

COVER

Masih dalam progress

Kurnia Sandi

www.thegadebasis.xyz

*“ Ilmu itu bukan yang di hafal, tetapi
yang memberi manfaat” .*

- Imam Syafi'i -

TENTANG PENULIS



Kurnia Sandi, seorang mahasiswa program studi Diploma-IV Teknik Informatika yang *falling in love* pada bidang *web development* dan *database engineer*. Memulai petualangan dalam dunia pemrograman pada tahun 2016 (semester 1). Lebih senang dengan dunia pemrograman daripada bidang IT lainnya. Senang berbagi pengetahuan baru terkait pemrograman khususnya *web development* dan teknologi tentang perintah-perintah pada *database* (DDL dan DML).

Memiliki *skill* pemrograman khususnya *Hypertext Pre-Processor* (PHP) dan Javascript untuk *front-end* dan *back-end* serta *skill* Bahasa pemrograman *Structure Query Language* (SQL) guna manajemen arsitektur *database* dalam hal *Data Definition Language* (DDL), *Data Controll Language* (DML) dan *Data Controll Language* (DCL).

Saat ini *doi* masih menempuh Pendidikan semester 7, selama proses Pendidikan semasa kuliah penulis sudah mengerjakan 3 *project* dalam kampus yaitu Manajemen Keuangan Berbasis Web, Toko Buku *Online* Chocobooks dan Bank Sampah Istimewa versi 1.0 serta beberapa *project* lainnya di luar kampus. Untuk mengetahui informasi lebih lanjut tentang penulis, silahkan email ke kurniya097@gmail.com.

PENGANTAR

Dengan nama Allah SWT yang Maha Pengasih lagi Maha Panyayang, Penulis panjatkan puja dan puji syukur atas kehadiran-Nya, yang telah melimpahkan rahmat, taufiq, serta hidayah-Nya kepada penulis, sehingga penulis dapat menyelesaikan buku “**Membuat *Website Powerfull and Safety* dengan mengimplementasikan *function, stored procedure, view* dan *opertor* MySQL Pada *Framework Codeigniter* (Studi Kasus Bank Sampah Tabungan Emas)**” ini. Penulis menyadari sepenuhnya masih ada kekurangan baik dari segi susunan kalimat maupun tata bahasa. Oleh karena itu, dengan tangan terbuka penulis menerima segala saran dan kritik dari pembaca agar menjadi lebih baik untuk ke depannya. Akhir kata penulis berharap buku ini dapat memberikan manfaat maupun inspirasi terhadap pembaca.

DAFTAR ISI

BAB I

PENGENALAN BAHASA PEMROGRAMAN PHP

Pada Bab ini, pembaca akan mempelajari tentang Bahasa pemrograman PHP yang *outputnya* dapat meahami terkait pengertian PHP, sejarah, cara kerja, kelebihan serta kekurangannya.

1.1 Pengertian PHP

Menurut Sibero (2012:49), “PHP (*Personal Home Page*) adalah pemrograman (interpreter) yang melakukan proses penerjemahan baris sumber menjadi kode mesin yang dimengerti oleh komputer secara langsung saat baris kode dijalankan”. [1]

Menurut Kustiyaningsih (2011:114), “PHP (atau resminya PHP: *Hypertext Pre-Processor*) adalah skrip yang bersifat *server-side* yang ditambahkan ke dalam HTML. PHP sendiri merupakan singkatan dari *Personal Home Page Tools*. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML sehingga suatu halaman web tidak lagi bersifat statis, namun menjadi bersifat dinamis. Sifat *server-side* berarti pengerjaan kode program dilakukan di *server*, baru kemudian hasilnya akan dikirim ke *browser*”. [2]

Berdasarkan pengertian di atas, dapat ditarik kesimpulan bahwa PHP (PHP: *Hypertext Pre-Processor*) adalah suatu Bahasa pemrograman bersifat *server side* yang digunakan untuk menerjemahkan sejumlah baris kode (bisa ditambahkan ke dalam HTML) program menjadi kode mesin dimengerti oleh mesin komputer.

1.2 Sejarah PHP

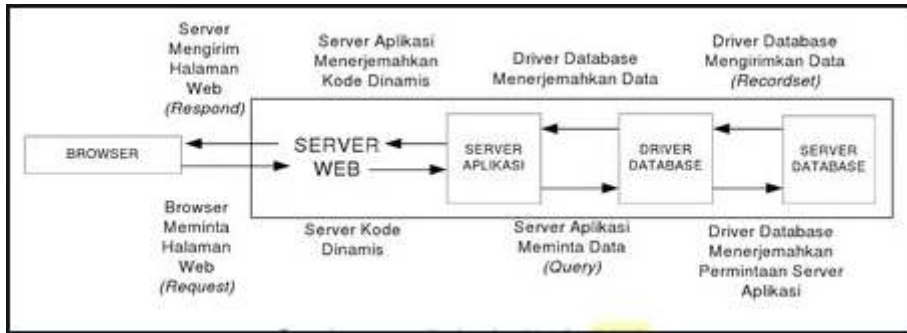
Pada mulanya PHP merupakan singkatan dari *Persnoal Home Page* (Situs Personal). PHP sendiri pertama kali ditemukan oleh Rasmus Lerdorf pada tahun 1995. Saat itu PHP masih bernama *Form Interpreted* (FI), yang bentuknya masih berupa sekumpulan *script* dan digunakan untuk mengolah data *form* dari web.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik dan lebih cepat. Kemudian (Juni 1998), perusahaan tersebut merilis interpreter baru untuk Bahasa pemrograman PHP dan meresmikan *relade version* sebagai PHP 3.0 dan singkatan PHP berubah menjadi *Hypertext Preprocessing*.

PHP difokuskan pada *scripting server-side*. *Scripting server-side* 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. Artinya, anda dapat melakukan apa yang bisa dilakukan CGI dengan menggunakan PHP seperti mengambil data inputan *form*, meng-*generate* konten halaman dinamis, mengirim dan menerima *cookies* serta masih banyak lagi. Kemampuan dan dukungannya untuk *database* juga sangat bisa diandalkan.

1.3 Prinsip Kerja PHP

Bahasa pemrograman PHP merupakan Bahasa pemrograman yang dikategorikan dalam kelompok *Server Side Programming*, yang artinya Bahasa pemrograman ini memerlukan penerjemah dalam hal ini *web server* untuk menjalankannya. Adapun penjelasan tentang cara kerja Bahasa pemrograman PHP digambarkan pada *Gambar 1. 1*.



Gambar 1. 1 Prinsip Kerja PHP

Berdasarkan gambaran prinsip kerja PHP pada *Gambar 1. 1* menurut Supono dan Viridiandry P. (2016:4), dapat dijelaskan sebagai berikut:

1. Client/user mengirimkan file PHP (menggunakan browser) melalui Web Server (Seperti Internet Explorer, Mozilla Firefox, Google Chrome, dll).
2. Web Server mendapatkan request atau permintaan dari user lalu meneruskan ke server melalui jaringan internet.
3. Web Server lalu meneruskan permintaan file PHP tersebut ke PHP processor dapat berupa modul (bagian dari web-server) atau terpisah (sebagai CGI/Fast - CGI).
4. Permintaan diproses oleh PHP dan diteruskan ke database (jika terdapat permintaan ke database), kemudian hasilnya dikirim kembali ke web-server.
5. Web Server memaket kembali hasil tersebut dengan menambahkan HTTP header dan dikirim kembali ke browser melalui jaringan internet.
6. Browser memproses HTTP paket dan menampilkannya kembali pada user sebagai file HTML.

1.4 Kelebihan PHP

Bahasa pemrograman PHP merupakan Bahasa pemrograman yang paling banyak digunakan, tentu karena berbagai alasan, salah satunya adalah mempunyai beberapa kelebihan dibandingkan dengan Bahasa pemrograman lainnya yang sejenis. Berikut ini kelebihan Bahasa pemrograman PHP: [3]

1. PHP adalah bahasa multiplatform yang artinya dapat berjalan di berbagai mesin dan sistem operasi (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah sistem lainnya.
2. PHP bersifat open source yang berarti dapat digunakan oleh siapa saja secara gratis.
3. Web server yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, nginx, hingga Xitami dengan konfigurasi yang relatif mudah dan tidak berbelit-belit, bahkan banyak yang membuat dalam bentuk paket atau package (PHP, MySQL dan Web Server).
4. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis. Komunitas dan developer yang siap membantu dalam pengembangan.
5. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
6. Banyak bertebaran Aplikasi dan Program PHP yang Gratis dan Siap pakai seperti WordPress, PestaShop, dan lain-lain.
7. Dapat mendukung banyak database, seperti MySQL, Oracle, MS-SQL, dan seterusnya.

1.5 Kekurangan PHP

Seperti layaknya manusia, Bahasa pemrograman PHP juga memiliki kekurangan dari sekian banyak kelebihan yang telah disebutkan sebelumnya. Berikut ini kekurangan Bahasa pemrograman web PHP yang mungkin dapat dijadikan sebagai bahan pertimbangan dalam memilih Bahasa pemrograman temuan Rasmus Lerdorf ini. [3]

1. PHP tidak mengenal *Package*
2. Jika tidak di-*encoding*, maka kode PHP dapat dibaca semua orang dan untuk meng-*encoding*-nya dibutuhkan tool dari Zend yang mahal sekali biayanya.
3. PHP memiliki kelemahan keamanan. Jadi programmer harus jeli dan berhati-hati dalam melakukan pemrograman dan konfigurasi PHP.

BAB II

PAKET WEB-SERVER

2.1 Pendahuluan

Paket *Web-Server* adalah *server* yang berupa *software* yang digunakan untuk menerima permintaan dalam bentuk situs *web* melalui HTTP atau HTTPS dari klien. [3]

Berikut ini beberapa *web-server* yang banyak dijumpai dan banyak digunakan oleh para *developer* dalam pengembangan *website*.

1. XAMPP terdiri Apache web server, MySQL, PHP, Perl, FTP Server dan PHPMyAdmin. Apache sendiri dapat diinstall di berbagai sistem operasi Linux, Solaris, Windows dan Mac OS X.
2. WampServer terdiri dari Apache, PHP5 dan MySQL. Wamp hanya mendukung system operasi Windows. WampServer dilengkapi dengan *manage service*, dengan tray icon yang memudahkan dalam mengelola server.
3. EasyPHP, paket ini terdiri dari Apache server, MySQL dan PHPMyAdmin.
4. PHPTriad terdiri dari Apache, MySQL, PHP dan PHPMyAdmin. Web server ini hanya dapat digunakan di sistem operasi Windows.
5. FoxServ terdiri dari Apache, MySQL, PHP, PEAR, Zend dengan versi yang terbaru. Web server ini mendukung system operasi Windows dan Linux.
6. PHPDev terdiri dari PHP, Apache, MySQL, PERL, phpMyAdmin. Phpdev secara kontinyu memperbaharui versi terbaru dari semua paket yang ada.
7. AppServ terdiri dari Apache, MySQL, PHP, dan PHPMyAdmin dengan semua komponen yang lengkap. AppServ mendukung system operasi Windows dan Linux.

8. Server2go terdiri dari Apache, PHP dan MySQL. Server2go dapat secara mandiri sebagai server diinstall dalam CD-ROM, USB Flash-Disk. Ukuran file sekitar 6-45 MB tergantung paket yang ingin digunakan.
9. Apache2Triad terdiri dari Apache2, MySQL, PostgreSQL, OpenSSL, Xmail, SlimFTPd, PHP, Perl dan Pytho + Apache2TriadCP, PHPmyadmin, PHPpgAdmin, AWStats, UebiMiau, PHPXmail, PHPSFTPd. Semua isi dalam paket Ap.ache2Triad tersebut dalam versi stabil
10. VertigoServ terdiri dari PHP development dan server environment untuk Windows secara komplit, juga Apache 2.x.x, PHP 5.x.x, MySQL 5.x.x, dan PHPMyAdmin.
11. Uniform Server terdiri dari paket terakhir dari Apache2, Perl5, PHP5, MySQL5, dan phpMyAdmin. Web server ini dapat dijalankan di system operasi Windows tanpa, tanpa diinstalasi, cukup di-*unpack* dan jalan.
12. MAMP terdiri dari Apache, PHP dan MySQL. MAMP hanya digunakan untuk system operasi OSX.
13. TYPO3. Terdiri dari Apache Webserver, PHP dan MySQL. System Operasi yang didukung Windows.

2.2 Web-Server

Berdasarkan penjelasan pada *point 2.1*, bahwa untuk menjalankan file program yang ditulis menggunakan Bahasa pemrograman PHP dibutuhkan sebuah *web-server*. Adapun paket *web-server* yang digunakan pada buku ini adalah XAMPP versi terbaru. Anda dapat mengunduh *software* tersebut di situs resminya secara gratis (<https://www.apachefriends.org/index.html>).

2.3 XAMPP



Xampp merupakan paket perangkat lunak (*software*) yang tersedia secara gratis sehingga bebas untuk digunakan tanpa perlu menggunakan *licence* dari pengembang *software*. XAMPP berfungsi sebagai server yang berdiri sendiri (*localhost*) yang terdiri dari Apache HTTP Server, MySQL database dan penerjemah Bahasa yang ditulis dengan Bahasa pemrograman PHP dan Perl. XAMPP saat ini dikembangkan oleh perusahaan apache friends yang biasanya digunakan untuk simulasi pengembangan *website*.

Tool pengembangan web ini mendukung teknologi web populer seperti PHP, MySQL dan Perl. Dengan menggunakan perangkat lunak XAMPP, programmer web dapat menguji aplikasi web yang dikembangkan kemudian mempresentasikannya ke pihak lain secara langsung dari komputer, tanpa harus terkoneksi ke internet. XAMPP juga dibekali dengan fitur manajemen *database* yaitu PHPMyAdmin seperti pada server *hosting* sungguhan, sehingga pengembang *web* dapat membagikan aplikasi *web* berbasis *database* dengan mudah.

Istilah XAMPP diambil dari kata “X” yang berarti empat system operasi (Windows, Linux, MacOS dan Solaris), sedangkan “A” diambil dari kata Apache, kemudian “M” merupakan singkatan dari kata MySQL, kemudian huruf “P” kepedekan dari PHP dan huruf “P” yang terakhir adalah akronim dari kata Perl.

Apache adalah aplikasi web server yang bertugas menampilkan halaman *web* yang benar kepada *user* berdasarkan kode HTML, PHP atau yang lainnya.

MySQL adalah aplikasi database server. SQL merupakan kependekan dari *Structured Query Language*. SQL sendiri merupakan Bahasa terstruktur yang digunakan untuk mengelola database. MySQL dapat digunakan untuk membuat serta mengelola *database* beserta isinya (DDL dan DML). *Programmer* atau *User* dapat memanfaatkan MySQL untuk proses *Create*, *Read*, *Update* dan *Delete* (atau sering disingkat sebagai CRUD) pada data yang berada dalam *database*.

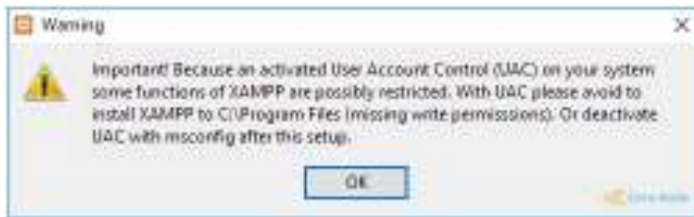
PHP merupakan kependekan dari *Personal Home Page* (Berubah menjadi *Hypertext Preprocessing* pada tahun 1998 [baca: 1.1 Sejarah PHP]). Bahasa pemrograman PHP digunakan untuk membuat *aplikasi* berbasis *web* yang bersifat *server-side scripting*. PHP memungkinkan untuk membuat halaman *web* yang bersifat dinamis. System manajemen basis data yang sering digunakan Bersama PHP adalah MySQL namun PHP juga mendukung system manajemen *database* atau disebut juga *database management system* (DBMS) seperti Oracle, Microsoft Access, Interbase, d-base, PostgreSQL, dan sebagainya. [4]

2.4 Instalasi Web-Server XAMPP

Setelah berkenalan dengan tool pengembang web yang populer ini, saatnya kita melakukan instalasi perangkat lunak ini ke dalam system komputer kita. Pada pembahasan buku ini, penulis menggunakan system operasi Windows versi 10 64-bit. untuk system operasi lain silahkan mencari sumber lainnya baik dari buku maupun internet. Baiklah, tanpa basa-basi lagi, berikut adalah langkah-langkah instalasi atau pemasangan XAMPP:

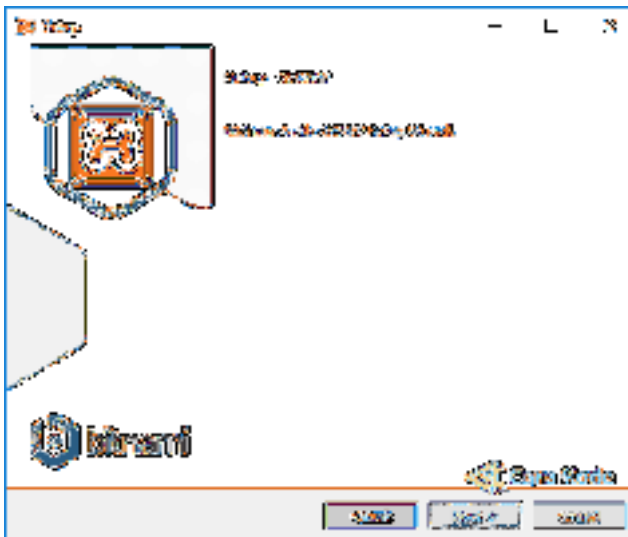
1. Download program file master *web-server* XAMPP di situs resminya (<https://www.apachefriends.org/download.html>).
2. Jalankan setup.exe XAMPP.

3. Jika muncul *warning UAC* seperti gambar di bawah ini, tidak perlu dihiraukan, klik OK.



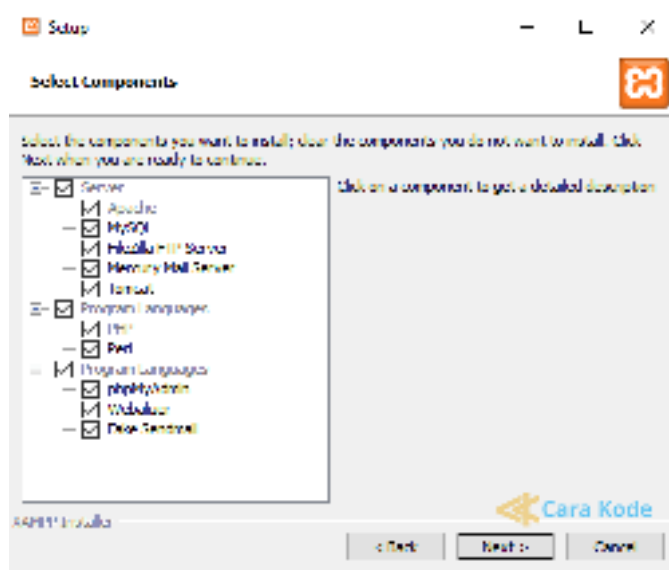
Gambar 2. 1 Install XAMPP - *Warning*

4. Setelah itu akan muncul gambar *Welcome Screen XAMPP installer*.



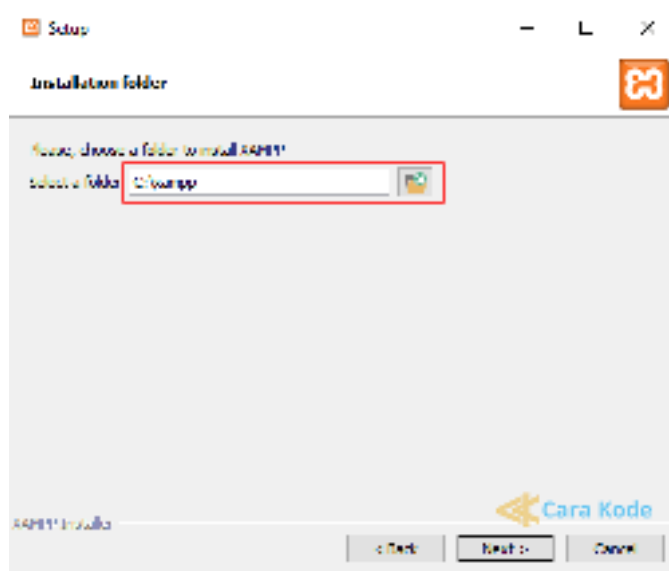
Gambar 2. 2 Install XAMPP – *Welcome Screen*

- Pilih komponen yang akan diinstal. Lalu klik Next.



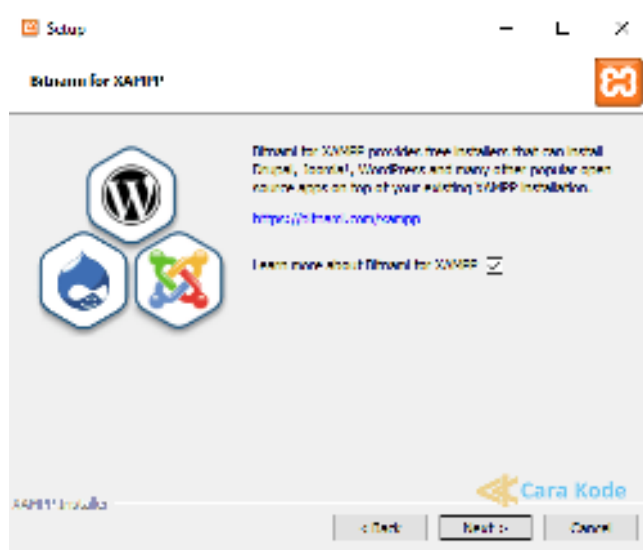
Gambar 2. 3 Install XAMPP – *Select Component*

- Pilih *destination folder* (default `c:\xampp\`). Fungsi pemilihan ini adalah untuk menentukan di mana nanti folder *root* disimpan. Lalu klik Next.



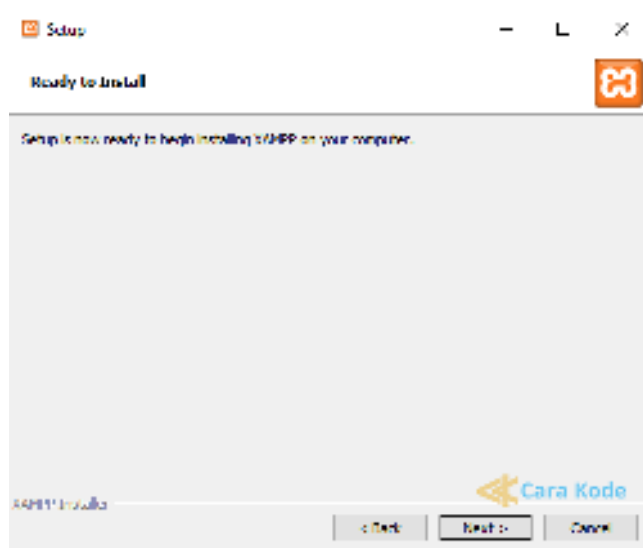
Gambar 2. 4 Install XAMPP – *Select Destination Folder*

7. Proses ini kita bebas untuk *check* atau *uncheck* Learn more about Bitnami for XAMPP. Jika kita memilih opsi *check* maka setelah proses instalasi selesai kita akan diarahkan ke halaman Bitnami dari XAMPP itu sendiri. Jika di *uncheck* maka saat instalasi selesai system tidak akan mengalihkan ke halaman Bitnami.



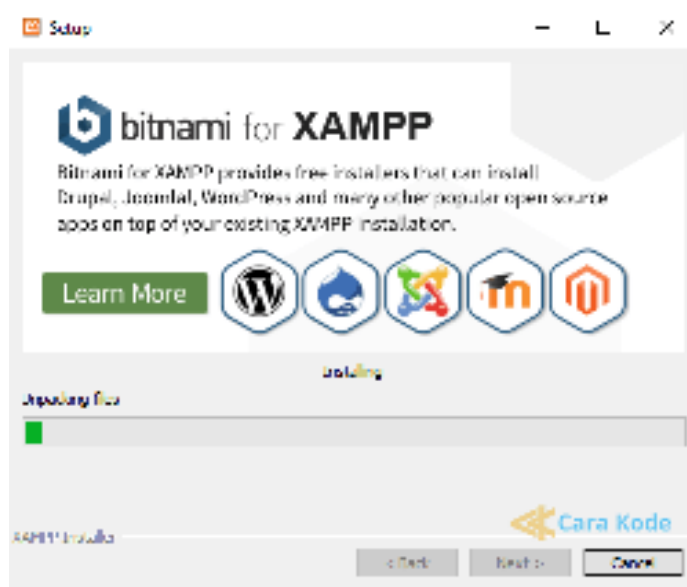
Gambar 2. 5 Install XAMPP – *Select Option*

8. Setelah itu, akan tampil XAMPP ready to Install, klik Next.



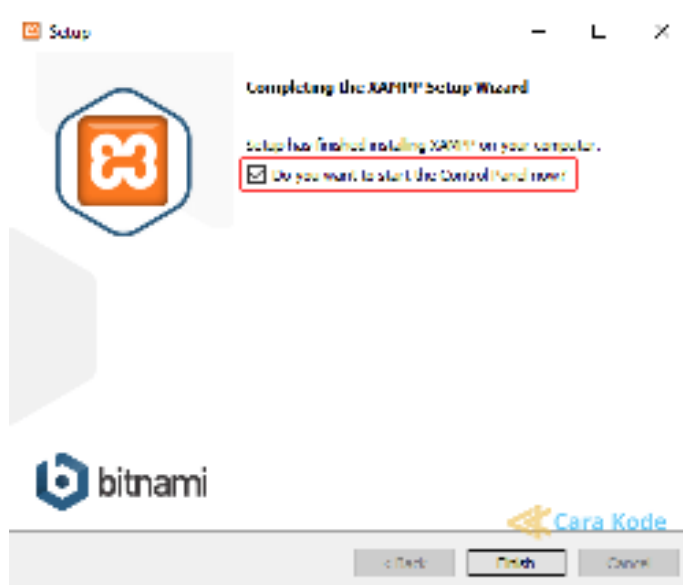
Gambar 2. 6 Install XAMPP – *Ready to Install*

9. Proses instalasi XAMPP. Silahkan ditunggu hingga proses selesai.



Gambar 2. 7 Install XAMPP – *Installing XAMPP*

10. Proses instalasi selesai. *Checklist* “Do you want to start the Control Panel now?” jika ingin menampilkan control panel xampp. Lalu pilih Finish.



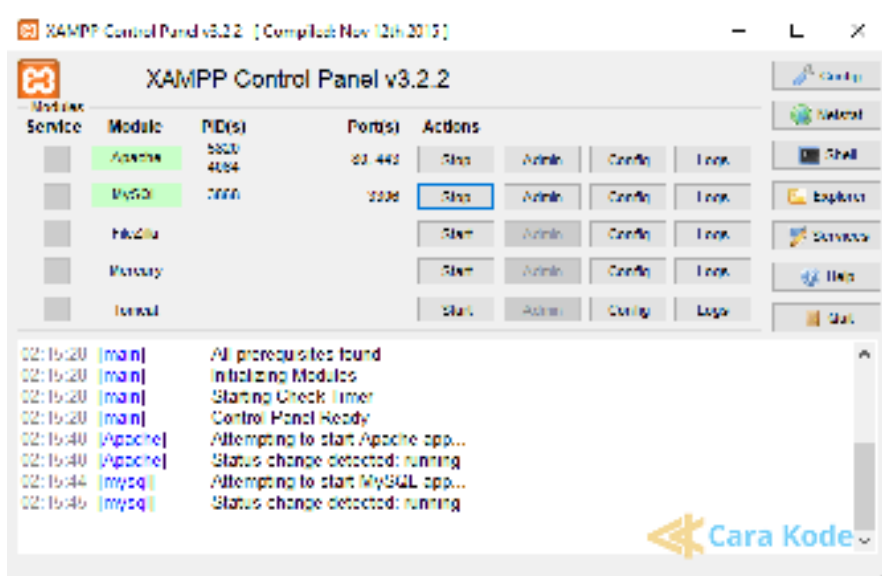
Gambar 2. 8 Install XAMPP – Instalasi selesai

11. Jika muncul kotak dialog untuk memilih Bahasa, maka silahkan pilih Bahasa yang diinginkan (Pada pembahasan ini, penulis memilih Bahasa Inggris), kemudian klik Save.



Gambar 2. 9 Install XAMPP – Pilih bahasa

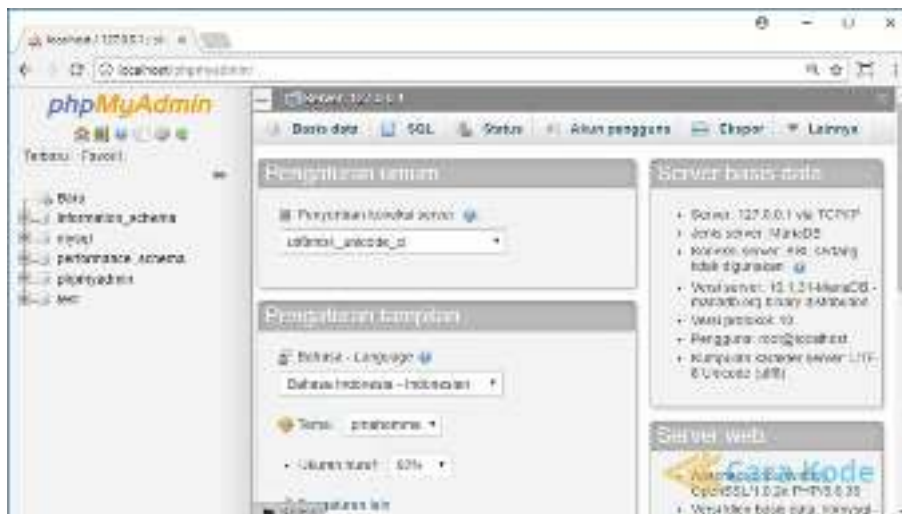
12. Setelah semua langkah dilakukan maka Xampp Control Panel akan langsung terbuka karena sebelumnya memilih *checklist Checklist* “Do you want to start the Control Panel now?” yang artinya setelah proses instalasi sukses maka secara otomatis Control Panel dari XAMPP akan langsung terbuka.



Gambar 2. 10 Install XAMPP – Control Panel

Silakan klik “start” pada *module* Apache dan MySQL.

13. Untuk melakukan *testing* silahkan buka *browser* favorit. Kemudian pada *address bar* masukkan url “<http://localhost/phpmyadmin/>”.



Gambar 2. 11 Install XAMPP – PHPMyAdmin

Jika pada *browser* anda telah tampil halaman PHPMyAdmin seperti pada Gambar 2. 11, maka *web-server* XAMPP telah berhasil terinstall.

BAB III

TEXT EDITOR: ATOM

Bagi developer, memilih text editor sebagai aplikasi untuk menulis koding sangatlah penting, karena hal tersebut dapat membantu dalam penulisan program dengan lebih cepat. Namun kebanyakan *developer*, jikalau sudah terbiasa menggunakan satu text editor, malas untuk mencoba yang lain, padahal semakin banyak kita mencoba, semakin tahu mana text editor yang paling baik. [5]

3.1 Kenapa Atom?

Editor seperti *Sublime* dan *TextMate* menawarkan kenyamanan tetapi memiliki ekstensibilitas terbatas. Di tempat lain seperti spektrum, Emacs dan Vim menawarkan fleksibilitas yang lebih *ekstream*, tetapi mereka tidak terlalu mudah didekati dan hanya dapat dikustomisasi dengan bahasa scripting tujuan yang khusus.

Kami pikir kami bisa melakukan yang lebih baik. Tujuan kami adalah kombinasi tanpa kompromi dari peretasan dan kegunaan: editor yang akan menyambut siswa sekolah dasar pada hari pertama mereka belajar kode, tetapi juga alat yang tidak akan mereka kalahkan saat mereka berkembang menjadi peretas berpengalaman.

Karena kami telah menggunakan Atom untuk membangun Atom, apa yang dimulai sebagai percobaan secara bertahap telah matang menjadi alat yang tidak bisa kita hidupi tanpa. Di permukaan, Atom adalah editor teks desktop modern yang Anda harapkan. Pasang tudung, dan Anda akan menemukan sistem yang meminta untuk diretas. [6]

3.1.1 Inti Atom

Web bukan tanpa kesalahannya, tetapi pengembangan selama dua dekade telah membentuknya menjadi platform yang luar biasa lunak dan kuat. Jadi ketika kami mulai menulis editor teks yang kami ingin

memperluas, teknologi web adalah pilihan yang jelas. Tapi pertamanya, kita harus membebaskannya dari rantai. [6]

3.1.2 Web Asli

Browser web bagus untuk menjelajahi halaman web, tetapi menulis kode adalah kegiatan khusus yang membutuhkan alat khusus. Lebih penting lagi, browser sangat membatasi akses ke sistem lokal untuk alasan keamanan, dan bagi kami, editor teks yang tidak bisa menulis file atau menjalankan subprocesses lokal adalah non-starter.

Karena alasan ini, kami tidak membangun Atom sebagai aplikasi web tradisional. Alih-alih, Atom adalah varian khusus Chromium yang dirancang untuk menjadi editor teks dan bukan browser web. Setiap jendela Atom pada dasarnya adalah halaman web yang dibuat secara lokal. [6]

Semua API yang tersedia untuk aplikasi Node.js juga tersedia untuk kode yang berjalan dalam konteks JavaScript setiap jendela. Hibrida ini memberikan pengalaman pengembangan sisi klien yang unik.

Karena semuanya lokal, Anda tidak perlu khawatir tentang pipa aset, penggabungan skrip, dan definisi modul asinkron. Jika Anda ingin memuat beberapa kode, cukup minta di bagian atas file Anda. Sistem modul Node memudahkan untuk memecah sistem menjadi banyak paket kecil yang fokus. [6]

3.1.3 Javascript, Penemuan C++

Berinteraksi dengan kode asli juga sangat sederhana. Sebagai contoh, kami menulis pembungkus di sekitar mesin ekspresi reguler Oniguruma untuk dukungan tata bahasa TextMate kami. Di browser, itu akan membutuhkan petualangan dengan NaCl atau Esprima. Integrasi node membuatnya mudah.

Selain Node API, kami juga mengekspos API untuk dialog asli, menambahkan item menu aplikasi dan konteks, memanipulasi dimensi jendela, dll. [6]

3.1.4 Web Tech: Bagian Menyenangkan

Manfaat besar lainnya, yang dilengkapi dengan penulisan kode untuk Atom, adalah jaminan bahwa itu berjalan pada versi terbaru Chromium. Itu berarti kita dapat mengabaikan masalah seperti kompatibilitas browser dan polyfill. Kita dapat menggunakan semua fitur web yang mengkilap di masa depan, hari ini.

Misalnya, tata letak ruang kerja dan panel kami didasarkan pada flexbox. Ini adalah standar yang muncul dan telah melalui banyak perubahan sejak kami mulai menggunakannya, tetapi tidak ada yang penting selama itu berhasil.

Dengan seluruh industri mendorong teknologi web ke depan, kami yakin bahwa kami sedang membangun Atom di tanah subur. Teknologi UI asli datang dan pergi, tetapi web adalah standar yang menjadi lebih mampu dan ada di mana-mana setiap tahun. Kami senang menggali lebih dalam kotak alatnya. [6]

3.1.5 Editor Teks Terbuka

Kami melihat Atom sebagai pelengkap sempurna bagi misi utama GitHub untuk membangun perangkat lunak yang lebih baik dengan bekerja bersama. Atom adalah investasi jangka panjang, dan GitHub akan terus mendukung pengembangannya dengan tim yang berdedikasi kedepannya. Tetapi kami juga tahu bahwa kami tidak dapat mencapai visi kami untuk Atom saja. Seperti yang telah ditunjukkan oleh Emacs dan Vim selama tiga dekade terakhir, jika Anda ingin membangun komunitas yang berkembang dan bertahan lama di sekitar editor teks, itu harus *open source*.

Seluruh editor Atom adalah sumber bebas dan terbuka dan tersedia di bawah organisasi <https://github.com/atom>. [6]

3.2 Instalasi Atom

Sebelumnya, Atom Editor merupakan editor khusus Sistem Operasi MacOS. Namun sekarang, bagi pengguna system operasi Windows sudah dapat menggunakan Editor ini dikarenakan pengembang telah menyediakan versi Windows namun hanya untuk versi 7 64 bit ke atas. Jadi, bagi pengguna Windows dibawah windows 7 64 bit silahkan menggunakan Text Editor alternatif seperti *Sublime Text*, *Visual Studio Code*, Notepad++, dan sebagainya yang menurut anda nyaman digunakan.

Baiklah, tanpa basa-basi lagi, berikut adalah langkah-langkah instalasi atau pemasangan Text Editor Atom pada Sistem Operasi Windows versi 10:

1. Download file master .exe di situs resminya (<https://atom.io/>).



Gambar 3. 1 Install Atom – Download Atom Editor

2. Jalankan file hasil download (biasanya bernama AtomSetup-x64.exe).

Name	Date modified	Type	Size
AtomSetup-x64	1/17/2020 1:45 AM	Application	183,403 KB
Scratch Desktop Setup 3.6.0	1/17/2020 9:05 PM	Application	175,701 KB

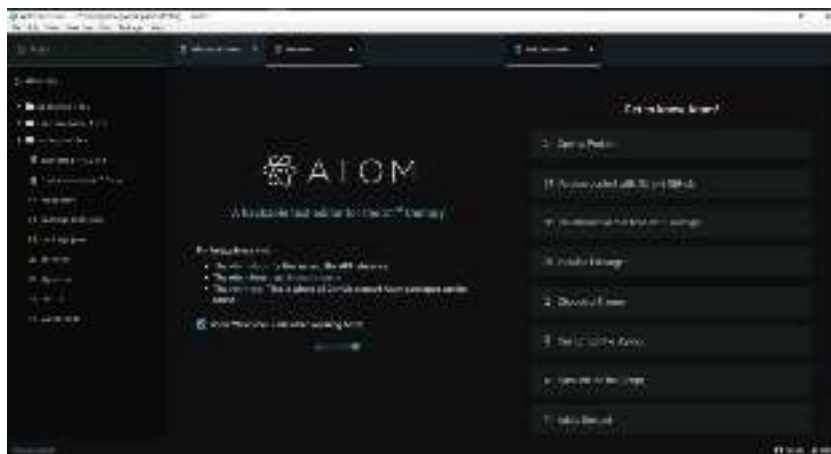
Gambar 3. 2 Install Atom – File Hasil Download Atom Editor

- ### 3. Proses instalasi Atom Editor



Gambar 3. 3 Install Atom – Instalasi Atom Editor

- Setelah proses instalasi selesai dan berhasil tanpa *error* maka Aplikasi Text Editor Atom akan langsung terbuka secara otomatis.



Gambar 3. 4 Install Atom – Instalasi selesai

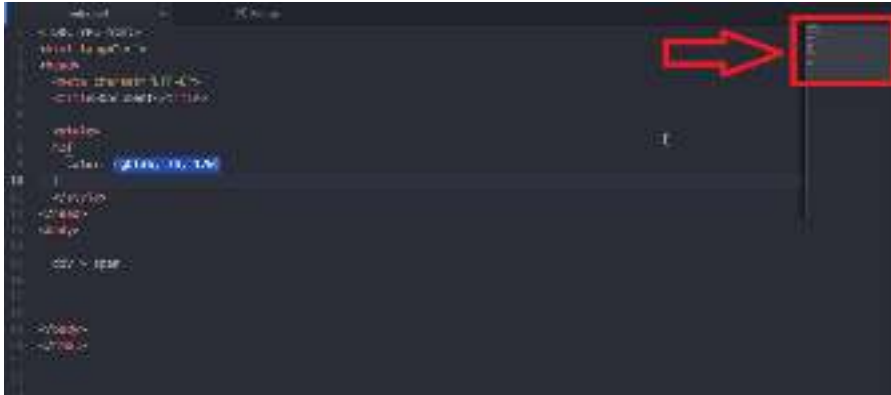
3.3 Package-Package Yang Diperlukan

Ketika bekerja dengan Text Editor Atom, *package* atau paket adalah sesuatu yang diperlukan oleh *developer* dalam menulis kodenya. Adapun beberapa *package* yang cukup diperlukan dan penulis menggunakannya juga, berikut *list package* yang digunakan:

1. Autocomplete-paths
Memunculkan nama file yang tersimpan dalam project ketika hendak dipanggil saat dibutuhkan.
2. Color-picker
Berguna saat menulis kode CSS untuk mempercantik tampilan web. Dengan menekan tombol CTRL+Shift+C maka akan muncul menu warna dan warna hexa. Tinggal dipilih maka akan langsung terinput di kode editor.
3. Pigments
Masih tentang warna. *Package* ini berfungsi untuk menampilkan warna apa yang sedang aktif atau sedang digunakan pada code CSS.
4. Emmet
Salah satu *package* yang powefull dalam atom untuk auto complete kode yang digunakan. Untuk lebih jelasnya silahkan kunjungi situs resminya (<http://emmet.io>).
5. Javascript-snippets
Package ini berguna ketika sedang menulis kode javascript atau Node.js di project.
6. Language-blade
Bagi pengguna Laravel dalam membuat aplikasi berbasis web maka *package* ini sangat disarankan. Karena akan mnampilkan sintaks-sintaks blade dapat ter-*highlight*.

7. Minimap

Package yang berguna untuk menampilkan jendela kecil yang tampilan kode yang telah kita buat.



Gambar 3. 5 Package - Minimap

8. Terminal-plus

Package ini digunakan untuk menampilkan *terminal* atau *command-prompt*. Bagi pengguna Windows *package* ini disarankan untuk dipasang pada editor atom.

BAB IV

BASISDATA (*DATABASE*) DAN DATABASE MANAGEMENT SYSTEM (DBMS): MySQL

Pada Bab ini, pembaca diharapkan dapat memahami konsep tentang apa itu basis data (*database*), kapan pertamakali muncul, siapa pencetus pertamakali, apa manfaatnya, dan apa saja *software* atau DBMS-nya, serta bagaimana penggunaannya.

4.1 Pengertian Basis Data (*Database*) dan Basis Data (*Database*) Relasional

Secara bahasa *database* terdiri atas dua kata yaitu “*data*” yang berarti data atau representasi dunia nyata yang belum memiliki makna serta memiliki wujud berupa huruf, angka, gambar, *audio* dan *video*. Sedangkan “*base*” atau basis adalah gudang atau tempat berkumpulnya data.

Secara umum, *database* berarti koleksi data yang saling terkait atau terintegrasi. Semua manusia dan organisasi di dunia membutuhkan informasi. Seorang tukang koran menyimpan alamat pelanggan, rekening koran, tanggal pembayaran, agen koran, dan lainnnya. Secara praktis, basis data (*database*) dapat dianggap sebagai suatu penyusunan data yang terstruktur yang disimpan dalam media pingingat (*hard disk*) yang tujuannya adalah agar data tersebut dapat diakses dengan mudah dan cepat.

Suatu data yang disimpan dalam media *database* memiliki ketentuan, beberapa ketentuan tersebut seperti harus memiliki *primary key*. *Primary key* adalah kunci utama suatu data yang bersifat *unique* sehingga dapat dibedakan antara baris data yang satu dengan baris data yang lain. Contoh penerapan *primary key* dalam kehidupan ialah menggunakan Nomor Induk Kependudukan (NIK) yang hanya dimiliki oleh satu orang satu NIK.

Jadi, basis data (*database*) adalah himpunan kelompok data (arsip) yang saling berhubungan dan tanpa pengulangan (*redudansi*) serta disimpan dalam

suatu media elektronik kemudia diorganiasi sedemikian rupa agar kelak dapat dimanfaatkan dengan cepat dan mudah.

Sesungguhnya ada beberapa macam *database*, antara lain yaitu *database* hierarkis, *database* jaringan dan *database* relasional. *Database* relasional merupakan *database* yang populer saat ini dan telah diterapkan pada berbagai *platform*, dari PC hingga minikomputer.

Sebuah *database* relasional tersusun atas sejumlah table. Sebagai contoh, *database* akademis mencakup table-tabel seperti table dosen, table mahasiswa, table KRS, table nilai, table ruangan, dan lain-lain. Basis data tentang Bank Sampah bisa mencakup info pribadi seperi nama, jenis kelamin, tanggal lahir dan sebagainya serta rekening yang telah didaftarkan. Gambar 4. 1 menampilkan sample data mengenai *database* bank sampah.

rek_id	rek_tgl	rek	rek_bar
123456/890123111	2019-12-28 09:11:35	123456/890123111	Aleef
123456/890123321	2019-12-29 05:14:15	123456/890123321	Aleef
123456/890123456	2019-12-28 06:58:30	123456/890123456	Aleef
1332619670000566	2019-12-28 08:55:21	1332619670000566	Aleef
1273075907950000	2020-01-07 04:49:27	1273075907950000	Aleef

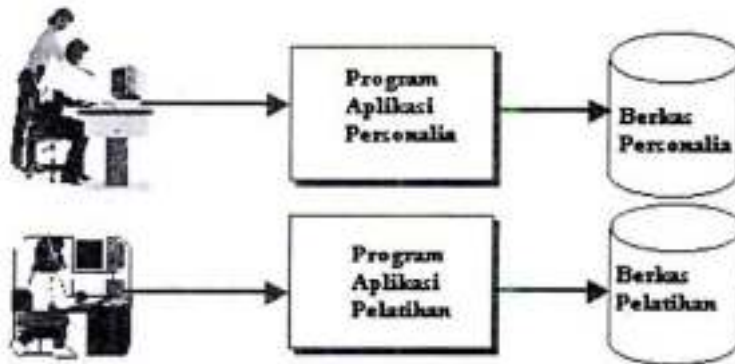
Gambar 4. 1 Gambaran table, baris dan kolom.

4.2 Sejarah Kemunculan Basis Data (*Database*)

Menurut sejarah, system pemroses basis data terbentuk setelah masa system pemrosesan manual dan system pemrosesan berkas.

System pemrosesan manual (berbasi berkas) merupakan bentuk pemrosesan yang menggunakan dasar berupa setumpuk rekaman yang disimpan pada rak-rak berkas. Jika sesuatu berkas diperlukan, berkas tersebut harus dicari pada rak-rak tersebut. Bentuk seperti ini masih banyak dijumpai dalam kehidupan sehari-hari. Contoh lain adalah buku telepon saku dimana seseorang relative mudah mencari nama-nama rekannya karena datanya telah

disusun secara alfabetis. Namun demikian kemudahan seperti ini tetap saja kurang luwes. Data tidak bisa diurutkan menurut kata atau nomor telepon. Jika hal seperti ini dikehendaki, tidak ada cara lain selain menuliskannya kembali. Tentu saja hal itu tidak praktis.



Gambar 4. 2 Gambaran system pemrosesan berkas: masing-masing aplikasi memiliki berkas tersendiri.

Pada saat awal penerapan system komputer, sekelompok rekaman disimpan pada sejumlah berkas secara terpisah. System yang menggunakan pendekatan seperti ini biasa disebut sebagai **system pemrosesan berkas**. System ini tentu saja memiliki kelebihan daripada system pemrosesan manual, yaitu dalam hal kecepatan dan keakuratannya. Kelemahannya, perancang system masih didasarkan pada kebutuhan individual pengguna, bukan kebutuhan sejumlah pengguna. Setiap kali ada kebutuhan baru dari seorang pengguna, kebutuhan segera diterjemahkan ke program komputer. Alhasil, setiap program aplikasi menuliskan data sendiri. Sementara itu ada kemungkinan data yang sama juga terdapat pada berkas-berkas lain yang digunakan oleh program aplikasi lainnya.

Konkretnya, system pemrosesan berkas memiliki kekurangan dalam beberapa hal, adapun kekurangannya sebagai berikut:

- Data mubazir,
- Berbagi data terbatas,
- Integritas terhadap data kurang dan tidak konsisten, serta
- Tidak luwes.

Data mubazir atau data yang teriduplikasi atau duplikasi data pada praktiknya disebabkan oleh setiap program aplikasi menggunakan data tersendiri, sebagaimana telah diutarakan sebelumnya. Sebagai contoh, dalam sebuah perusahaan terdapat bagian Personalia dan bagian Pelatihan. Program aplikasi personalia dipakai untuk mengarsipkan data para pegawai untuk kepentingan personalia semata sedangkan program aplikasi pelatihan dipakai untuk mencatat kepentingan yang menyangkut pelatihan. Data yang terdapat pada kedua program aplikasi tersebut (nomor pegawai, nama pegawai, dan bagian tempat kerja pegawai) akan dicatat. Hal seperti inilah yang memungkinkan adanya duplikasi data.

Data yang mubazir sebenarnya terjadi karena keterbatasan berbagi data. Seandainya suatu data dapat dipakai oleh beberapa program aplikasi, ataupun sejumlah orang maka kemubaziran terhadap data dapat dihindari.

Penyebab **tidak konsisten** suatu data diakibatkan oleh adanya perubahan terhadap data yang sama, tetapi tidak semuanya diubah. Sebagai contoh, bila seseorang telah pindah bagian dan data tentang hal ini telah dicatat oleh bagian Personalia, tetapi tidak dicatat oleh bagian Pelatihan, maka pengguna pada bagian pelatihan tetap dianggap bahwa orang tersebut belum pindah bagian.

Ketidakkonsistenan data berkaitan erat dengan integritas data. Bila data tidak konsisten, maka integritasnya akan berkurang. **Integritas** berarti bahwa

data selalu dalam keadaan valid. Sesuatu perubahan, misalnya data alamat pelanggan, seharusnya tidak menimbulkan kerancuan. Hal ini hanya dapat terjadi apabila data tentang hal tersebut hanya ada pada satu tempat. Pada lingkungan *multiuser*, integritas merupakan hal yang sangat kritis, disebabkan tindakan yang dilakukan oleh masing-masing pengguna akan mempengaruhi pengguna yang lain. Bayangkan apa jadinya dua buah biro perjalanan bisa menjual sebuah tiket untuk tempat duduk yang sama pada pesawat terbang kepada dua pelanggan?

Kekurangluwesannya dan/atau tidak luwesnya suatu sistem pemrosesan berkas terletak pada hal pengembangan atau perubahan. Sebagai gambaran, sekiranya terdapat perubahan struktur pada berkas (misalnya jumlah angka suatu data yang menyatakan uang diperbesar) maka setiap program harus diubah. Hal ini disebabkan setiap program berisi definisi data untuk setiap berkas yang diaksesnya. Sekalipun data yang diubah tidak dipakai oleh suatu program, tetap saja program tersebut harus diubah. Keadaan seperti ini banyak dijumpai pada program-program aplikasi lama yang dibangun dengan menggunakan Bahasa pemrograman COBOL.

Pada basis data terdapat istilah independensi data. Independensi data adalah sifat yang memungkinkan perubahan struktur berkas tidak mempengaruhi program, begitupun sebaliknya. Independensi data juga bisa berarti bahwa data bersifat tidak tergantung pada data lain. Bila suatu data dihapus, data lain harus tidak terpengaruh oleh tindakan tersebut.

Sistem pemrosesan basis data dimaksudkan untuk mengatasi kelemahan-kelemahan yang ada pada sistem pemrosesan berkas. Sistem seperti ini dikenal dengan sebutan DBMS (*Database Management System*). [7]

4.3 Evolusi Teknologi Basis Data (*Database*)

Perkembangan teknologi basis data tidak lepas dari perkembangan perangkat keras dan perangkat lunak. Perkembangan teknologi jaringan komputer dan komunikasi data merupakan salah satu penyumbang kemajuan penerapan basis data, yang kemudian melahirkan system basis data terdistribusi. Dampak perkembangan ini tentu saja dapat dirasakan dalam kehidupan kita, seperti kemudahan dalam penarikan sejumlah uang dengan fasilitas *Automatic Machine Teller* (ATM) yang banyak diterapkan pada perbankan di Indonesia.

Perkembangan pada dunia perangkat lunak (*software*), seperti kecerdasan buatan (*Artificial Intelligence*), system pakar (*expert system*), dan pemrograman berorientasi objek (*object oriented programming*). Namun, tidak hanya itu perkembangan dunia perangkat lunak tersebut juga mempengaruhi perkembangan basis data, sehingga muncul istilah basis data berorientasi objek (*object oriented database*), dan basis data cerdas (*database intelligence*). Tabel 4. 1 memperlihatkan evolusi perkembangan teknologi basis data. [7]

Tabel 4. 1 Evolusi teknologi basis data.

Masa	Perkembangan basis data
1960-an	<ul style="list-style-type: none">• Sistem pemrosesan berkas• DBMS• Layanan informasi secara <i>online</i> berbasis manajemen teks
1970-an	<ul style="list-style-type: none">• Penerapan system pakar pada system pengambilan keputusan• Basis data berorientasi objek

1980-an	<ul style="list-style-type: none"> • Sistem <i>hypertext</i>, yang memungkinkan untuk melihat basis data secara acak menurut kata kunci (sebagaimana yang diterapkan pada internet)
1990-an	<ul style="list-style-type: none"> • Sistem basis data cerdas • System basis data multimedia cerdas.

4.4 Tujuan Basis Data (*Database*)

Tujuan awal dan tujuan utama dari penggunaan basis data (*database*) adalah untuk memberikan keceatan dan kemudahan dalam menemukan kembali arsi/data yang tersimpan dalam media elektroniknya. Selain itu, tujuan secara lengkap dari basis data (*database*) adalah sebagai berikut:

1. Kecepatan dan Kemudahan (*Speed*)

Basi data mendukung kecepatan dan kemudahan dalam mengakses data untuk membantu pekerjaan.

2. Efisiensi Ruang Penyimpanan (*Space*)

Menekan redudansi data dengan adanya relasi antar kelompok data yang saling berhubungan.

3. Keakuratan (*Accuracy*)

Pembuatan relasi antar data yang disertai penerapan aturan tipe data, domain data dan keunikan data yang secara ketat bisa diterapkan dalam basis data (*database*).

4. Ketersediaan (*Availability*)

Data dalam basis data (*database*) dapat dipilah menjadi data utama/master/referensi, data transaksi dan data histori, hingga data kadaluwarsa sesuai dengan kegunaannya.

5. Kelengkapan (*Completeness*)

Untuk mengakomodasi kebutuhan kelengkapan data yang semakin berkembang, maka kita tidak hanya dapat menambah *record-record* data, tetapi juga dapat melakukan perubahan struktur dalam basis data.

6. Keamanan (*Security*)

Penerapan aspek keamanan dapat ditentukan siapa saja yang boleh menggunakan *database* dan menentukan jenis operasi apa saja yang boleh digunakan.

7. Kebersamaan Pemakaian (*Sharability*)

Basis data (*database*) yang dikelola mendukung *multiuser* dan dapat dipakai bersama-sama oleh beberapa system aplikasi pada saat bersamaan.

4.5 Penerapan Basis Data (*Database*)

Basis data merupakan salah satu komponen utama dalam setiap system informasi sehingga banyak diterapkan pada bidang-bidang tertentu. Secara nyata, bidang-bidang yang memanfaatkan basis data antara lain sebagai berikut:

1. Kepegawaian

Dalam hal instansi baik formal maupun non-formal.

2. Pergudangan (*Inventory*)

Untuk perusahaan manufaktur (pabrik), grosir(*reseller*), apotik, dan lain-lain.

3. Reservasi

Untuk hotel, restoran, pesawat, kereta api, dan lain-lain.

4. Layanan Pelanggan (*Customer Care*)

Untuk perusahaan yang berhubungan dengan banyak pelanggan seperti bank, konsultan dan penyedia layanan dan lain-lain.

5. Perbankan dan Koperasi

Dalam pengelolaan data nasabah, data tabungan, data pinjaman, pembuatan laporan-laporan akuntansi, pelayanan informasi pada nasabah atau calon nasabah.

6. Asuransi

Dalam melakukan pengelolaan data nasabah atau data pembayaran premi, pemrosesan pengajuan klaim asuransi, dan lain-lain.

7. Rumah Sakit

Dalam melakukan pengelolaan *history* penyakit, pengelolaan pengobatan pasien, menangani pembayaran perawatan, melayani administrasi pasien dan lain-lain.

8. Pendidikan/Sekolah

Dalam melakukan pengelolaan data siswa/mahasiswa, penjadwalan kegiatan belajar mengajar, melayani pembayaran SPP, pengisian KRS online dan lain-lain.

4.6 Database Management System (DBMS) dan *Structure Query Language* (SQL)

Database Management System (DBMS) adalah *software* atau tools dari basis data (*database*) yang dibangun untuk melakukan pengelolaan *database* dengan operasi-operasi yang telah disediakan menggunakan Bahasa tertentu yang telah disepakati (dalam hal ini adalah SQL). SQL merupakan singkatan dari *Structure Query Language*. Dalam Bahasa Inggris disebut SEQUEL atau ES-KYU-EL [8]. Bahasa ini merupakan standar yang digunakan untuk mengakses *database relasional*.

Standar SQL, mula-mula didefinisikan oleh ISO (*International Standards Organization*) dan ANSI (*the American National Standards Institute*), yang dikenal dengan sebutan SQL86. Namun, seiring berjalannya waktu, sejumlah standar baru telah ditetapkan.

Saat ini banyak perangkat lunak *database* yang menggunakan SQL sebagai perintah untuk mengakses *database*. Beberapa diantaranya sebagai berikut:

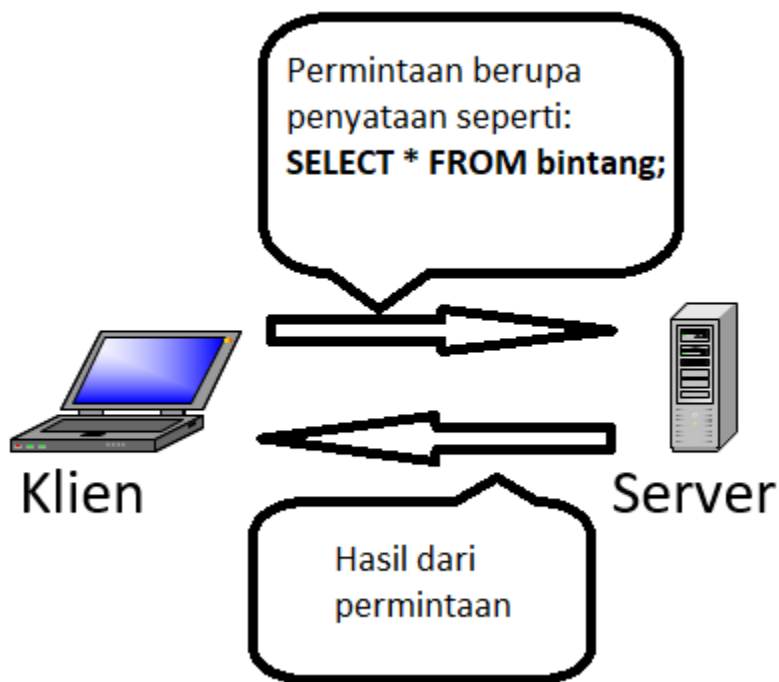
- DB2
- Ingres
- Informix
- ORACLE
- Microsoft Access
- MySQL
- PostgreSQL
- Rdb
- Sybase

Pada praktiknya implementasi SQL sangat bervariasi. Tidak semua fitur SQL didukung oleh vendor perangkat-lunak. Beberapa perintah SQL memiliki perbedaan. Sebagian lagi disebabkan sejumlah fitur memang diperuntukkan di masa mendatang, sehingga belum ada yang mengimplementasikannya. Walaupun begitu, pemahaman terhadap SQL akan mempermudah perpindahan dari suatu *database* ke *database* yang lain karena secara fungsional banyak yang bersifat umum (dapat diterapkan dengan menggunakan perangkat lunak apa pun yang memang mendukungnya).

MySQL sebagai *database* server juga mendukung perintah SQL. Secara khusus, MySQL juga menambahkan sejumlah fungsi yang membuat perintah

SQL pada MySQL sangat variatif. Tentu saja tambahan-tambahan tersebut akan membuat keleluasaan dalam mengakses *database* dan melakukan berbagai tindakan lainnya (misalnya untuk mengambil jam sekarang pada server).

Perintah yang dapat dipahami oleh *database* server MySQL disebut dengan istilah pernyataan [8]. Pernyataan adalah sebuah perintah yang dapat dikerjakan oleh MySQL dengan ciri-ciri diakhiri dengan tanda titik koma (;). Begitu anda mengetikkan titik koma kemudian menekan tombol Enter, program klien MySQL akan segera mengirimkannya ke *database* server MySQL akan segera menanggapi.



Gambar 4. 3 Permintaan dari klien dan tanggapan dari *database* server MySQL

4.7 Pembangunan Basis Data (*Database*) Menggunakan DBMS MySQL

4.7.1 Membuat *Database*

Sebelum mempraktikkan pembuatan table dan berbagai operasi yang terkait dengan table, anda perlu membuat sebuah *database* terlebih dahulu. Sebuah *database* dapat diciptakan dengan menggunakan pernyataan `CREATE DATABASE`. Perintah berikut akan menghasilkan *database* dengan nama **db_inventory**:

```
CREATE DATABASE db_inventory;
```

CATATAN :

Jangan lupa untuk memberikan tanda titik koma (;) di akhir perintah atau pernyataan. Perintah SQL biasa disebut dengan istilah **pernyataan** (*statement*).

Jika anda menjumpai *output* berupa informasi seperti berikut:

```
Query OK, 1 row affected (0.016 sec)
```

Maka *database* **db_inventory** berhasil dibuat.

4.7.2 Memilih *Database*

Sebelum anda bisa mengakses table ataupun hal yang berkaitan dengan *database*, anda perlu mengoneksikan diri ke *database* bersangkutan. Dalam hal ini MySQL *Client* untuk mengakses ke *database* yang hendak dipilih menggunakan perintah:

```
USE db_inventory;
```

4.7.3 Membentuk Tabel

Setelah *database* tercipta dan pengguna memilih *database* terkait (*db_inventory*), maka anda bisa segera mempraktikkan pembuatan table. Dalam hal ini, anda bisa menggunakan pernyataan **CREATE TABLE**. Kaidah pernyataan ini sebagai berikut:

```
CREATE TABLE nama_tabel (  
  Nama_kolom_1 tipe_data ([ukuran]),  
  Nama_kolom_2 tipe_data ([ukuran]),  
  ...,  
  Nama_kolom_n tipe_data ([ukuran]) );
```

Contoh berikut digunakan untuk menciptakan table dengan nama **tbl_barang**:

```
CREATE TABLE tbl_barang (  
  id          VARCHAR (5) NOT NULL PRIMARY KEY,  
  nama        VARCHAR(25),  
  jumlah      INTEGER,  
  ket         TEXT(100) );
```

Pada contoh table *tbl_barang* terdiri atas 4 buah kolom (*field*) dengan rincian sebagai berikut:

- Kolom (*field*) **id** dengan tipe data **VARCHAR** (untuk menampung string) dengan Panjang (*length*) maksimal 5 karakter. Digunakan untuk menyatakan id barang. Dalam hal ini, kolom ini digunakan sebagai kunci primer (dinyatakan dengan **PRIMARY KEY**) dan kolom ini isinya tidak boleh kosong (**NOT NULL**).
- Kolom (*field*) **nama** dengan tipe data **VARCHAR** dengan panjang (*length*) maksimal 25 karakter.
- Kolom (*field*) **jumlah** dengan tipe data **INTEGER** (untuk menampung bilangan bulat) dengan panjang (*length*) *default* yaitu

maksimal 11 karakter (karena kita tidak menginisiasikan berapa maksimal *length* yang dibutuhkan).

- Kolom (*field*) **ket** dengan tipe data TEXT (digunakan untuk menampung string yang memiliki banyak karakter) dengan panjang (*length*) maksimal 100 karakter.

4.7.4 Mengetahui Struktur Tabel

Dalam MySQL ketika anda ingin melihat stuktur tabel, anda dapat menggunakan perintah:

```
DESC nama_tabel;
```

Contoh:

```
DESC tbl_barang;
```

Memberikan hasil berikut:

Field	Type	Null	Key	Default	Extra
id	varchar(5)	NO	PRI	(NULL)	
nama	varchar(25)	YES		(NULL)	
jumlah	int(11)	YES		(NULL)	
ket	tinytext	YES		(NULL)	

Gambar 4. 4 Struktur tabel diperoleh via DESC tbl_barang.

4.7.5 Memasukkan Data

Tabel `tbl_barang` yang sebelumnya anda buat tentu saja masih berupa tabel yang kosong karena belum ada baris (*row*) data yang sama sekali tersimpan. Oleh karena itu, untuk memasukkan data ke dalam tabel tersebut, kita bisa gunakan pernyataan INSERT. Sebagai contoh, perintah berikut digunakan digunakan untuk memasukkan data barang bernama Buku.

```
INSERT INTO tbl_barang  
VALUES ('B0001','Buku',100,'Tersedia');
```

Urutan data pada VALUES sesuai dengan urutan nama kolom dalam pendefinisian struktur tabel. Untuk data jumlah kenapa tidak menggunakan tanda petik (‘’)? Hal itu disebabkan pada MySQL atau DBMS lain yang menggunakan SQL sebagai Bahasa standar dalam mengelola *database* bahwa tipe data integer atau bilangan bulat berupa angka tidak perlu menggunakan tanda petik, tanda petik tersebut diperuntukkan untuk nilai yang berupa string.

Pernyataan INSERT memungkinkan penambahan baris dengan kolom-kolom tertentu saja yang diisi. Kolom-kolom yang tidak disebutkan akan diisi dengan NULL. Perlu diperhatikan, kolom-kolom yang tidak disebutkan dalam INSERT haruslah yang tidak mengandung NOT NULL dalam pendefinisian struktur tabel.


4.7.6 Melihat Isi Tabel

Sejauh ini anda telah memasukkan data dua buah barang. Kemudian, bagaimana cara anda melihat isi tabel barang yang baru saja anda isi sebelumnya? MySQL memiliki perintah yang dapat kita gunakan ketika hendak melihat isi data pada suatu tabel yaitu dengan perintah SELECT. Sebagai contoh, cobalah pernyataan berikut:

```
SELECT * FROM tbl_barang;
```

Pada pernyataan ini, tanda “*” merupakan tanda untuk memilih semua kolom sedangkan “tbl_barang” yang terletak setelah kata FROM menyatakan nama tabel yang diproses.

Contoh berikut memberikan gambaran hasil (*output*) yang diperoleh dari pernyataan di atas.

 id	nama	jumlah	ket
B0001	Buku	100	Tersedia
B0002	Pulpen	100	Tersedia

Gambar 4. 5 Isi tbl_barang ditampilkan dengan perintah SELECT.

CATATAN :

Ada banyak variasi dalam menggunakan pernyataan SELECT. Untuk penggunaan SELECT yang lebih kompleks mungkin akan dibahas pada buku tersendiri tentang *Database*.

4.7.7 Mengganti Struktur Tabel

Adakalanya suatu ketika diperlukan untuk mengubah struktur suatu tabel. Oleh karena itu, MySQL menyediakan perintah yang dapat anda gunakan, untuk menjalankan perintah tersebut gunakan pernyataan ALTER TABLE. Sebagai contoh, anda bisa mempraktikkan pengubahan nama kolom **ket** menjadi **keterangan**. Perintah yang diperlukan:

```
ALTER TABLE tbl_barang  
CHANGE ket keterangan TEXT(100);
```

Pada contoh di atas, nama kolom **ket** dalam tabel tbl_barang diubah menjadi **keterangan** namun dengan tetap memepertahankan tipe data dan *length*-nya.

Dalam kasus lainnya, terkadang kita bisa saja salah mendefinisikan suatu tipe data pada suatu kolom. Oleh karena itu, anda dapat menggunakan perintah yang sama seperti sebelumnya. Sebagai contoh:

```
ALTER TABLE tbl_barang  
CHANGE keterangan keterangan ENUM('Tersedia','Tidak  
Tersedia') NOT NULL;
```

Membuat tipe data kolom keterangan menjadi ENUM ('Tersedia','Tidak Tersedia') dan kolom harus diisi (NOT NULL). Dengan menggunakan tipe data seperti ini, maka kolom keterangan hanya bisa diisi dengan salah satu nilai yang tercantum dalam ENUM, yaitu "Tersedia" atau "Tidak Tersedia", adapun penulisannya harus sesuai (*case sensitive*).

CATATAN :

Ketika kolom **keterangan** diisi selain dengan "Tersedia" atau "Tidak Tersedia" maka MySQL akan mengubah isian di luar nilai dari pada ENUM dengan spasi, sehingga anda akan mendapati isi pada baris tertentu pada kolom keterangan kosong (Bukan Nilai yang diinputkan sebelumnya).

Selain pernyataan sebelumnya yaitu:

```
ALTER TABLE tbl_barang  
CHANGE keterangan keterangan ENUM('Tersedia','Tidak  
Tersedia') NOT NULL;
```

Hasilnya adalah mengubah definisi suatu kolom, dapat disederhanakan menjadi:

```
ALTER TABLE tbl_barang  
CHANGE keterangan ENUM('Tersedia','Tidak Tersedia') NOT  
NULL;
```

MySQL juga mendukung pernyataan ALTER TABLE untuk menambahkan kolom, contoh:


```
ALTER TABLE tbl_barang
ADD COLUMN id_jenis_barang INTEGER NOT NULL;
```

Dengan cara seperti di atas, kolom bernama **id_jenis_barang** dengan tipe data INTEGER dengan sendirinya ditambahkan pada tabel **tbl_barang**. Gambar 4. 6 memperlihatkan keadaan struktur tabel setelah perintah **DESC tbl_barang** diberikan.

Field	Type	Null	Key	Default	Extra
id	varchar(5)	NO	PRI	(NULL)	
nama	varchar(25)	YES		(NULL)	
jumlah	int(11)	YES		(NULL)	
ket	tinytext	YES		(NULL)	
id_jenis_barang	int(11)	NO		(NULL)	

Gambar 4. 6 Struktur Tabel **tbl_barang** setelah kolom **id_jenis_barang** di tambahkan.

Tampaknya bahwa kolom **id_jenis_barang** telah ditambahkan dan diletakkan diakhir pada struktur tabel.

CATATAN :

Ketika kolom **id_jenis_barang** dengan perintah berikut:

```
ALTER TABLE tbl_barang
ADD COLUMN id_jenis_barang INTEGER NOT NULL;
```

Tentunya hal ini akan membuat kolom **id_jenis_barang** secara default ditambahkan pada akhir sruktur tabel. Namun, apa yang terjadi jika **id_jenis_barang** ingin ditambahkan setelah kolom **id**? MySQL menyediakan pernyataan **BEFORE/AFTER**. **BEFORE** digunakan untuk menambahkan kolom sebelum kolom acuan dan **AFTER** digunakan untuk menambahkan kolom setelah kolom acuan. Penggunaannya dapat dilihat dari perintah berikut:

```
ALTER TABLE tbl_barang
ADD COLUMN id_jenis_barang INTEGER NOT NULL
AFTER id;
```

Pernyataan ALTER TABLE juga bisa digunakan untuk menghapus suatu kolom yang dikehendaki. Contoh:

```
ALTER TABLE tbl_barang  
DROP id_jenis_barang;
```

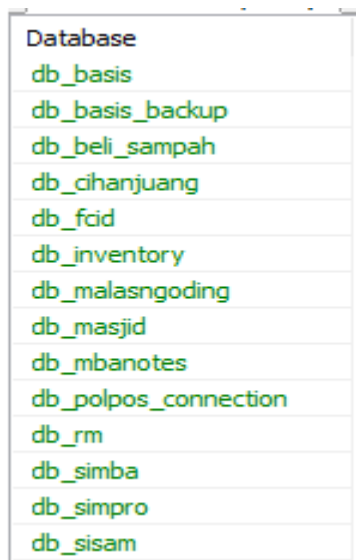
Perintah di atas membuat kolom bernama “**id_jenis_barang**” yang sebelumnya terdapat pada tabel `tbl_barang` dihapus.

4.7.8 Mengetahui Daftar *Database* dan Tabel

Ketika anda menggunakan program klien MySQL dan anda hendak ingin menampilkan daftar *database*, anda dapat menggunakan perintah berikut:

```
SHOW DATABASES;
```

Gambar 4. 7 memperlihatkan contoh hasil setelah perintah di atas diberikan:



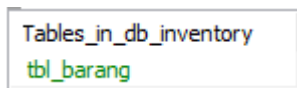
Database
db_basis
db_basis_backup
db_beli_sampah
db_cihanjuang
db_fcid
db_inventory
db_malasngoding
db_masjid
db_mbanotes
db_polpos_connection
db_rm
db_simba
db_simpro
db_sisam

Gambar 4. 7 Daftar *database* pada MySQL via SHOW DATABASES.

Selain itu, untuk mengetahui daftar tabel yang terdapat pada suatu *database*, anda bisa memberikan menggunakan perintah berikut:

```
SHOW TABLES;
```

Gambar 4. 8 Memberikan contoh hasil dari perintah di atas guna menampilkan tabel yang terdapat pada *database* db_inventory.



Gambar 4. 8 Daftar tabel yang diperoleh via SHOW TABLES.

Hasil pada Gambar 4. 8 menyatakan bahwa hanya ada satu tabel pada *database* db_inventory.

4.7.9 Menghapus Tabel

Sebagai manusia yang pastinya akan melakukan kesalahan, maka kerap kali ada satu atau beberapa tabel yang seharusnya tidak perlukan tapi sebelumnya sudah pernah dibuat/diciptakan. Oleh karena itu, MySQL menyediakan pernyataan DROP TABLE yang dapat anda gunakan untuk menghapus tabel yang sudah tidak dibutuhkan lagi. Untuk menggunakannya, anda bisa menggunakan perintah:

```
DROP TABLE tbl_barang;
```

Pernyataan di atas menghapus tabel tbl_barang yang telah dibuat sebelumnya. Jangan ragu-ragu melakukan perintah tersebut supaya anda mengerti dan tahu betul efeknya sekarang, sehingga tidak melakukan hal serupa pada kesempatan lain ketika sebenarnya anda tidak berniat menghapus tabel tersebut. Sekali anda mengeksekusi perintah tersebut, tabel akan hilang dan tidak ada acara untuk

mengembalikannya! Namun untuk keperluan latihan, sekali lagi jangan ragu-ragu untuk melakukannya.

4.7.10 Menghapus *Database*

Apabila anda bermaksud untuk menghapus suatu *database* pada MySQL karena suatu keadaan atau keperluan tertentu, anda dapat melakukannya dengan memakai pernyataan DROP DATABASE. Contoh penerapannya sebagai berikut:

```
DROP TABLE db_inventory;
```

Pernyataan di atas menghapus *database* db_inventory yang telah dibuat sebelumnya. Namun, sangat disarankan untuk tidak melakukan hal ini karena dalam suatu industry dengan skala besar, menghapus *database* tentunya akan menghapus semua data yang terkandung di dalamnya. Namun, jika anda ingin menghapus *database* tersebut karena keperluan rekonstruksi dan lainnya, maka cukup lakukan backup terlebih dahulu untuk mencegah membangun *database* dari awal.

BAB V

FRAMEWORK CODEIGNITER

5.1 Say Hello Dengan *Framework* Codeigniter

Welcome to Jumanji. Eh, salah hahaha, maksud saya Selamat datang di pembahasan tentang *Framework* Codeigniter. Sebelum *era Object Oriented Programming* (OOP) memegang kendali dunia pemrograman yang populer saat ini. *Programmer* dulunya masih menggunakan paradigma pemrograman Struktural. Apa itu paradigma pemrograman struktural? Lalu, apa itu paradigma pemrograman OOP? Mengapa OOP bisa menggantikan posisi paradigma pemrograman struktural? Baiklah, akan segera kita bahas.

Paradigma pemrograman struktural atau terstruktur yaitu suatu bahasa pemrograman yang sangat mendukung penuh dalam pembuatan program sebagai kumpulan dari prosedur untuk memecahkan masalah. Prosedur/langkah-langkah ini lah nantinya akan saling memanggil dan dipanggil dari manapun di dalam sebuah program tersebut dengan perintah berbeda dari setiap *syntak* pemanggilan program [9]. Namun, seiring berjalannya waktu paradigma pemrograman ini sedikit mulai ditinggalkan dan digantikan paradigma baru yaitu OOP. OOP menutupi kekurangan dari Pemrograman Terstruktur.

OOP adalah sebuah istilah yang diberikan kepada bahasa pemrograman yang menggunakan teknik berorientasi atau berbasis pada sebuah obyek dalam pembangunan program aplikasi, maksudnya bahwa orientasi pembuatan program tidak lagi menggunakan orientasi linear melainkan berorientasi pada objek-objek yang terpisah-pisah. Suatu perintah dalam bahasa ini diwakili oleh sebuah Obyek yang didalamnya berisi beberapa perintah-perintah standar sederhana. Obyek ini dikumpulkan dalam Modul form atau Report atau modul lain dan disusun didalam sebuah project [10].

Pada pembahasan ini sesuai dengan judul akan di jelaskan tentang pengertian dan cara menggunakan codeigniter. tentu anda pasti sudah sering

mendengar tentang framework codeigniter. Baik itu dari teman, kampus, sekolah atau dari forum-forum pemrograman di internet. Tapi tidak sedikit juga yang masih bertanya-tanya tentang apa sih framework codeigniter itu?

Oleh sebab itu, pada pembahasan khusus BAB *framework codeigniter* ini ini kita akan sedikit berkenalan dengan codeigniter. Dan belajar bagaimana cara melakukan instalasi codeigniter. Kita akan mulai dengan sedikit pengertian dari apa itu *framework*? dan apa itu *framework* codeigniter?

Baikklah, tanpa basa basi lagi, langsung saja, mari kita mulai...

Framework adalah kumpulan intruksi-intruksi yang di kumpulkan dalam *class* dan *function-function* dengan fungsi masing-masing untuk membantu memudahkan developer/programmer dalam pemanggilannya tanpa harus menuliskan syntax program yang sama secara berulang-ulang. hal ini memiliki kegunaan untuk menghemat waktu dan mencegah penulisan syntax secara berulang-ulang agar tercipta-nya *source code* (kode program) yang bersih dan terstruktur.

Codeigniter adalah sebuah *framework* PHP yang bersifat *open source* dan menggunakan metode Model, View dan Controller (MVC). codeigniter bersifat free alias tidak berbayar jika anda menggunakannya. *Framework* codeigniter di buat dengan tujuan sama seperti *framework* lainnya yaitu untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuat-nya dari awal.

Framework codeigniter juga termasuk salah satu *framework* PHP yang paling banyak digunakan oleh web developer untuk membangun sebuah aplikasi berbasis *website*.

Pengembangan codeigniter oleh developer-nya juga sangat baik. Dan memiliki dokumentasi yang baik dan jelas sebagai panduan kita menggunakan *framework* codeigniter.

5.1.1 Pengertian Model, View dan Controller (MVC)

Pada pengertian codeigniter sebelumnya dijelaskan bahwa codeigniter menggunakan metode MVC. Apa itu MVC? kita juga harus mengetahui apa itu MVC sebelum masuk dan lebih dalam mengoprek codeigniter.

MVC adalah teknik atau konsep yang memisahkan komponen utama menjadi tiga komponen yaitu model, view dan controller.

1. Model

Model merupakan bagian penanganan yang berhubungan dengan pengolahan atau manipulasi *database*. seperti misalnya mengambil data dari *database*, meng-*input* dan pengolahan *database* lainnya. semua intruksi atau fungsi yang berhubungan dengan pengolahan *database* di letakkan di dalam model.

2. View

View merupakan bagian yang menangani halaman *user interface* atau halaman yang muncul pada *user* (pada *browser*). Tampilan dari *user interface* di kumpulkan pada view untuk memisahkannya dengan controller dan model sehingga memudahkan *web designer* dalam melakukan pengembangan tampilan halaman *website*.

3. Controller

Controller merupakan kumpulan intruksi aksi yang menghubungkan model dan view, jadi *user* tidak akan berhubungan dengan model secara langsung, intinya data yang tersimpan di *database* (model) di ambil oleh controller dan

kemudian controller pula yang menampilkan-nya ke view. Jadi controller lah yang mengolah intruksi.

Dari penjelasan tentang model view dan controller sebelumnya dapat disimpulkan bahwa controller sebagai penghubung view dan model. Jadi, misalnya pada aplikasi yang menampilkan data dengan menggunakan metode konsep MVC, controller memanggil intruksi pada model yang mengambil data pada *database*, kemudian controller yang meneruskannya pada view untuk di tampilkan. Jadi sudah jelas dan sangat mudah dalam pengembangan aplikasi dengan cara MVC ini karena *web designer* atau *front-end developer* tidak perlu lagi berhubungan dengan controller, dia hanya perlu berhubungan dengan view untuk men-*design* tampilann aplikasi, karena *back-end developer* yang menangani bagian controller dan modelnya. jadi pembagian tugas pun menjadi mudah dan pengembangan aplikasi dapat dilakukan dengan cepat dan terstruktur.

5.1.2 Kelebihan *Framework Codeigniter*

Adapun beberapa kelebihan dari codeigniter adalah :

- *Syntax* yang terstruktur dan bersih
- Kemudahan dalam menggunakannya
- Kodeigniter menyediakan fasilitas helper dan library yang dapat membantu developer dalam membuat pagination, session, manipulasi url dan lainnya secara cepat yang akan dipelajari pada tutorial codeigniter selanjutnya.
- keamanan yang sudah lumayan karena user atau pengakses aplikasi tidak berhubungan langsung dengan *database*.

Setelah selesai berkenalan dengan framework codeigniter, selanjutnya kita akan masuk ke tahap belajar cara menginstal framework codeigniter dan mulai menggunakannya.

5.2 Instalasi *Framework* Codeigniter

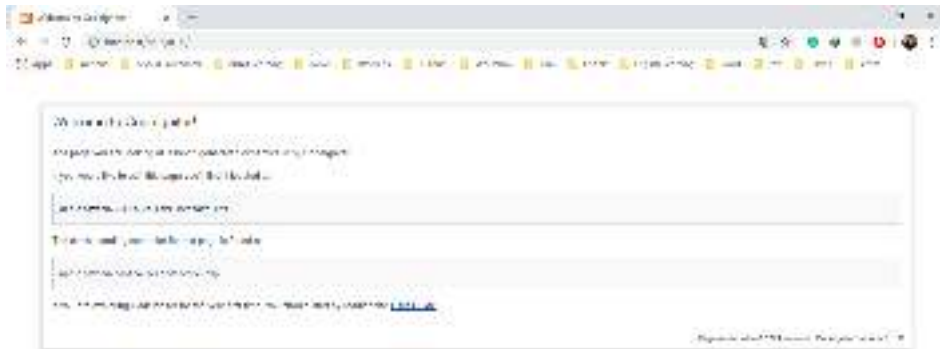
Instalasi Framework codeigniter caranya sangatlah mudah. anda dapat langsung mendownload CodeIgniter pada situs resminya di www.codeigniter.com.



Gambar 5. 1 Halaman utama *website* resmi Codeigniter.

Setelah selesai mendownload *framework* codeigniter, kemudian ekstrak di folder htdocs (localhost) jika menggunakan XAMPP. Lalu, ubah nama foldernya dengan nama project yang ingin anda buat. Pada contoh ini anda dapat mengubah nama foldernya dari “codeigniter.3.1.5” menjadi “belajar-ci”.

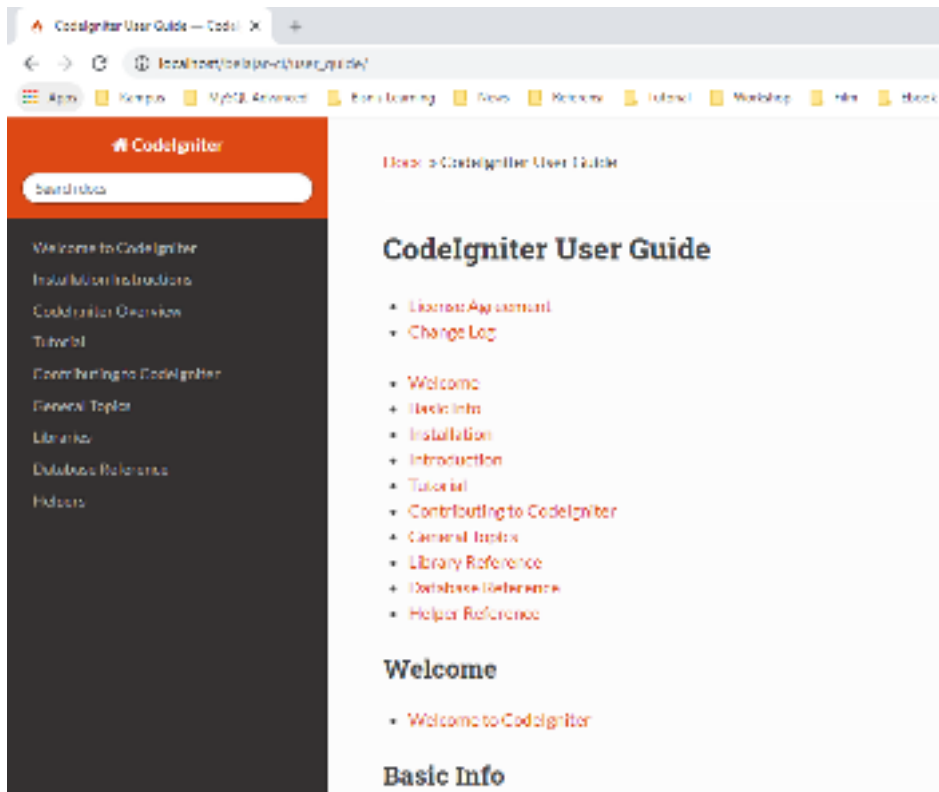
Terakhir, Anda dapat langsung menjalankannya pada *browser* kesayangan anda dengan mengetikan alamat “http://localhost/belajar-ci” pada *address bar browser* anda.



Gambar 5. 2 Halaman *welcome codeigniter* dari project belajar-ci

Jika berhasil, maka akan muncul tampilan seperti *Gambar 5. 2* yang menandakan bahwa codeigniter telah berhasil di *install* dan siap untuk di gunakan. Tampilan tersebut merupakan halaman *default* dari codeigniter saat pertama kali di-*install*.

Secara *default* Codeigniter menyediakan panduan cara penggunaan codeigniter pada folder codeigniter yang anda *download* sebelumnya, yaitu terdapat pada folder *user_guide*. Adapun untuk mempelajari panduan dari *user guide* tersebut anda dapat langsung menjalankannya pada browser dengan mengetikkan alamat “http://localhost/belajar-ci/user_guide/” pada *address bar*.



Gambar 5. 3 Halaman *user-guide codeigniter* dari project belajar-ci

Segala tentang penggunaan codeigniter bisa di baca pada panduan yang telah di sediakan seperti pada Gambar 5. 3.

5.3 Apa itu Uri Segment?

Pada tutorial codeigniter di buku akan menjelaskan tentang salah satu yang paling terpenting dalam konsep penggunaan codeigniter, yaitu Uri Segment pada codeigniter. Jadi pada tutorial ini kita akan belajar tentang pengenalan dari uri segment codeigniter. apa pengertian uri segment pada codeigniter?

URI adalah singkatan dari *Uniform Resource Identifier*. bisa kita bilang URI yang membantu kita dalam mengambil data melalui url codeigniter. Cara penyebutan uri segment pada codeigniter sendiri misalnya segment 1, segment

2, segment 3 dan seterusnya. jika teman-teman perhatikan pada url codeigniter pada project codeigniter anda, pasti menjumpai index.php lalu di lanjutkan dengan nama class codeigniter anda kan?. index.php di sini tidak berpengaruh dengan uri segment bahkan index.php dapat kita hilangkan untuk membuat url codeigniter kita lebih rapi.

Untuk tutorial menghilangkan index.php pada codeigniter akan dibahas pada tutorial selanjutnya. Kembali lagi ke penjelasan URI segment, yang di katakan sebagai segment 1 pada codeigniter adalah nama class atau controller yang sedang anda jalankan. segment ke 2 adalah nama method atau function pada dari class/controller anda. dan segment 3 berupa data berbentuk id yang di kirim.

Jadi segment codeigniter di hitung dari setelah index.php pada codeigniter. katakan saja misalnya anda memiliki url seperti contoh berikut ini:

`http://localhost/belajar-ci/index.php/belajar/codeigniter/1`

Baiklah, mari kita perhatikan pada contoh url di atas.

segment 1 = belajar

segment 2 = codeigniter

segment 3 = 1

Untuk memberikan penjelasan yang lebih rinci berikut ini adalah rumus cara mudah mengerti tentang url codeigniter.

`http://localhost/belajar-ci/index.php/controller/method/id`

Data yang di kirimkan melalui url di codeigniter biasanya terletak pada segment 3. Perhatikan contoh berikut ini.

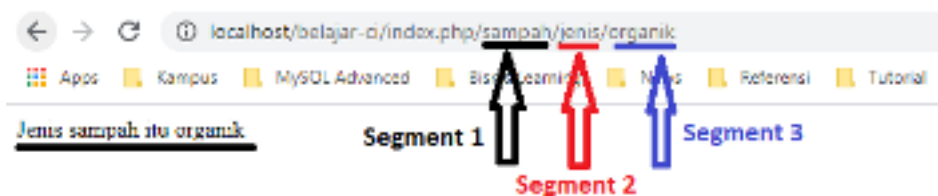
Pertama, buatlah sebuah controller baru dengan nama “Sampah.php” (tanpa tanda petik).

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

class Sampah extends CI_Controller
{
    public function jenis(){
        echo "Jenis sampah itu " . $this->uri->segment('3');
    }
}
?>
```

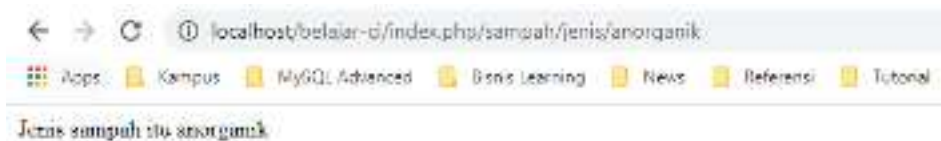
Coba anda perhatikan pada contoh di atas. kita membuat sebuah controller dengan nama sampah.php dan method/function jenis. Pada function jenis ini saya menampilkan isi dari URI segment ketiga (\$this->uri->segment('3')).

Oleh karena itu, untuk mengakses uri segment ini, anda dapat menggunakan syntax “\$this->uri->segment(‘urutan segment beberapa’)”. Maka hasilnya akan terlihat pada Gambar 5. 4



Gambar 5. 4 Belajar uri segment: sampah/jenis/organik

Dari Gambar 5. 4, bisa anda perhatikan pada segment 3, segment 3 yang digaris bawahhi dengan tanda berwarna biru adalah “organik”. Maka hasil yang muncul adalah sampah itu termasuk jenis samoah organik. Untuk mengujinya lagi, sekarang coba anda ubah pada segment 3 dari “organik” menjadi “anorganik”.



Gambar 5. 5 Belajar uri segment: sampah/jenis/anorganik

Perhatikan lagi contoh berikut ini dan semoga dapat memberi pemahaman lebih mendalam tentang uri segment pada codeigniter.

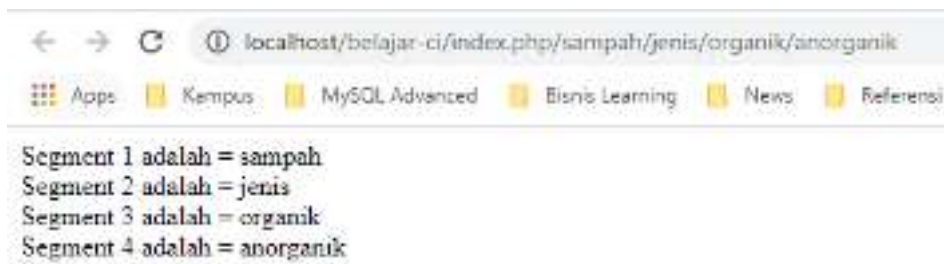
```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Sampah extends CI_Controller
{
    public function jenis(){
        echo "Segment 1 adalah = " . $this->uri->segment('1') . "<br/>";
        echo "Segment 2 adalah = " . $this->uri->segment('2') . "<br/>";
        echo "Segment 3 adalah = " . $this->uri->segment('3') . "<br/>";
        echo "Segment 4 adalah = " . $this->uri->segment('4') . "<br/>";
    }
}
?>
```

Kemudian, silahkan akses alamat berikut:

<http://localhost/belajar-ci/index.php/sampah/jenis/organik/anorganik>

Maka akan muncul hasil seperti pada Gambar 5. 6.



Gambar 5. 6 Belajar uri segment: sampah/jenis/organik/anorganik

5.4 Berkenalan Dengan *Controller* di Codeigniter

Pada tutorial kali ini tentang Berkenalan dengan Controller Codeigniter ini akan menjelaskan tentang cara membuat controller pada codeigniter, pengaturan dasar dan cara menggunakan controller pada codeigniter. Seperti yang sudah di jelaskan pada tutorial sebelumnya tentang pengertian dan cara menggunakan codeigniter bahwa codeigniter menggunakan metode MVC untuk menciptakan kode atau *syntax* yang bersih.

Sebelum memasuki pengenalan ini, anda di asumsikan untuk memahami konsep OOP terlebih dahulu sebelum mulai mempelajari codeigniter. karena codeigniter di bangun dengan menggunakan konsep OOP.

Controller dalam bahasa indonesia yang berarti pengontrol atau pengatur, yang di maksud pengontrol atau pengatur di sini adalah controller yang berperan sebagai pengatur dari aksi pada aplikasi yang di bangun, seperti misalnya jika di codeigniter controller yang berperan paling penting dari mulai mengirimkan parameter, menangani inputan form (*form handling*), mengatur view dan model, mengaktifkan atau memanggil library dan helper codeigniter

dan masih banyak lagi peran controller dalam membangun sebuah aplikasi dengan menggunakan *framework* php codeigniter.

Untuk memahami cara penggunaan controller codeigniter sekarang coba jalankan project Codeigniter yang sudah di install pada localhost, di buku ini nama folder project *root*-nya adalah “belajar-ci”, sehingga untuk menjalankannya pada browser dapat di akses langsung dengan alamat <http://localhost/belajar-ci/>.

Setelah berhasil, maka akan muncul halaman *welcome* codeigniter, hal ini terjadi karena pengaturan default controller yang di jalankan pertama kali pada konfigurasi routes adalah controller welcome. dapat di lihat pada `application/controllers/welcome.php`.

Controller ini memanggil view `welcome_message.php` yang terletak pada folder view. view tidak kita jelaskan secara mendetail pada sub judul ini. melainkan akan kita bahas pada tutorial selanjutnya di ebook ini tentang cara membuat view pada codeigniter.

Secara *default* saat pertama kali meng-*install* codeigniter dan menjalankannya, controller codeigniter yang pertama kali di jalankan adalah controller welcome, saat anda menjalankannya pada *browser* maka yang pertama kali di jalankan adalah controller welcome. Hal ini dapat di lihat pada pengaturan routes codeigniter yang menetapkan controller welcome sebagai controller default yang di jalankan. Buka project controller yang sudah anda letakkan pada localhost sesuai dengan tutorial sebelumnya. Kemudian, buka *file* pengaturan routes.php yang terletak di `application/config/routes.php`

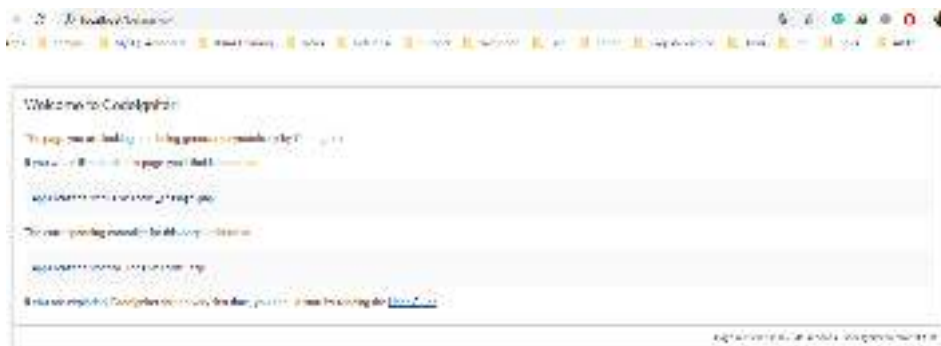
Application/config/routes.php

```
$route['default_controller'] = 'welcome';  
$route['404_override'] = '';  
$route['translate_uri_dashes'] = FALSE;
```

Dapat dilihat pada pengaturan routes codeigniter di atas, pada pengaturan **default_controller** di setting controller “ welcome “, selain itu, terdapat juga pengaturan untuk menangani halaman 404 atau halaman yang di tampilkan jika tidak di temukannya data suatu url. Anda dapat mengatur halaman 404 anda dengan cara memasukkan controller yang ingin anda jadikan untuk menetapkan halaman 404 pada aplikasi anda.

Pada pengaturan \$route[‘translate_uri_dashes’]=false adalah pengaturan untuk menetapkan nilai true atau false untuk izin penggunaan tanda “-” (dash) pada controller di url pada saat dijalankan. Namun pada buku ini kita tidak akan terlalu banyak membahasnya.

Controller *default* ini dapat diakses secara langsung dengan alamat <http://localhost/belajar-ci>.



Gambar 5. 7 Berkenalan dengan controller: default controller

5.4.1 Cara Membuat Controller Pada Codeigniter

Jika sebelumnya kita menggunakan controller *default* bawaan saat pertama kali codeigniter di-*install*, maka sekarang kita akan belajar membuat controller baru.

Untuk membuat controller baru pada codeigniter, Silahkan buat sebuah file baru pada application/controllers/. untuk contoh di sini kita akan membuat controller “belajar”. Jadi kita harus membuat file belajar.php di application/controllers/.

Selanjutnya kita akan mendefinisikan class belajar dalam file “**belajar.php**”.

Application/controllers/belajar.php

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

class Belajar extends CI_Controller
{
    function __construct(){
        parent::__construct();
    }

    public function index(){
        echo "ini method index pada controller belajar";
    }

    public function halo(){
        echo "ini method halo pada controller belajar";
    }
}
?>
```

Penjelasan:

Mari perhatikan pada controller yang telah dibuat. Pertama kali yang harus dilakukan adalah meng-*extends* controller baru ini dengan CI_Controller.

```
Class Belajar extends CI_Controller {
```

Dalam hal ini, nama class harus diawali dengan huruf besar seperti contoh di atas. Nama class harus sesuai dengan nama *file* controller yang dibuat tadi. Karena tadi kita membuat controller **belajar.php** maka penulisan class-nya seperti di atas.

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

Syntax di atas berfungsi untuk mencegah akses langsung pada *file* controller. Kemudian, anda dapat membuat function construct() untuk menjalankan fungsi yang anda inginkan pada saat controller diakses. biasanya pada fungsi construct() di letakkan fungsi untuk memanggil helper atau library (akan dijelaskan pada sub selanjutnya).

Kemudian, untuk contoh method pada controller dalam buku ini, saya membuat dua buah method yaitu index dan halo.

```
public function index(){  
    echo "ini method index pada controller belajar";  
}  
  
public function halo(){  
    echo "ini method halo pada controller belajar";  
}
```

Untuk menjalankan method index anda bisa mengaksesnya menggunakan alamat:

<http://localhost/belajar-ci/index.php/belajar>

Untuk method index boleh tidak dituliskan karena pada saat controller di akses, maka yang pertama di jalankan adalah method index atau bisa juga dengan menambahkan index langsung seperti alamat berikut:

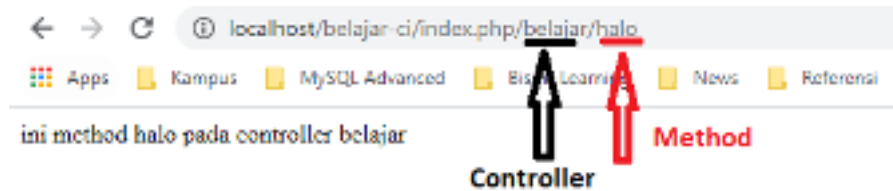
<http://localhost/belajar-ci/index.php/belajar/index>



Gambar 5. 8 Berkenalan dengan controller: belajar/index

Kemudian, untuk mengakses function atau method halo dapat di akses dengan alamat:

<http://localhost/belajar-ci/index.php/belajar/halo>



Gambar 5. 9 Berkenalan dengan controller: belajar/halo

Jika anda ingin menjadikan controller belajar ini sebagai controller *default* anda dapat melakukan *setting*-nya di `application/config/routes.php`.

Ubah pengaturan `default_controller` menjadi belajar (nama controller yang ingin di set default).

```
$route['default_controller'] = 'belajar';
```

5.5 Berkenalan Dengan View

Setelah mempelajari tentang cara membuat controller pada codeigniter dan pengertian serta cara menggunakan codeigniter pada tutorial sebelumnya, Maka untuk kelanjutan dari tutorial codeigniter dasar kita akan membahas tentang cara membuat view pada codeigniter. Seperti yang sudah di jelaskan bahwa view pada codeigniter berguna untuk meng-*handle* urusan tampilan dari aplikasi yang kita buat menggunakan codeigniter.

5.5.1 Pengenalan View

View bertugas menampilkan *user interface* kepada *user*, sesuai dengan fungsi MVC yang memisahkan model, controller dan view sehingga memudahkan developer atau programmer dalam membuat

pembaharuan serta dapat memudahkan developer bekerja dalam tim pada saat membangun aplikasi berbasis web menggunakan codeigniter. Sehingga *web designer* yang menangani tampilan *user interface* tidak perlu berurusan dengan *back-end* karena jatah kerja *web designer* ada pada view yang terletak pada folder view pada codeigniter. application/view/.

5.5.2 Cara Membuat View

Cara membuat view pada codeigniter caranya sangat mudah, anda hanya perlu membuat file baru pada folder view. Yaitu di application/view/.

Kemudian, buat view dengan nama *file*-nya terserah anda. Di contoh ini, kita akan membuat view dengan nama view_belajar.php “application/view/view_belajar.php” langkah selanjutnya adalah membuat isi dari view. **application/view/view_belajar.php**.

```
<html>
<head>
  <title>Cara Membuat View Pada CodeIgniter</title>
</head>
<body>
  <h1>Cara Membuat View Pada CodeIgniter</h1>
  <h2>Ini adalah view view_belajar.php</h2>
  <h3>Ini adalah view yang di tampilkan pada controller belajar,
method halo</h3>
</body>
</html>
```

Dan untuk cara memanggil atau menampilkan view nya silahkan buka controller anda, di sini saya membuat controller belajar.php dan membuat method halo.

```

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

class Belajar extends CI_Controller
{
    function __construct(){
        parent::__construct();
    }

    public function index(){
        echo "ini method index pada controller belajar";
    }

    public function halo(){
        $this->load->view('view_belajar');
    }
}
?>

```

Perhatikan pada method halo di atas. Pada method halo ini, controller memerintahkan system untuk memanggil komponen view view_belajar.php untuk ditampilkan di *user interface*.

```

public function halo(){
    $this->load->view('view_belajar');
}

```

Pada kasus kali ini, anda tidak perlu lagi menuliskan ekstensi ‘.php’ pada saat memanggil view. syntax `$this->load->view()` dengan otomatis mengakses folder application/view codeigniter.

Dan coba kita jalankan method halo nya pada browser untuk menampilkan view view_belajar.php dengan mengakses alamat <http://localhost/belajar-ci/index.php/belajar/halo>.



Gambar 5. 10 Berkenalan dengan view: view_belajar.php

Dapat di perhatikan pada contoh di atas. view view_belajar.php pun muncul pada saat kita akses method halo. Kemudian muncul pertanyaan, Bagaimana cara membuat view di dalam subfolder?. Misalnya untuk sebagian view ada yang ingin anda kumpulkan pada sebuah folder untuk membuat view lebih rapi dan terstruktur. misalnya anda meletakkan view yang ingin anda panggil di dalam suatu folder. Jika anda meletakkan view view_belajar.php di dalam folder “front” di dalam view codeigniter.

application/view/front/view_belajar.php

Dan cara memanggil view view_belajar.php yang terletak di dalam folder front, caranya anda hanya perlu memasukkan juga nama folder tempat view yang anda ingin ditampilkan pada berada.

```
$this->load->view('front/view_belajar');
```

5.5.3 Cara Parsing Data Ke View

Untuk memarsing data dari controller ke view anda dapat memarsingnya dengan menggunakan bantuan array. Jadi data yang akan diparsing kita masukkan ke array.

Perhatikan contoh cara memarsing data ke view codeigniter berikut ini.

Buka controller **belajar.php**

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

class Belajar extends CI_Controller
{
    function __construct() {
        parent::__construct();
    }

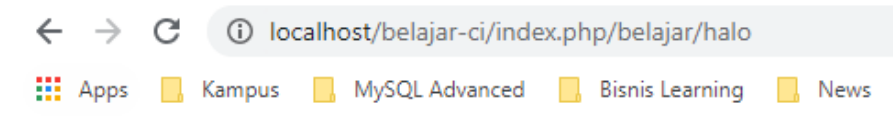
    public function index() {
        echo "ini method index pada controller belajar";
    }

    public function halo() {
        $data['nama_web'] = "Webkita.com";
        $this->load->view('view_belajar',$data);
    }
}
?>
```

Application/view/view_belajar.php

```
<html>
<head>
    <title>Cara Membuat View Pada CodeIgniter</title>
</head>
<body>
    <h1><?php echo $nama_web; ?></h1>
</body>
</html>
```

Untuk menguji hasil parsing data ke view, silahkan akses halaman [“http://localhost/belajar-ci/index.php/belajar/halo”](http://localhost/belajar-ci/index.php/belajar/halo).



Webkita.com

Gambar 5. 11 Berkenalan dengan view: parsing data ke view

Perhatikan pada contoh di atas data di parsing pada dengan memasukkan variabel ke dalam parameter kedua pada syntax:

`$this->load->view()`.

```
public function halo() {  
    $data['nama_web'] = "Webkita.com";  
    $this->load->view('view_belajar',$data);  
}
```

Dan dari view tinggal mengakses variabel `$nama_web`.

```
<h1><?php echo $nama_web; ?></h1>
```

Kemudia bisa juga menggunakan cara berikut.

`application/controller/belajar.php`

```
defined('BASEPATH') OR exit('No direct script access allowed');  
class Belajar extends CI_Controller  
{  
    function __construct() {  
        parent::__construct();  
    }  
}
```

```

public function index() {
    echo "ini method index pada controller belajar";
}

public function halo() {
    $data = array( 'judul' => "Cara Membuat View Pada
                  CodeIgniter", 'tutorial' => "CodeIgniter" );
    $this->load->view('view_belajar', $data);
}
}
?>

```

Application/view/view_belajar.php

```

<html>
<head>
    <title>Cara Membuat View Pada CodeIgniter</title>
</head>
<body>
    <h2><?php echo $judul; ?></h2>
    <h3><?php echo $tutorial; ?></h3>
</body>
</html>

```

Untuk menguji hasil parsing data ke view, silahkan akses halaman [“http://localhost/belajar-ci/index.php/belajar/halo”](http://localhost/belajar-ci/index.php/belajar/halo).



Gambar 5. 12 Berkenalan dengan view: parsing data ke view cara lain

5.6 Membuat *Template Web* Sederhana Pada Codeigniter

Pada sub bagian kali ini, buku ini menjelaskan tentang bagaimana membuat template *website* sederhana dengan menggunakan codeigniter. pada tutorial ini akan di jelaskan tentang cara menggunakan teknik *multiple view* pada codeigniter untuk membuat template website yang dinamis, sama seperti menggunakan include() pada bagian header dan footer pada penggunaan PHP *native*.

Tetapi di codeigniter kita membuat nya dengan men-*load* view codeigniter yang kita pisah-pisahkan sesuai keinginan, misalnya anda bisa memisahkan bagian header, footer, sidebar dan konten untuk mencegah pengulangan penulisan syntax dan memudahkan dalam hal memodifikasi *template website*.

5.6.1 *Templating* Dengan Codeigniter

Buat sebuah controller yang menampilkan sebuah view. Pada buku ini diarahkan dengan membuat controller web.php dan sebuah view dengan nama v_index.php.

application/controller/web.php

```
<html>
<head>
  <meta charset="UTF-8">
  <title>Templating dengan codeigniter</title>
  <link rel="stylesheet" type="text/css" href="php echo
base_url() ?&gt;assets/css/style.css"&gt;
&lt;/head&gt;
&lt;body&gt;
  &lt;div id="wrapper"&gt;
    &lt;header&gt;
      &lt;hgroup&gt;
        &lt;h1&gt;WebKita.com&lt;/h1&gt;
        &lt;h3&gt;Membuat Template sederhana dengan codeigniter&lt;/h3&gt;
      &lt;/hgroup&gt;</pre
```

```

<nav>
  <ul>
    <li><a href="<?php echo base_url(). 'index.php/web'
?>">Home</a></li>
    <li><a href="<?php echo base_url(). 'index.php/web/about'
?>">About</a></li>
  </ul>
</nav>
<div class="clear">

</div>
</header>
<section>
  <h1><?php echo $judul; ?></h1>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed
    do eiusmod tempor incididunt ut labore et dolore magna aliqua.
    Ut enim ad minim veniam, quis nostrud exercitation ullamco
    laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure
    dolor in reprehenderit in voluptate velit esse cillum dolore eu
    fugiat nulla pariatur. Excepteur sint occaecat cupidatat non
    proident, sunt in culpa qui officia deserunt mollit anim id est
    laborum.
  </p>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed
    do eiusmod tempor incididunt ut labore et dolore magna aliqua.
    Ut enim ad minim veniam, quis nostrud exercitation ullamco
    laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure
    dolor in reprehenderit in voluptate velit esse cillum dolore eu
    fugiat nulla pariatur.
  </p>
</section>
<footer>
  <a href="#">WebKita</a>
</footer>
</div>
</body>
</html>

```

Sebelum lanjut, jangan lupa untuk setting `base_url()` codeigniternya terlebih dulu agar kita bisa menghubungkan codeigniter dengan file css. file css-nya dibuat pada folder project di dalam folder belajar-ci/assets/css/style.css (Jika folder assets dan css belum ada silahkan buat terlebih dahulu)

assets/css/style.css

```
body{
  background: #eee;
  color: #333;
  font-family: sans-serif;
  font-size: 15px;
}

#wrapper{
  background: #eee;
  width: 1100px;
  margin: 20px auto;
}

#wrapper header{
  background: #232323;
  padding: 20px;
}

#wrapper header hgroup{
  float: left;
  color: #fff;
}

#wrapper header nav{
  float: right;
  margin-top: 50px;
}

#wrapper header nav ul{
  padding: 0;
  margin: 0;
}
```

```

#wrapper header nav ul li {
    float: left;
    list-style: none;
}

#wrapper header nav ul li a {
    padding: 15px;
    color: #fff;
    text-decoration: none;
}

.clear {
    clear: both;
}

footer {
    background: #232323;
    padding: 20px;
}

footer a {
    color: #fff;
    text-decoration: none;
}

section {
    padding: 20px;
}

```

5.6.2 *Setting Base_URL Pada Codeigniter*

Base_URL merupakan halaman di mana folder root dari project kita alamatkan guna membantu saat proses templating dan *redirect* pada codeigniter.

Untuk setting `base_url()` buka pada folder applications lalu pilih file `config.php` di `application/config/config.php`. Kemudian cari syntax berikut:


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

Lalu sesuai kan dengan letak project codeigniter anda sehingga menjadi:

```
$config['base_url'] = 'http://localhost/belajar-ci ';
```

Setelah proses setting selesai dan base_url sudah diarahkan ke lokasi project. Selanjutnya, dapat kita perhatikan pada contoh di atas bahwa untuk menghubungkan file view dengan file css memerlukan bantuan base_url().

```
<link rel="stylesheet" type="text/css" href="<?php echo  
base_url() ?>assets/css/style.css">
```

Hasil perintah di atas akan sama seperti:

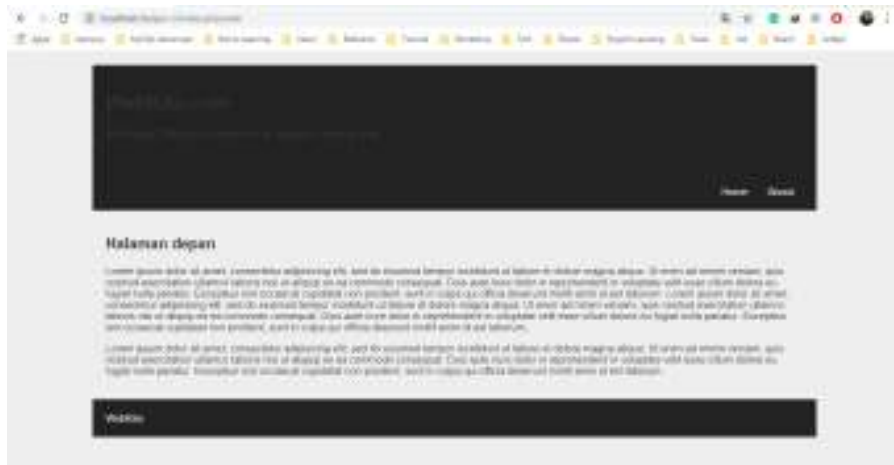
```
<link rel="stylesheet" type="text/css"  
href="http://localhost/belajar-ci/assets/css/style.css">
```

Kenapa demikian? Karena base_url() pada project sudah ditentukan menjadi “localhost/belajar-ci”. Dikarenakan kita menggunakan base_url() codeigniter, maka kita perlu mengaktifkan juga helper url seperti yang sudah kita buat pada controller web.php. silahkan perhatikan. Kita mengaktifkan helper url pada function construct().

```
function __construct(){  
    parent::__construct();  
    $this->load->helper('url');  
}
```

Jalankan controller web.php untuk melihat hasil dari *template* sederhana yang telah dibuat dengan mengakses halaman <http://localhost/belajar-ci/index.php/web>.

Berikut hasilnya:



Gambar 5. 13 *Templating* sederhana: tampilan web sederhana

Akhirnya, tampilan web sederhana pun berhasil kita buat pada codeigniter. Akan tetapi, halaman ini masih statis. kita masih perlu membuat bagian header dan footer secara berulang-ulang pada halaman lainnya.

5.6.3 Membuat Halaman *Template* Dinamis Dengan Codeigniter

Halaman *template* dinamis merupakan halaman yang dimana membantu developer atau programmer dalam membuat tampilan yang tidak berulang-ulang contoh untuk bagian header dan footer pada project kali ini. Untuk membuat halaman *template* yang dinamis caranya kita harus memecah *template* ini menjadi beberapa bagian. yaitu header dan footer. sehingga template sederhana ini menjadi:

application/view/v_header.php

```
<html>
<head>
  <meta charset="UTF-8">
  <title>WebKita | Membuat Template sederhana
codeigniter</title>
  <link rel="stylesheet" type="text/css" href="php echo
base_url() ?&gt;assets/css/style.css"&gt;
&lt;/head&gt;
&lt;body&gt;
  &lt;div id="wrapper"&gt;
    &lt;header&gt;
      &lt;hgroup&gt;
        &lt;h1&gt;WebKita.com&lt;/h1&gt;
        &lt;h3&gt;Membuat Template Sederhana Codeigniter &lt;/h3&gt;
      &lt;/hgroup&gt;
      &lt;nav&gt;
        &lt;ul&gt;
          &lt;li&gt;&lt;a href="<?php echo base_url().'.index.php/web'
?&gt;"&gt;Home&lt;/a&gt;&lt;/li&gt;
          &lt;li&gt;&lt;a href="<?php echo base_url().'.index.php/web/about'
?&gt;"&gt;About&lt;/a&gt;&lt;/li&gt;
        &lt;/ul&gt;
      &lt;/nav&gt;
      &lt;div class="clear"&gt;

    &lt;/div&gt;
  &lt;/header&gt;</pre
```

application/view/v_footer.php

```
<footer>
  <a href="http://github.com/k-sandi"> Developer Webkita.id</a>
</footer>
</div>
</body>
</html>
```

application/view/v_index.php

```
<section>
  <h1><?php echo $judul; ?></h1>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed
    do eiusmod tempor incididunt ut labore et dolore magna aliqua.
    Ut enim ad minim veniam, quis nostrud exercitation ullamco
    laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure
    dolor in reprehenderit in voluptate velit esse cillum dolore eu
    fugiat nulla pariatur. Excepteur sint occaecat cupidatat non
    proident, sunt in culpa qui officia deserunt mollit anim id est
    laborum. Lorem ipsum dolor sit amet, consectetur adipisicing elit,
    sed do eiusmod tempor incididunt ut labore et dolore magna
    aliqua. Ut enim ad minim veniam, quis nostrud exercitation
    ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis
    aute irure dolor in reprehenderit in voluptate velit esse cillum
    dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat
    non proident, sunt in culpa qui officia deserunt mollit anim id est
    laborum.
  </p>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed
    do eiusmod tempor incididunt ut labore et dolore magna aliqua.
    Ut enim ad minim veniam, quis nostrud exercitation ullamco
    laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure
    dolor in reprehenderit in voluptate velit esse cillum dolore eu
    fugiat nulla pariatur. Excepteur sint occaecat cupidatat non
    proident, sunt in culpa qui officia deserunt mollit anim id est
    laborum.
  </p>
</section>
```

Sekarang template ini sudah menjadi tiga view dan cara memanggilnya dengan cara memanggil view secara berurut. dari v_header, v_index dan kemudian v_footer pada controller. Berikut cara pemanggilannya:

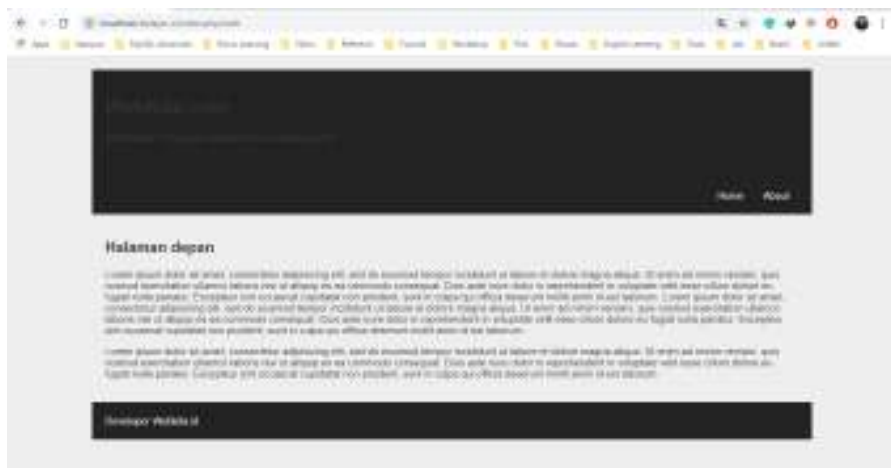
application/controller/web.php

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

class Web extends CI_Controller
{
    function __construct() {
        parent::__construct();
        $this->load->helper('url');
    }

    public function index() {
        $data['judul'] = "Halaman depan";
        $this->load->view('v_header',$data);
        $this->load->view('v_index',$data);
        $this->load->view('v_footer',$data);
    }
}
?>
```

Dan hasilnya juga akan sama, tetapi kelebihanannya kita bisa dengan mudah membuat halaman lain dan dengan memanggil v_header dan v_footer, hanya perlu mengganti v_index untuk halaman lainnya.



Gambar 5. 14 *Templating* sederhana: tampilan halaman web dinamis

Perhatikan lagi pada bagian *hyperlink* tepatnya pada menu *template* di atas (v_header.php) yang sudah kita setting untuk menuju method index dan method about.

```
<li><a href="<?php echo base_url().'index.php/web'
?>">Home</a></li>
<li><a href="<?php echo base_url().'index.php/web/about'
?>">About</a></li>
```

Sekarang buat sebuah view lagi dengan nama v_about.php

```
<section>
  <h1><?php echo $judul ?></h1>
  <p>
    halaman about yang bisa anda ubah sesuai keinginan. - Lorem
    ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
    tempor incididunt ut labore et dolore magna aliqua. Ut enim ad
    minim veniam, quis nostrud exercitation ullamco laboris nisi ut
    aliquip ex ea commodo consequat. Duis aute irure dolor in
    reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
    pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
    culpa qui officia deserunt mollit anim id est laborum.
  </p>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do
    eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut
    enim ad minim veniam, quis nostrud exercitation ullamco laboris
    nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in
    reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
    pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
    culpa qui officia deserunt mollit anim id est laborum. Lorem
    ipsum dolor sit amet, consectetur adipisicing elit, seddo eiusmod
    tempor incididunt ut labore et dolore magna aliqua.
  </p>
  <p>
    webkita.com
  </p>
</section>
```

Kemudian tambahkan lagi *method* about pada controller web untuk membuat halaman about.

application/controller/web.php

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

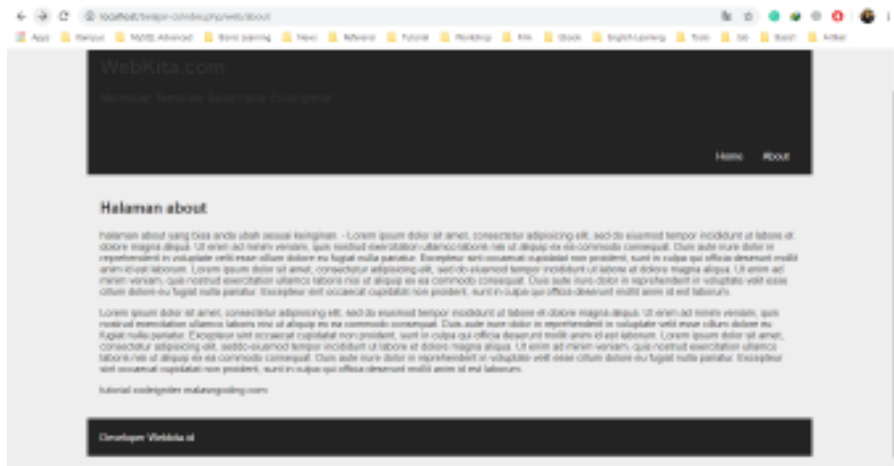
class Web extends CI_Controller
{
    function __construct() {
        parent::__construct();
        $this->load->helper('url');
    }

    public function index() {
        $data['judul'] = "Halaman depan";
        $this->load->view('v_header',$data);
        $this->load->view('v_index',$data);
        $this->load->view('v_footer',$data);
    }

    public function about() {
        $data['judul'] = "Halaman about";
        $this->load->view('v_header',$data);
        $this->load->view('v_about',$data);
        $this->load->view('v_footer',$data);
    }
}
?>
```

Selanjutnya jalankan lagi pada *browser*, Kemudia klik pada menu about atau bisa mengakses langsung halaman <http://localhost/belajar-ci/index.php/web/about>.

Berikut hasilnya:



Gambar 5. 15 *Templating* sederhana: tampilan halaman web about

5.6.4 Kesimpulan

Dengan membuat *template web* dengan *multiple view* atau memecahkan *template* menjadi beberapa bagian dan kemudian memanggilnya secara berurutan seperti contoh di atas yaitu dengan memanggil view header, index dan kemudian footer, maka akan memudahkan kita karena tidak perlu menuliskan syntax berulang-ulang. Misalnya, kita tidak perlu lagi membuat header khusus untuk halaman index dan tidak perlu juga membuat header khusus lagi untuk halaman about.

Pada dasarnya, header dan footer dapat di gunakan secara bersama-sama dan hanya bagian kontennya saja yang berubah-ubah sesuai keinginan dengan cara memanggilnya pada *method* controller web.

5.7 Fitur *Form Validation* Codeigniter

Pada sub bagian kali ini, buku ini menjelaskan tentang bagaimana membuat atau memanfaatkan *form validation* pada codeigniter. Terkadang kita memerlukan *form validation* dalam penginputan data, seperti misalnya kita menentukan *form* apa saja yang wajib di sisi, *form* dengan format *input* tertentu, membuat konfirmasi *password* dan sebagainya. Oleh karena itu, untuk membuat *form validation* pada codeigniter kita perlu memanggil atau membuka library **form_validation** pada codeigniter. berikut ini adalah cara membuat form validation pada codeigniter.

5.7.1 Membuat Form Validation

Untuk contoh membuat form validation pada codeigniter, kita akan membuat sebuah controller dengan nama **form.php**.

application/controller/form.php

```
<?php
class Form extends CI_Controller
{
    function __construct() {
        parent::__construct();
        $this->load->library('form_validation');
    }

    function index() {
        $this->load->view('v_form');
    }

    function aksi() {
        $this->form_validation->set_rules('nama','Nama','required');
        $this->form_validation->set_rules('email','Email','required');
        $this->form_validation->set_rules('konfir_email','Konfirmasi
        Email','required');
```

```
if($this->form_validation->run() != false) {  
    echo "Form validation oke"; }else{ $this->load-  
>view('v_form');  
    }  
}  
}  
?>
```

5.7.2

5.8 Membuat *Upload File* Dengan Codeigniter

5.9 Membuat *Download File* Dengan Codeigniter

5.10 Membuat *Library* Codeigniter Sendiri

5.11 Membuat URL Cantik Dengan Menghilangkan “index.php” Pada Codeigniter

5.12 Berkenalan Dengan Model, Membuat Hubungan dengan *Database* dan DBMS MySQL

5.13 Membuat CRUD Dengan Codeigniter

5.14 Membuat Pagination Dengan Codeigniter

5.15 Membuat Login Sederhana Dengan Codeigniter

1.1 PENGENALAN WEB PROGRAMMING

Web programming, Pada buku ini penulis akan memaparkan materi tentang teknologi tersebut dengan penerapan teknologi pembuatan aplikasi berbasis Web Programming.

1.1.1 Pengantar Web Programming

1.1.1.1 HTTP & WEB



Web Programming atau biasa disebut Pemrograman Web, Web programming terdiri dari 2 kata yaitu web dan programming, programming dapat di artikan sebagai proses pembuatan suatu program. Sedangkan web dapat di artika sebagai jaringan computer atau biasa disebut *website* yang terdiri dari situs jaringan internet yang menawarkan fitur dengan berbagai jenis seperti text, grafik, suara, Serta memelihara kode untuk membuat suatu pemrograman computer, kode ini ditulis dalam berbagai Bahasa pemrograman yang bertujuan untuk membuat suatu program yang dapat melakukan suaut perhitungan atau proses sessuai dengan keinginan pemrogram yang di akses melalui protocol HTTP.

Apa aitu HTTP? HTTP Merupakan protocol lapisan aplikasi (application layer) yang dikembangkan untuk membantu suatu proses transfer antar computer. Protokol ini berfungsi untuk melakukan transfer dokumen, file, gambar, dan video antar computer, Protokol HTTP menyediakan kumpulan suatu perintah didalam komunikasi antar suatu jaringan. Komunikasi tersebut

Fungsi HTTP yaitu mengatur Format dan bagaimana suatu data di transmisikan. HTTP juga berfungsi untuk mengatur bagaimana web server dan web browser saling terhubung dan memproses berbagai suatu perintah yang masuk. Fungsi lain dari HTTP ialah mengamankan data dari suatu pencurian dan hacker. Hal ini ditandai dengan munculnya HTTPS (Hypertext transfer Protocol Secure).

Jadi secara singkat untuk Web Programming adalah suatu proses pembuatan program yang memiliki output disajikan oleh web browser yang di akses melalui HTTP dan Bahasa yang digunakan adalah Bahasa HTML.

1.1.1.2 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 Engineering 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 dirilis HTML4 pada akhir tahun tersebut yaitu 1997 dan 1998.

Secara Umum Pemrograman Web dibagi menjadi 2 bagian yaitu :

1. Client Side Scripting (CSS)

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 (SSS)

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.

Contoh Script Paling Populer:

- 1.PHP
- 2.ASP
- 3.JSP

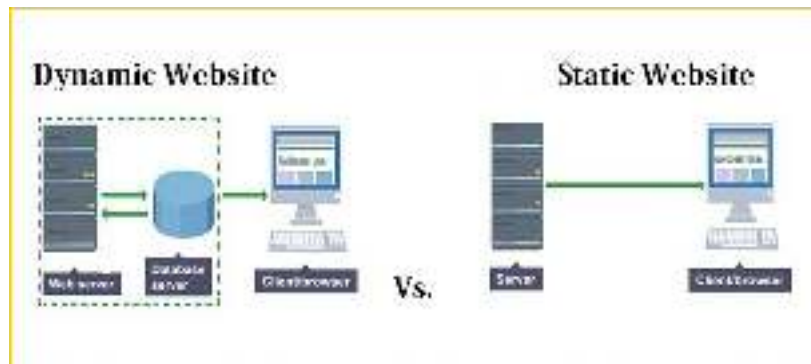
1.1.1.3 HUBUNGAN PHP DAN HTML

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 dibubuhi program php harus diganti ekstensi-nya menjadi .php3 atau php. Php merupakan bahasa pemrograman web yang bersifat server-side HTML=embedded scripting, di mana script-nya menyatu dengan HTML dan berada di 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).

1.1.1.4 Maksud 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 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.

1.1.1.5 BASIS DATA

Secara sederhana *Database* (basis data) bisa artikan 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 pemanipulasian 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

1.1.2 SEJARAH

Cabang ilmu Pemrograman cukup luas, dan erat kaitannya dengan disiplin ilmu yang lainnya. Hal ini bisa dilihat dari berbagai aplikasi yang merupakan hasil kombinasi dari berbagai ilmu.

1.1.2.1 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.

1.1.2.2 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 perilis 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 pemrograman PHP saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.

1.1.2.3 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.

1.1.3 BAHASA PEMROGRAMAN

1.1.3.1 Pemrograman Web



Seperti yang sudah di sebutkan bahwa ada 2 kategori didalam pemrograman web, yaitu pemrograman Server Side dan Client side. Dalam pemrograman server side terdapat perintah program yang dijalankan pada server web lalu hasil akan dikirimkan dalam bentuk HTML biasa. Adapun pada Client Side program dijalankan pada browser web dari sever kemudian akan dijalankan oleh browser yang bersangkutan

A. Mengenal Script Clinet Side

Program web yang tergolong dalam Client Side Seperti Java Script, VB , HTML dll. Hasil dari parshing script pemrograman Client Side yang berupa Html dari web server dapat dilihat dengan menu view >

source code dapat terlihat bahwa script program yang ditulis ditampilkan pada halaman source code.

B. Java Script

JavaScript diperkenalkan pertama kali oleh Netscape di tahun 1995. pada mulanya Bahasa ini memiliki nama Livescript dan memiliki fungsi sebagai Bahasa sederhana untuk browser NetScope Navigator2. Pada masa pertama kali rilis Bahasa ini memiliki sangat banyak kritikan karena kurang aman, tidak ada pesan kesalahan dari setiap script program yang ditampilkan. Kemudian dengan kerja sama dengan Netscape dengan SUN (pengembang Bahasa pemrograman) maka Netscape merubah nama menjadi javascript pada tanggal 4 desember 1995.

Java Script adalah Bahasa pemrograman yang sederhana karena Bahasa yang digunakan tidak dapat membuat aplikasi, java script dapat dengan mudah kita temukan dalam suatu program yang sudah interaktif . program java script dituliskan pada file HTML. Dengan kata lain tidak perlu menuliskan program java script pada file terpisah. Bahasa ini adalah Bahasa pemrograman untuk memberikan kemampuan tambahan terhadap Bahasa HTML. Dengan mengizinkan menjalankan perintah dari sisi Klien. Yang artinya disisi browser bukan disisi server web javascript bergantung kepada browser memanggil pada halaman web yang berisi script java script tidak memerlukan penerjemah khusus untuk menjalankan script tersebut.

1.1.3.2 Keistimewaan MySQL

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

a. Portability

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

b. 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.

c. 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.

d. Performance Tuning

MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu

e. 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.

f. Command dan Functions

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

g. Security

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

h. 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.

i. Connectivity

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

j. Localisation

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

k. Interface

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

l. Clients dan Tools

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

m. Struktur Tabel

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

1.1.3.3 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

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 mahasiswa 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 proyek 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 kodenya. 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 proyek 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 licensi atas PHP yang dimiliki PHP Group. Dengan kata lain selama pemakai program PHP tidak mengakui produk PHP adalah buaatannya maka perjual belian program yang menggunakan PHP diperbolehkan tanpa harus membayar licensi apapun.

7. Perkembangan Pesa

Karena sifat PHP yang open source, banyak sekali bermunculan proyek proyek 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

1.1.3.4 Definisi 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 :

- a. Ukuran file lebih kecil
- b. Load file lebih cepat
- c. Dapat berkolaborasi dengan Javascript
- d. Pasangan setia XHTML
- e. Menghemat pekerjaan tentunya, dimana hanya membuat 1 halaman CSS.
- f. Mudah mengganti-ganti tampilan dengan hanya merubah file CSS nya saja.

1.1.3.5 Definisi 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.

1.1.3.5.1 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.

1.1.3.6 Definisi 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 mengizinkan 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 ekuivalen 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.

1.1.3.7 Definisi 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.

1.1.3.7.1 Hal-hal yang dapat dilakukan dengan python.

Sebelum belajar Python 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 kompleks.

1.1.3.7.2 Manfaat 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.

1.1.3.7.3 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 __name__ == '__main__':
    print("Indentation is the key to Python")
```

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 __name__ == '__main__':
print("Indentation is the key to Python")
```

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 = 5; y = "Budi"
```

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 = 5; y = "Budi"; z = 5.5
```

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:

```
x = True; y = 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 pendeklarasiannya, 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 # int; y = 3.14 # float; z = 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)) # int; print(type(y)) # float; print(type(z)) # complex
```

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" # String dengan tanda kutip ganda  
y = 'Budi' # String dengan tanda kutip tunggal
```

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.

```
su = "Hello, World!"
print(su[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), perbandingan (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	Kategori	Contoh
+	Penjumlahan	x + y
-	Pengurangan	x - y
*	Pengkalangan	x * y
/	Pembagian	x / y
%	Modulus	x % y
**	Pengangkatan (pangkat)	x ** y
//	Pembagian bulat	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	Notasi	Notasi
+	$x + y$	$x + y$
-	$x - y$	$x - y$
*	$x * y$	$x * y$
/	x / y	x / y
%	$x \% y$	$x \% y$
**	$x ** y$	$x ** y$
~	$\sim x$	$\sim x$
&	$x \& y$	$x \& y$
	$x y$	$x y$
^	$x ^ y$	$x ^ y$
~	$\sim x$	$\sim x$
&&	$x \&\& y$	$x \&\& y$
	$x y$	$x y$
!=	$x != y$	$x != y$
==	$x == y$	$x == y$

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	Notasi	Notasi
==	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 <= b`), lebih besar dari (`a > b`), lebih besar atau sama dengan (`a >= 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:

```
if a < b:
    print('a lebih kecil dari b')
else:
    print('a lebih besar dari b')
```

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.

```
if a < b:
    print('a lebih kecil dari b')
elif a <= b:
    print('a lebih besar atau sama dengan b')
else:
    print('a lebih kecil dari b')
```

1.1.3.8 Definisi 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 Python. 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.

1.1.3.9 Definisi 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 Language.

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 distandarisasi 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, Struktur Utama dari suatu program Cobol terdiri dari 4 divisi yaitu:

- IDENTIFICATION DIVISION
- ENVIRONMENT DIVISION
- DATA DIVISION
- 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.

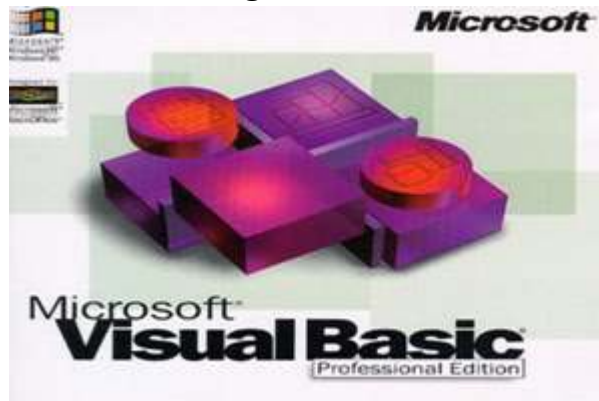
1.1.3.9.1 Yang dapat dikerjakan COBOL

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

1.1.3.9.2 Keuntungan Bahasa pemrograman cobol

- Program COBOL dibuat dalam instruksi bahasa inggris, sehingga lebih mudah dipelajari dan dibuat .
- program COBOL sesuai untuk pengolahan data, yang banyak diterapkan pada permasalahan bisnis.
- program COBOL sifatnya standard, sehingga dapat dipergunakan pada computer – computer yang berbeda, tanpa banyak perbedaan.
- struktur program COBOL jelas, sehingga dapat dimengerti oleh orang seperti akuntan, auditor atau manajer – manajer yang hanya mempunyai pengetahuan pengolahan data yang sedikit.
- COBOL menyediakan fasilitas listing program , bila mana perlu dapat diperiksa oleh orang lain selain programmernya.
- mudah didokumentasikan dan dikembangkan bilamana perlu.

1.1.3.10 Definisi Bahasa Pemrograman Microsoft 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.

1.1.3.10.1 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 pilihan yang tersedia saat objek tersebut diklik.
- g. Commond 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.