"></i>
      </div>
      <div class="mr-5">13 New Tickets!</div>
    </div>
    <a class="card-footer text-white clearfix small z-1" href="#">
      <span class="float-left">View Details</span>
      <span class="float-right">
        <i class="fas fa-angle-right"></i>
      </span>
    </a>
  </div>
</div>
</div>

<!-- Area Chart Example-->
<div class="card mb-3">
  <div class="card-header">
    <i class="fas fa-chart-area"></i>
    Area Chart Example</div>
  <div class="card-body">
    <canvas id="myAreaChart" width="100%" height="30"></canvas>
  </div>
  <div class="card-footer small text-muted">Updated yesterday at 11:59 PM</div>
</div>

<!-- DataTables Example -->
<div class="card mb-3">
  <div class="card-header">
    <i class="fas fa-table"></i>
    Data Table Example</div>
  <div class="card-body">
    <div class="table-responsive">

```

```


| Name            | Position             | Office        | Age | Start date | Salary    |
|-----------------|----------------------|---------------|-----|------------|-----------|
| Name            | Position             | Office        | Age | Start date | Salary    |
| Jonas Alexander | Developer            | San Francisco | 30  | 2010/07/14 | \$86,500  |
| Shad Decker     | Regional Director    | Edinburgh     | 51  | 2008/11/13 | \$183,000 |
| Michael Bruce   | Javascript Developer | Singapore     | 29  | 2011/06/27 | \$183,000 |


```

```

</tr>
<tr>
    <td>Donna Snider</td>
    <td>Customer Support</td>
    <td>New York</td>
    <td>27</td>
    <td>2011/01/25</td>
    <td>$112,000</td>
</tr>
</tbody>
</table>
</div>
</div>
<div class="card-footer small text-muted">Updated yesterday at
11:59 PM</div>
</div>

</div>
<!-- /.container-fluid -->

<!-- Sticky Footer -->
<footer class="sticky-footer">
    <div class="container my-auto">
        <div class="copyright text-center my-auto">
            <span>Copyright © Your Website 2019</span>
        </div>
    </div>
</footer>

</div>
<!-- /.content-wrapper -->

</div>
<!-- /#wrapper -->

<!-- Scroll to Top Button-->
<a class="scroll-to-top rounded" href="#page-top">
    <i class="fas fa-angle-up"></i>
</a>

<!-- Logout Modal-->
<div class="modal fade" id="logoutModal" tabindex="-1"
role="dialog" aria-labelledby="exampleModalLabel" aria-
hidden="true">
    <div class="modal-dialog" role="document">
        <div class="modal-content">

```

```

<div class="modal-header">
    <h5 class="modal-title" id="exampleModalLabel">Ready to
Leave?</h5>
    <button class="close" type="button" data-dismiss="modal" aria-
label="Close">
        <span aria-hidden="true">×</span>
    </button>
</div>
<div class="modal-body">Select "Logout" below if you are ready
to end your current session.</div>
<div class="modal-footer">
    <button class="btn btn-secondary" type="button" data-
dismiss="modal">Cancel</button>
    <a class="btn btn-primary" href="login.html">Logout</a>
</div>
</div>
</div>

<!-- Bootstrap core JavaScript-->
<script src="php echo base_url('assets/vendor/jquery/jquery.min.js')?&gt;""></script>
<script src="php echo base_url('assets/vendor/bootstrap/js/bootstrap.bundle.min.js')?&gt;""></scri
pt>

<!-- Core plugin JavaScript-->
<script src="php echo base_url('assets/vendor/jquery-easing.min.js')?&gt;""></script>

<!-- Page level plugin JavaScript-->
<script src="php echo base_url('assets/vendor/chart.js/Chart.min.js')?&gt;""></script>
<script src="php echo base_url('assets/vendor/datatables/jquery.dataTables.js')?&gt;""></script>
<script src="php echo base_url('assets/vendor/datatables/dataTables.bootstrap4.js')?&gt;""></scri
t>

<!-- Custom scripts for all pages-->
<script src="php echo base_url('assets/js/sb-
admin.min.js')?&gt;""></script>

<!-- Demo scripts for this page-->
<script src="php echo base_url('assets/js/demo/datatables-
demo.js')?&gt;""></script>

```

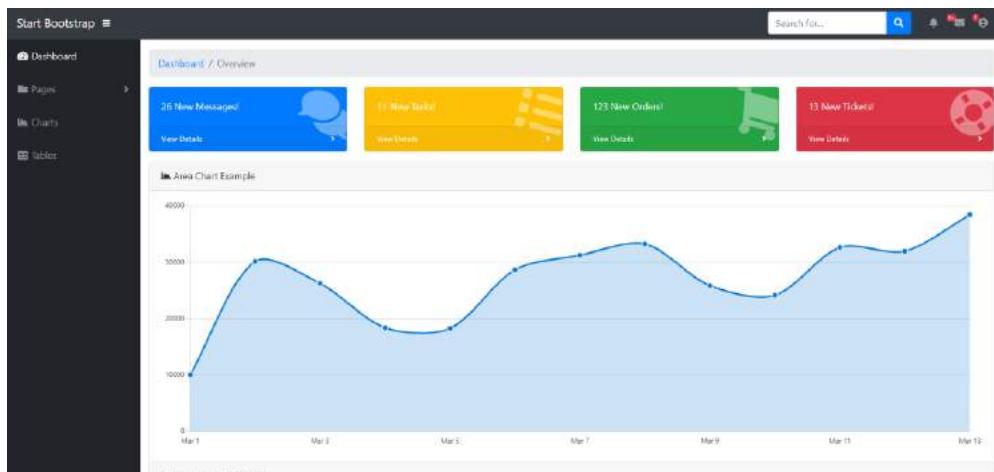
```
<script src="php echo base_url('assets/js/demo/chart-area-demo.js')?&gt;"&gt;&lt;/script&gt;</pre

```

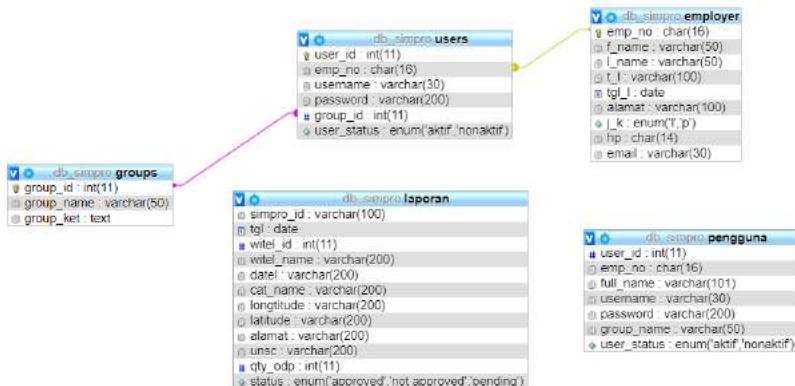
```
</body>
```

```
</html>
```

16. Hasilnya jika sudah diubah akan menjadi seperti ini



6.3 Racangan Database



6.4 Membuat Model

Sebagai Penghubung dengan database

1. Buat Model dengan Nama Bebas M_simros.php

M_simros.php

```
<?php

/**
 *
 */

class M_simpro extends CI_Model
{

    function edit_data($where, $table) {
        return $this->db->get_where($table,$where);
    }

    function get_data($table) {
        return $this->db->get($table);
    }

    function insert_data($data, $table) {
        return $this->db->insert($table,$data);
    }

    function update_data($where,$data,$table) {
        $this->db->where($where);
        $this->db->update($table,$data);
    }

    public function delete_data($where,$table) {
        $this->db->where($where);
        $this->db->delete($table);
    }

    function get_trs_auto(){
        $q   = $this->db->query("SELECT MAX(RIGHT(rek_no,4))
AS      kd_max      FROM      transaksi_sampah      WHERE
DATE(trs_tgl)=CURDATE()");
    }
}
```

```

$kd = "";
if($q->num_rows()>0){
    foreach($q->result() as $k){
        $tmp = ((int)$k->kd_max)+1;
        $kd = sprintf("%04s", $tmp);
    }
}else{
    $kd = "0001";
}
date_default_timezone_set('Asia/Jakarta');
return "TRS".date('dmy').$kd;
}

/*
function get_notrans_auto(){
    $q = $this->db->query("SELECT MAX(RIGHT(trs_no,4)) AS
kd_max      FROM      transaksi_sampah      WHERE
DATE(trs_tgl)=CURDATE()");
    $kd = "";
    if($q->num_rows()>0){
        foreach($q->result() as $k){
            $tmp = ((int)$k->kd_max)+1;
            $kd = sprintf("%04s", $tmp);
        }
    }else{
        $kd = "0001";
    }
    date_default_timezone_set('Asia/Jakarta');
    return "TRS".date('dmy').$kd;
}
*/
}

function get_rek_auto(){
    $q = $this->db->query("SELECT MAX(RIGHT(rek_no,4))
AS      kd_max      FROM      rekening      WHERE
DATE(rek_tgl)=CURDATE()");
    $kd = "";
    if($q->num_rows()>0){
        foreach($q->result() as $k){
            $tmp = ((int)$k->kd_max)+1;
        }
    }
}

```

```

        $kd = sprintf("%04s", $tmp);
    }
}else{
    $kd = "0001";
}
date_default_timezone_set('Asia/Jakarta');
return "1001".date('dmy').$kd;
}

function get_no_pengajuan(){
    $q = $this->db->query("SELECT MAX(RIGHT(simpro_id,4))
AS kd_max FROM simpro WHERE DATE(tgl)=CURDATE()");
    $kd = "";
    if($q->num_rows()>0){
        foreach($q->result() as $k){
            $tmp = ((int)$k->kd_max)+1;
            $kd = sprintf("%04s", $tmp);
        }
    }else{
        $kd = "0001";
    }
    date_default_timezone_set('Asia/Jakarta');
    return "P".date('dmy').$kd;
}

function get_data_saldo_bycif($cif){
    $hsl=$this->db->query("SELECT * FROM cek_saldo_nasabah
WHERE cif='$cif'");
    if($hsl->num_rows()>0){
        foreach ($hsl->result() as $data2) {
            $hasil=array(
                'cif' => $data2->cif,
                'nik' => $data2->nik,
                'pemilik' => $data2->pemilik,
                'saldo' => $data2->saldo,
            );
        }
    }
    return $hasil;
}

```

```
    }  
?>
```

6.5 Membuat login

1. Pertama Buat terlebih dahulu controller dengan nama login.php dan ketik kodingannya sebagai berikut

Admin_login.php

```
?php  
defined('BASEPATH') OR exit('No direct script access allowed');  
  
class Admin_login extends CI_Controller {  
  
    function __construct() {  
        parent::__construct();  
        $this->load->model('m_simpro');  
    }  
  
    public function index() {  
        $this->load->view('admin/v_login');  
    }  
  
    function login() {  
        //menangkap data yang dikirim dari form dan menetapkan validasi  
        //formn.  
        $username      = $this->input->post('username');  
        $password      = $this->input->post('password');  
        $this->form_validation->set_rules('username','Username','trim|required');  
        $this->form_validation->set_rules('password','Password','trim|required');  
  
        //cek validasi  
        if ($this->form_validation->run() != false) {  
            $where = array(  
                'username'          => $username,
```

```

'password'          => $password,
 //'user_status'    => $user_status,
 //'group_id'        => $group_id
);

$data  = $this->m_simpro->edit_data($where, 'users');
$d      = $this->m_simpro->edit_data($where, 'users')-
>row();
$cek   = $data->num_rows();
if ($cek > 0) {
    $session = array(
        'id'           => $d->user_id,
        'nama'         => $d->username,
        'group_id'     => $d->group_id,
        'status'       => $d->user_status
    );
    $this->session->set_userdata($session);
    redirect(base_url().'admin');
} else {
    redirect(base_url().'admin_login?pesan=gagal');
}
} else {
    $this->load->view('admin/V_login');
}
}

}

```

2. Buat View Dengan Nama V_login.php

V_login.php

```

<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1, shrink-to-fit=no">
    <meta name="description" content="">

```

```

<meta name="author" content="">

<title>Login</title>

<!-- Custom fonts for this template-->
<link href="<?php echo base_url('assets/vendor/fontawesome-free/css/all.min.css') ?>" rel="stylesheet" type="text/css">

<!-- Page level plugin CSS-->
<link href="<?php echo base_url('assets/vendor/datatables/dataTables.bootstrap4.css') ?>" rel="stylesheet">

<!-- Custom styles for this template-->
<link href="<?php echo base_url('assets/css/sb-admin.css') ?>" rel="stylesheet">

</head>

<body class="bg-dark">

<div class="container">
<div class="card card-login mx-auto mt-5">
<div class="card-header">Login</div>
<div class="text-center">
</div>
<div class="card-body">

<?php
if (isset($_GET['pesan'])) {
    if ($_GET['pesan'] == "gagal") {
        echo "<div class='alert alert-danger'>Login gagal! Username dan Password Salah.</div>";
    } elseif ($_GET['pesan'] == "logout") {
        echo "<div class='alert alert-danger'>Anda Telah logout.</div>";
    } elseif ($_GET['pesan'] == "belumlogin") {
        echo "<div class='alert alert-warning'>Bukan Level Anda. Silahkan Mencoba lagi.</div>";
    } elseif ($_GET['pesan'] == "belumaktif") {

```

```

echo "<div class='alert alert-success'>Akun anda belum aktif,  

silahkan kontak administrator.</div>";
    }
}
?>
<hr/>
<form      method="post"      action="=base_url() . 'admin_login/login' ?">
<br/>
<div class="form-group">
<div class="form-label-group">
    <input type="text" name="username" id="username" class="form-control" placeholder="username" required="required" autofocus="autofocus">
        <label for="username">Username</label>
    </div>
</div>
<div class="form-group">
<div class="form-label-group">
    <input type="password" name="password" id="inputPassword" class="form-control" placeholder="Password" required="required">
        <label for="inputPassword">Password</label>
    </div>
</div>
<div class="form-group">
<div class="checkbox">
    <label>
        <input type="checkbox" value="remember-me">
        Remember Password
    </label>
</div>
</div>
<div class="form-group">
<input type="submit" value="Login" class="btn btn-primary col-xs-12">
</div>
</form>
<div class="text-center">

```

```

<!--class="d-block small mt-3" href="register">Register an
Account?-->
</div>
</div>
</div>

<!-- Bootstrap core JavaScript-->
<script src="vendor/jquery/jquery.min.js"></script>
<script src="vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

<!-- Core plugin JavaScript-->
<script src="vendor/jquery-easing/jquery.easing.min.js"></script>

</body>

</html>

```

- Hasilnya akan menjadi seperti ini

The screenshot shows a login interface. At the top, the word "Login" is displayed. Below it, a yellow box contains the message "Bukan Level Anda. Silahkan Mencoba lagi.". The main form area has two input fields: one for "Username" with the value "Siangga" and another for "Password" with a redacted value. A "Remember Password" checkbox is present, and a blue "Login" button is at the bottom.

6.6 Membuat form Inputan User

- Membuat Controller dengan nama Admin.php dan isi kodingan sebagai berikut :

Admin.php

```
<?php
defined('BASEPATH') OR exit ('No direct script access allowed');

/**
 *
 */
class Admin extends CI_Controller
{

    function __construct() {
        parent::__construct();
        // $this->load->model('m_chain');
        // $this->load->helper('tgl_indo');
        // Cek LOGIN
        if ($this->session->userdata('group_id') != "1") {
            redirect(base_url().'admin_login?pesan=belumlogin');
        } if ($this->session->userdata('status') != 'aktif') {
            redirect(base_url().'admin_login?pesan=belumaktif');
        }
    }

    function index() {
        // parsing data dari database
        // $data['users'] = $this->db->query("select * from users order by user_id asc limit 10")->result();
        // $data['employer'] = $this->db->query("select * from employer order by emp_no asc limit 10")->result();
        // $data['transaksi_sampah'] = $this->db->query("select * from transaksi_sampah order by trs_no desc limit 10")->result();
        $this->load->view('admin/v_header');
        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_index');
        $this->load->view('admin/v_footer');
    }

    function logout() {
        $this->session->sess_destroy();
        redirect(base_url().'admin_login?pesan=logout');
    }
}
```

```

}

function employer() {
    $data['employer'] = $this->m_simpro->get_data('employer')-
>result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_employer',$data);
    $this->load->view('admin/v_footer');
}

function employer_add(){
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_employer_add');
    $this->load->view('admin/v_footer');
}

function employer_add_act(){
    $empno = $this->input->post('empno');
    $fname = $this->input->post('fname');
    $lname = $this->input->post('lname');
    $tl = $this->input->post('tl');
    $tgl_1 = $this->input->post('tgl_1');
    $alamat = $this->input->post('alamat');
    $jk = $this->input->post('jk');
    $hp = $this->input->post('hp');
    $email = $this->input->post('email');
    $this->form_validation->set_rules('empno','Nomor
Karyawan','required');
    $this->form_validation->set_rules('fname','Nama
Depan','required');
    $this->form_validation->set_rules('lname','Nama
Belakang','required');
    $this->form_validation->set_rules('tl','Tempat Lahir','required');
    $this->form_validation->set_rules('tgl_1','Tanggal
Lahir','required');
    $this->form_validation->set_rules('alamat','Alamat
Lengkap','required');
}

```

```

$this->form_validation->set_rules('jk','Jenis Kelamin','required');
$this->form_validation->set_rules('hp','No. HP','required');
$this->form_validation->set_rules('email','Email','required');

if ($this->form_validation->run() != false) {
    $data = array(
        'emp_no' => $empno,
        'f_name' => $fname,
        'l_name' => $lname,
        't_l' => $tl,
        'tgl_l' => $tgl_l,
        'alamat' => $alamat,
        'j_k' => $jk,
        'hp' => $hp,
        'email' => $email
    );
}

$this->m_simpro->insert_data($data, 'employer');
redirect(base_url().'admin/employer');
} else {
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_employer_add');
    $this->load->view('admin/v_footer');
}
}

function employer_edit($id) {
    $where = array(
        'emp_no' => $id
    );
    $data['employer'] = $this->m_simpro->edit_data($where, 'employer')->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_employer_edit',$data);
    $this->load->view('admin/v_footer');
}

function employer_update() {

```

```

$empno    = $this->input->post('empno');
$fname     = $this->input->post('fname');
$lname     = $this->input->post('lname');
$tl        = $this->input->post('tl');
$tgl_1     = $this->input->post('tgl_1');
$alamat    = $this->input->post('alamat');
$jk        = $this->input->post('jk');
$hp        = $this->input->post('hp');
$email     = $this->input->post('email');
$this->form_validation->set_rules('empno','Nomor
Karyawan','required');
$this->form_validation->set_rules('fname','Nama
Depan','required');
$this->form_validation->set_rules('lname','Nama
Belakang','required');
$this->form_validation->set_rules('tl','Tempat Lahir','required');
$this->form_validation->set_rules('tgl_1','Tanggal
Lahir','required');
$this->form_validation->set_rules('alamat','Alamat
Lengkap','required');
$this->form_validation->set_rules('jk','Jenis Kelamin','required');
$this->form_validation->set_rules('hp','No. HP','required');
$this->form_validation->set_rules('email','Email','required');

if ($this->form_validation->run() != false) {
$where = array(
'emp_no' => $empno
);
$data = array(
'f_name' => $fname,
'l_name' => $lname,
'tl' => $tl,
'tgl_1' => $tgl_1,
'alamat' => $alamat,
'j_k' => $jk,
'hp' => $hp,
'email' => $email
);
}

$this->m_simpro->update_data($where,$data,'employer');

```

```

redirect(base_url().'admin/employer');

} else {
    $where = array(
        'emp_no' => $id
    );
    $data['employer'] = $this->m_simpro->edit_data($where,
        'employer')->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_employer_edit',$data);
    $this->load->view('admin/v_footer');
}
}

/* AKHIR ZONA CRUD EMPLOYER */

/* ZONA CRUD USERS */
function users_apr() {
    $data['users'] = $this->db->query("select * from pengguna where
        user_status='nonaktif")->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_users_apr',$data);
    $this->load->view('admin/v_footer');
}

function users_apr_edit($id) {
    $where = array(
        'user_id' => $id
    );
    $data['users'] = $this->m_simpro->edit_data($where, 'users')-
        >result();
    $data['role'] = $this->m_simpro->get_data('groups')->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_users_apr_edit',$data);
    $this->load->view('admin/v_footer');
}

function users_apr_update() {

```

```

$id      = $this->input->post('id');
$empno   = $this->input->post('empno');
$username = $this->input->post('username');
$status   = $this->input->post('status');
$this->form_validation->set_rules('empno','Nomor
Karyawan','required');
$this->form_validation-
>set_rules('username','Username','required');
$this->form_validation->set_rules('status', 'Status User', 'required');

if ($this->form_validation->run() != false) {
$where = array(
    'user_id' => $id
);
$data = array(
    'emp_no'    => $empno,
    'username'  => $username,
    'user_status' => $status
);
$this->m_simpro->update_data($where,$data,'users');
redirect(base_url().'admin/users_apr');
} else {
$where = array(
    'user_id' => $id
);
$data['users'] = $this->m_simpro->edit_data($where, 'users')-
>result();
$this->load->view('admin/v_header');
$this->load->view('admin/v_sidebar');
$this->load->view('admin/v_users_apr_edit',$data);
$this->load->view('admin/v_footer');
}
}

function users() {
$data['users'] = $this->m_simpro->get_data('pengguna')-
>result();
$this->load->view('admin/v_header');
$this->load->view('admin/v_sidebar');
$this->load->view('admin/v_users',$data);
}

```

```

        $this->load->view('admin/v_footer');
    }

function users_tambah(){
    $data['role'] = $this->m_simpro->get_data('groups')->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_users_add',$data);
    $this->load->view('admin/v_footer');
}

function users_add_act(){
    $empno    = $this->input->post('empno');
    $username = $this->input->post('username');
    $userpass = $this->input->post('userpass');
    $usergroup = $this->input->post('usergroup');
    $email    = $this->input->post('email');
    $this->form_validation->set_rules('empno','Nomor
Karyawan','required');
    $this->form_validation-
>set_rules('username','Username','required');
    $this->form_validation-
>set_rules('userpass','Password','required');
    $this->form_validation->set_rules('usergroup','Role','required');
    $this->form_validation->set_rules('email','E-Mail','required');

    if ($this->form_validation->run() != false) {
        $data = array(
            'emp_no'    => $empno,
            'username'  => $username,
            'password'  => $userpass,
            'group_id'  => $usergroup,
            'user_status' => 'nonaktif'
        );

        $this->m_simpro->insert_data($data, 'users');
        $pesan = '<center><font color="green"><strong>-----';
        $pesan .= STATUS . ' ' . PENAMBAHAN . ' ' . USERS . ' -----';
        $pesan .= '</strong></font></center>';
    }
}

```

```

<center><font color="black"><br><br> Kepada Yth.
Bapak/Ibu : '$email.
'<br>Username Anda : '$username.
'<br>Password Anda : '$userpass.
'<br>Silahkan tunggu beberapa saat untuk pengaktifan akun anda'.
'<br>Silahkan Login menggunakan username dan password anda'.
'<br>Kami sistem produksi dan terimakasih banyak'.

'</center><br><br></font><center><font
color="green"><strong>----- SIMPRO -----</strong></font></center>';
// Konfigurasi email

$config1 = [
'mailtype' => 'html',
'charset' => 'utf-8',
'protocol' => 'smtp',
'smtp_host' => 'smtp.gmail.com',
'smtp_user' => 'internship1wildan@gmail.com', // Email gmail
'smtp_pass' => 'npm1164058', // Password gmail
'smtp_crypto' => 'ssl',
'smtp_port' => 465,
'crlf'
=> "\r\n",
'newline' => "\r\n"
];
// Load
library email dan konfigurasinya

```

```

    $this->load-
>library('email', $config1);

        // Email dan
        nama pengirim
        $this->email-
>from('internship1wildan@gmail.com', 'SIMPRO NOTIFICATION
(No-Reply)');

        // Email
        penerima
        $this->email-
>to($email); // Ganti dengan email tujuan

        // Subject
        email
        $this->email-
>subject('SIMPRO NOTIFICATION (No-Reply)');

        // Isi email
        $this->email-
>message($pesan);

        // Tampilkan
        pesan sukses atau error
        if      ($this-
>email->send()) {
            echo
            'Sukses! email berhasil dikirim.';

        } else {
            echo
            'Error! email tidak dapat dikirim.';
        }

        $this->session->set_flashdata('flash',
        'Berhasil ditambahkan');
        redirect(base_url().'admin/users');
    } else {
        $data['role'] = $this->m_simpro->get_data('groups')->result();
    }
}

```

```

        $this->load->view('admin/v_header');
        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_users_add',$data);
        $this->load->view('admin/v_footer');
    }
}

function users_edit($id) {
    $where = array(
        'user_id' => $id
    );
    $data['users'] = $this->m_simpro->edit_data($where, 'users')->result();
    $data['role'] = $this->m_simpro->get_data('groups')->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_users_edit',$data);
    $this->load->view('admin/v_footer');
}

function users_update() {
    $id      = $this->input->post('id');
    $empno   = $this->input->post('empno');
    $username = $this->input->post('username');
    $userpass = $this->input->post('userpass');
    $usergroup = $this->input->post('usergroup');
    $status   = $this->input->post('status');
    $this->form_validation->set_rules('empno','Nomor
Karyawan','required');
    $this->form_validation-
>set_rules('username','Username','required');
    $this->form_validation-
>set_rules('userpass','Password','required');
    $this->form_validation->set_rules('usergroup','Role','required');
    $this->form_validation->set_rules('status','Status User','required');

    if ($this->form_validation->run() != false) {
        $where = array(
            'user_id' => $id
        );
    }
}

```

```

$data = array(
    'emp_no' => $empno,
    'username' => $username,
    'password' => $userpass,
    'group_id' => $usergroup,
    'user_status' => $status
);

$this->m_simpro->update_data($where,$data,'users');
redirect(base_url().'admin/users');

} else {
    $where = array(
        'user_id' => $id
    );
    $data['users'] = $this->m_simpro->edit_data($where, 'users')-
>result();
    $data['role'] = $this->m_simpro->get_data('groups')->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_users_edit',$data);
    $this->load->view('admin/v_footer');
}
}

function users_hapus($id) {
    $where = array (
        'user_id' => $id
    );

    $this->m_simpro->delete_data($where,'users');
    redirect(base_url().'admin/users');
}

/* AKHIR ZONA CRUD USERS */

/* ZONA CRUD PENGAJUAN */
function pengajuan() {
    $data['simpro'] = $this->db->query("select * from simpro, witel,
categories where
simpro.witel_id=witel.witel_id and
simpro.cat_id=categories.cat_id")->result();

```

```

        $this->load->view('admin/v_header');
        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_pengajuan',$data);
        $this->load->view('admin/v_footer');
    }

function pengajuan_add() {
    $data['witel']    = $this->m_simpro->get_data('witel')->result();
    $data['categories'] = $this->m_simpro->get_data('categories')-
>result();
    $data['no_simpro'] = $this->m_simpro->get_no_pengajuan();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_pengajuan_add',$data);
    $this->load->view('admin/v_footer');
}

function pengajuan_add_act() {
    $id    = $this->input->post('id');
    $witel = $this->input->post('witel');
    $noted = $this->input->post('noted');
    $date1 = $this->input->post('date1');
    $long  = $this->input->post('long');
    $lat   = $this->input->post('lat');
    $alamat = $this->input->post('alamat');
    $unsc  = $this->input->post('unsc');
    $qtyodp = $this->input->post('qtyodp');
    $this->form_validation->set_rules('witel','Wilayah
Telekomunikasi','required');
    $this->form_validation->set_rules('noted','Noted','required');
    $this->form_validation->set_rules('date1','Daerah
Telekomunikasi','required');
    $this->form_validation->set_rules('long','Longitude','required');
    $this->form_validation->set_rules('lat','Latitude','required');
    $this->form_validation->set_rules('alamat','Alamat
Lengkap','required');
    $this->form_validation->set_rules('unsc','UNSC','required');
    $this->form_validation->set_rules('qtyodp','QTY
ODP','required');
}

```

```

if ($this->form_validation->run() != false) {
    $data = array(
        'simpro_id' => $id,
        'tgl'      => date('Y-m-d'),
        'witel_id'  => $witel,
        'datel'     => $datel,
        'cat_id'    => $noted,
        'longitude'=> $long,
        'latitude'  => $lat,
        'alamat'   => $alamat,
        'unsc'      => $unsc,
        'qty_odp'   => $qtyodp,
        'status'    => 'pending'
    );
}

$this->m_simpro->insert_data($data, 'simpro');
redirect(base_url().'admin/pengajuan');
} else {
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_pengajuan_add');
    $this->load->view('admin/v_footer');
}
}

function pengajuan_edit($id) {
    $where = array(
        'simpro_id' => $id
    );
    $data['simpro'] = $this->m_simpro->edit_data($where, 'simpro')-
>result();
    $data['simpro'] = $this->db->query("select * from simpro, witel,
categories where
simpro.witel_id=witel.witel_id      and
simpro.cat_id=categories.cat_id and simpro_id='".$id."")->result();
    $data['witel'] = $this->m_simpro->get_data('witel')->result();
    $data['categories'] = $this->m_simpro->get_data('categories')-
>result();
// $data['no_simpro'] = $this->m_simpro->get_no_pengajuan();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
}

```

```

        $this->load->view('admin/v_pengajuan_edit',$data);
        $this->load->view('admin/v_footer');
    }

function pengajuan_update() {
    $id    = $this->input->post('id');
    $witel = $this->input->post('witel');
    $noted = $this->input->post('noted');
    $datel = $this->input->post('datel');
    $long   = $this->input->post('long');
    $lat    = $this->input->post('lat');
    $alamat = $this->input->post('alamat');
    $unsc   = $this->input->post('unsc');
    $qtyodp = $this->input->post('qtyodp');
    $status = $this->input->post('status');
    $this->form_validation->set_rules('witel','Wilayah
Telekomunikasi','required');
    $this->form_validation->set_rules('noted','Noted','required');
    $this->form_validation->set_rules('datel','Daerah
Telekomunikasi','required');
    $this->form_validation->set_rules('long','Longitude','required');
    $this->form_validation->set_rules('lat','Latitude','required');
    $this->form_validation->set_rules('alamat','Alamat
Lengkap','required');
    $this->form_validation->set_rules('unsc','UNSC','required');
    $this->form_validation->set_rules('qtyodp','QTY ODP','required');
    $this->form_validation->set_rules('status','Status Pengajuan',
'required');

if ($this->form_validation->run() != false) {
    $where = array(
        'simpro_id' => $id
    );
    $data = array(
        'witel_id'  => $witel,
        'datel'     => $datel,
        'cat_id'    => $noted,
        'longitude'=> $long,
        'latitude'  => $lat,
        'alamat'    => $alamat,
}

```

```

        'unsc'    => $unsc,
        'qty_odp' => $qtyodp,
        'status'   => $status
    );
$this->m_simpro->update_data($where,$data,'simpro');
redirect(base_url().'admin/pengajuan');
} else {
    $where = array(
        'simpro_id' => $id
    );
    $data['simpro']      = $this->m_simpro->edit_data($where,
'simpro')->result();
    $data['simpro']      = $this->db->query("select * from simpro, witel,
categories where
                                simpro.witel_id=witel.witel_id      and
simpro.cat_id=categories.cat_id and simpro_id='".$id."'");
->result();
    $data['witel']       = $this->m_simpro->get_data('witel')->result();
    $data['categories'] = $this->m_simpro->get_data('categories')-
>result();
//$data['no_simpro'] = $this->m_simpro->get_no_pengajuan();
$this->load->view('admin/v_header');
$this->load->view('admin/v_sidebar');
$this->load->view('admin/v_pengajuan_edit',$data);
$this->load->view('admin/v_footer');
}
}

function pengajuan_hapus($id) {
    $where = array (
        'simpro_id' => $id
    );

    $this->m_simpro->delete_data($where,'simpro');
    redirect(base_url().'admin/pengajuan');
}

/* AKHIR ZONA CRUD PENGAJUAN */

/* AWAL ZONA CRUD KATEGORI / WITEL */
function kategori(){

```

```

$data['kategori'] = $this->m_simpro->get_data('categories')-
>result();
$this->load->view('admin/v_header');
$this->load->view('admin/v_sidebar');
$this->load->view('admin/v_kategori',$data);
$this->load->view('admin/v_footer');
}

function kategori_add(){
$this->load->view('admin/v_header');
$this->load->view('admin/v_sidebar');
$this->load->view('admin/v_kategori_add');
$this->load->view('admin/v_footer');
}

function kategori_add_act(){
$kn = $this->input->post('kname');
$kk = $this->input->post('kket');
$this->form_validation->set_rules('kname','Nama
Kategori','required');

if ($this->form_validation->run() != false) {
$data = array(
'cat_name' => $kn,
'cat_ket' => $kk
);
$this->m_simpro->insert_data($data, 'categories');
redirect(base_url().'admin/kategori');
} else {
$this->load->view('admin/v_header');
$this->load->view('admin/v_sidebar');
$this->load->view('admin/v_kategori_add');
$this->load->view('admin/v_footer');
}
}

function kategori_edit($id) {
$where = array(
'cat_id' => $id
);

```

```

        $data['kategori']      =      $this->m_simpro->edit_data($where,
'categories')->result();
        $this->load->view('admin/v_header');
        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_kategori_edit',$data);
        $this->load->view('admin/v_footer');
    }

function kategori_update() {
    $id    = $this->input->post('id');
    $kn    = $this->input->post('kname');
    $kk    = $this->input->post('kket');
    $this->form_validation->set_rules('kname','Nama
Kategori','required');

    if ($this->form_validation->run() != false) {
        $where = array(
            'cat_id' => $id
        );
        $data = array(
            'cat_name' => $kn,
            'cat_ket'  => $kk
        );
        $this->m_simpro->update_data($where,$data,'categories');
        redirect(base_url().'admin/kategori');

    } else {
        $where = array(
            'cat_id' => $id
        );
        $data['kategori']      =      $this->m_simpro->edit_data($where,
'categories')->result();
        $this->load->view('admin/v_header');
        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_kategori_edit',$data);
        $this->load->view('admin/v_footer');
    }
}

```

```

function kategori_delete($id) {
    $where = array (
        'cat_id' => $id
    );
    $this->m_simpro->delete_data($where,'categories');
    redirect(base_url().'admin/kategori');
}
/* AKHIR ZONA CRUD KATEGORI / NOTED */

/* ZONA CRUD WITEL */
function witel(){
    $data['witel'] = $this->m_simpro->get_data('witel')->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_witel',$data);
    $this->load->view('admin/v_footer');
}

function witel_add(){
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_witel_add');
    $this->load->view('admin/v_footer');
}

function witel_add_act(){
    $wn      = $this->input->post('witelname');
    $this->form_validation->set_rules('witelname','Nama
Witel','required');

    if ($this->form_validation->run() != false) {
        $data = array(
            'witel_name' => $wn
        );
        $this->m_simpro->insert_data($data, 'witel');
        redirect(base_url().'admin/witel');
    } else {
        $this->load->view('admin/v_header');
        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_witel_add');
    }
}

```

```

        $this->load->view('admin/v_footer');
    }
}

function witel_edit($id) {
    $where = array(
        'witel_id' => $id
    );
    $data['witel'] = $this->m_simpro->edit_data($where, 'witel')-
>result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_witel_edit',$data);
    $this->load->view('admin/v_footer');
}

function witel_update() {
    $id = $this->input->post('witelid');
    $wn = $this->input->post('witelname');
    $this->form_validation->set_rules('witelname','Nama
Witel','required');

    if ($this->form_validation->run() != false) {
        $where = array(
            'witel_id' => $id
        );
        $data = array(
            'witel_name' => $wn
        );

        $this->m_simpro->update_data($where,$data,'witel');
        redirect(base_url().'admin/witel');

    } else {
        $where = array(
            'witel_id' => $id
        );
        $data['witel'] = $this->m_simpro->edit_data($where, 'witel')-
>result();
        $this->load->view('admin/v_header');
    }
}

```

```

        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_witel_edit',$data);
        $this->load->view('admin/v_footer');
    }
}

function witel_delete($id) {
    $where = array (
        'witel_id' => $id
    );
    $this->m_simpro->delete_data($where,'witel');
    redirect(base_url().'admin/witel');
}
/* AKHIR ZONA WITEL */

/* ZONA LAPORAN */
function laporan_rekap() {
    $data['witel'] = $this->m_simpro->get_data('witel')->result();
    $this->load->view('admin/v_header');
    $this->load->view('admin/v_sidebar');
    $this->load->view('admin/v_laporan_filter',$data);
    $this->load->view('admin/v_footer');
}

function laporan_rekap_result() {
    $dari = $this->input->post('dari');
    $sampai = $this->input->post('sampai');
    $witel = $this->input->post('witel');
    $this->form_validation->set_rules('dari','Dari Tanggal', 'required');
    $this->form_validation->set_rules('sampai','Sampai Tanggal',
    'required');
    $this->form_validation->set_rules('witel','Wilayah
Telekomunikasi', 'required');

    if($this->form_validation->run() != false) {
        $data['laporan'] = $this->db->query("select * from laporan
        where date(tgl) BETWEEN '$dari' AND
        '$sampai' AND witel_id='$witel'")->result();
        $data['witel'] = $this->m_simpro->get_data('witel')->result();
        $this->load->view('admin/v_header');
    }
}

```

```

        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_laporan_filter_result',$data);
        $this->load->view('admin/v_footer');
    } else {
        $data['witel'] = $this->m_simpro->get_data('witel')->result();
        $this->load->view('admin/v_header');
        $this->load->view('admin/v_sidebar');
        $this->load->view('admin/v_laporan_filter',$data);
        $this->load->view('admin/v_footer');
    }
}

function laporan_rekap_print() {
    $dari = $this->input->get('dari');
    $sampai = $this->input->get('sampai');
    $witel = $this->input->get('witel');

    if ($dari != "" && $sampai != "") {
        $data['laporan'] = $this->db->query("select * from laporan
                                                where date(tgl) BETWEEN '$dari' AND
                                                '$sampai' AND witel_id='$witel'")->result();
        $data['witel'] = $this->m_simpro->get_data('witel')->result();
        $this->load->view('admin/v_laporan_print',$data);
    } else {
        redirect('admin/laporan_rekap_result',$data);
    }
}

function laporan_rekap_pdf() {
    $nama_dokumen='laporan_rekapitulasi_pengajuan'; //Beri nama
file PDF hasil.
    $kertas = array(
        'mode' => 'utf-8',
        'format' => 'A4-L',
        'orientation' => 'L'); //Mengatur mode,ukuran kertas dan orientasi
    $mpdf = new \Mpdf\Mpdf($kertas);
    $dari = $this->input->get('dari');
    $sampai = $this->input->get('sampai');
    $witel = $this->input->get('witel');
}

```

```

        $data['laporan'] = $this->db->query("select * from laporan
                                                where date(tgl) BETWEEN '$dari' AND
                                                '$sampai' AND witel_id='$witel'")->result();
        $data2 = $this->load->view('admin/v_laporan_pdf',$data, TRUE);
        $mpdf->WriteHTML($data2);
        $mpdf->Output($nama_dokumen.".pdf" , T);
    }
    /* AHIR ZONA LAPORAN */
}
?>

```

2. Membuat View dengan nama V_user_add.php

V_user_add.php

```

<div class="container">
    <div class="card card-input mx-auto mt-5">
        <div class="card-header bg-danger text-white"><p style="text-align:center">TAMBAH PENGGUNA </div></p>
        <div class="card-body">
            <div class="dropdown">
                <form action="=php echo base_url().'admin/users_add_act'" method="post"&gt;
                    &lt;div class="form-group"&gt;
                        &lt;label&gt;No. Karyawan&lt;/label&gt;
                        &lt;input type="number" name="empno" class="form-control"&gt;
                    &lt;/div&gt;

                    &lt;div class="form-group"&gt;
                        &lt;label&gt;Email&lt;/label&gt;
                        &lt;input type="email" name="email" class="form-control"&gt;
                    &lt;/div&gt;

                    &lt;div class="form-group"&gt;
                        &lt;label&gt;Username&lt;/label&gt;
                        &lt;input type="text" name="username" class="form-control"&gt;
                    &lt;/div&gt;
                &lt;/form&gt;
            &lt;/div&gt;
        &lt;/div&gt;
    &lt;/div&gt;
&lt;/div&gt;
</pre

```

```

<div class="form-group">
    <label>Password</label>
    <input type="password" name="userpass" class="form-control">
</div>

<div class="form-group">
    <label>Hak Akses</label>
    <select name="usergroup" class="form-control">
        <option value="">Pilih Role</option>
        <?php foreach($role as $r){ ?>
        <option value="<?php echo $r->group_id; ?>"><?php echo ucwords($r->group_name); ?></option>
        <?php } ?>
    </select>
    <?php echo form_error('usergroup'); ?>
</div>

<!--
<div class="form-group">
    <label>Status</label>
    <select name="userssts" class="form-control">
        <option value="aktif">Aktif</option>
        <option value="nonaktif">Tidak Aktif</option>
    </select>
</div>
-->
<div class="form-group">
    <input type="submit" value="Simpan" class="btn btn-primary">
</div>
<br/><br/>
</form>
</div>
</div>
</div>
</div>

```

3. Hasilnya akan seperti Ini

TAMBAH PENGGUNA

No. Karyawan	<input type="text"/>
Email	<input type="text"/>
Username	<input type="text" value="siangga"/>
Password	<input type="password" value="*****"/>
Hak Akses	<input type="text" value="-Pilih Role-"/>
<input type="button" value="Simpan"/>	

TUTORIAL PEMBUATAN SISTEM INFORMASI DENGAN MUDAH DAN MENYENANGKAN MENGGUNAKAN FRAMEWORK CODEIGNITER

Buku ini berisi mengenai tutorial pembuatan sistem informasi dengan mudah dan menyenangkan men-ggunakan Framework Codeigniter. Buku ini ditunjukkan untuk berbagai kalangan yang ingin belajar membuat sebuah sistem dengan mudah dan tidak membosankan .

Pada Buku ini menjelaskan beberapa hal mulai dari Pengenalan web , Pengenalan bahasa Pemrograman , Instalasi alat / tools yang digunakan ,Pengenalan dan Pembuatan Basis Data . serta Penjelasan mengenai Framework Codeigniter. Kemudian sebelum membuat sebuah sistem informasi terdapat contoh pembuatan sebuah sistem yang dibuat secara runtut , serta adanya kodingan-kodingan agar pembaca bisa lebih mudah dan paham dalam membuat sistem informasi yang pembaca inginkan.

ISBN 978-602-53897-8-8

