

MJ-515 PEMROGRAMAN WEB 2019 – 2020

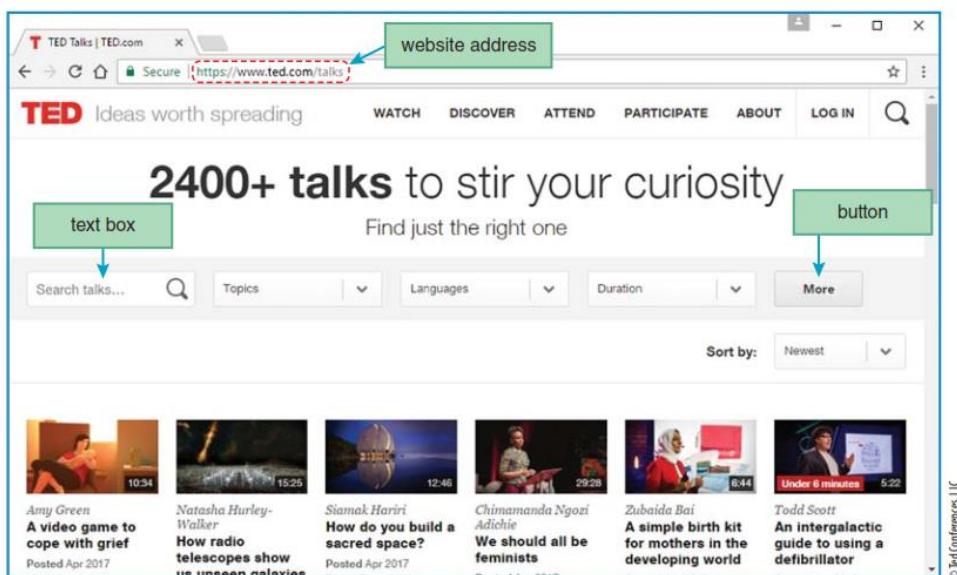
POLITEKNIK NEGERI BATAM

Disusun oleh Supardianto, S.S.T., M.Eng.

I PEMROGRAMAN WEB

Pengantar

Situs web (bahasa Inggris: website) adalah sekumpulan halaman web yang saling berhubungan yang umumnya berada pada server yang sama berisikan kumpulan informasi yang disediakan secara perorangan, kelompok, atau organisasi. Penemu situs web adalah Sir Timothy John "Tim" Berners-Lee, sedangkan situs web yang tersambung dengan jaringan pertamakali muncul pada tahun 1991. Maksud dari Tim ketika merancang situs web adalah untuk memudahkan tukar menukar dan memperbarui informasi pada sesama peneliti di tempat ia bekerja. Pada tanggal 30 April 1993, CERN (tempat di mana Tim bekerja) mengumumkan bahwa WWW dapat digunakan secara gratis oleh publik.



Gambar 1. Contoh sebuah website

Secara garis besar, situs web bisa digolongkan menjadi 2 bagian yaitu:

1. Situs web statis

Situs web statis merupakan situs web yang memiliki isi tidak dimaksudkan untuk diperbarui secara berkala sehingga pengaturan ataupun pemutakhiran isi atas situs web tersebut dilakukan secara manual.

2. Situs web dinamis

Situs web dinamis merupakan situs web yang secara spesifik didisain agar isi yang terdapat dalam situs tersebut dapat diperbarui secara berkala dengan mudah. Sesuai dengan namanya, isi yang terkadung dalam situs web ini umumnya akan berubah setelah melewati satu periode tertentu.

Web Programming Saat Ini

Web Programming memiliki beberapa manfaat seperti:

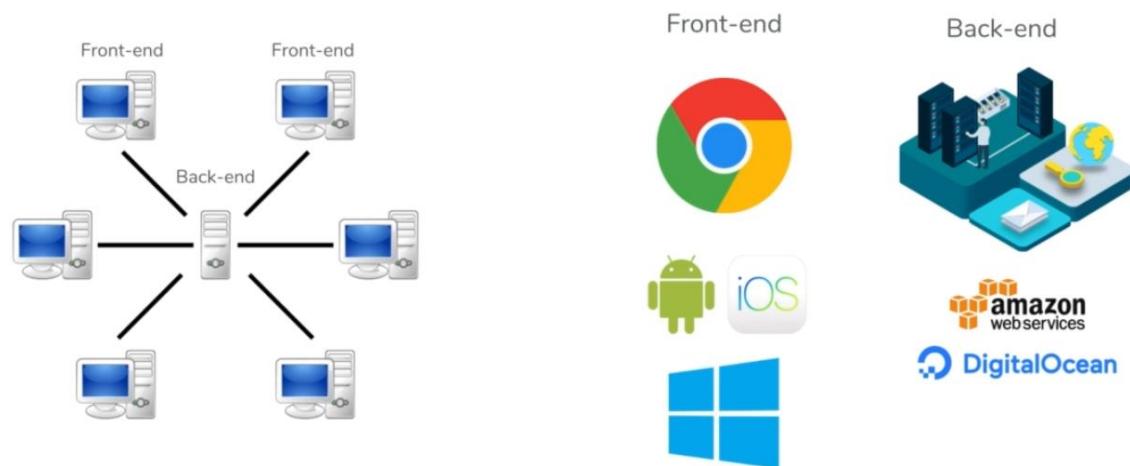
1. Mengembangkan cara berpikir sistematis.
Secara singkat, sistematis merupakan salah satu unsur dalam proses perencanaan dimana produk perencanaan dihasilkan melalui tahapan tahapan yang teratur.
2. Membuat solusi dari suatu masalah dengan menggunakan teknologi.
Banyak masalah saat ini yang hampir semua dapat diselesaikan dengan menggunakan teknologi. Hadirnya aplikasi seperti Go-Jek telah menyelesaikan masalah seperti layanan transportasi, dll, atau penggunaan media website dihadirkan sebagai media pendukung untuk pembelajaran jarak jauh seperti hadir nya aplikasi Ruang Guru.

Mengapa Web Programming?

1. Teknologi yang terus berkembang sampai saat ini
2. Masih dipakai oleh banyak orang
3. Mudah untuk dipelajari oleh orang awam.

Konsep client dan Server

Client dan server adalah salah satu hal dalam web development. Pada web development terbagi menjadi 2, yaitu Front-end dan Back-end.



Gambar 2. Konsep Client-server dengan Front-end dan Backend

1. Front-end

Front-End adalah sebuah bagian dari website yang menyuguhkan tampilan kepada user. Bagian ini dibangun menggunakan HTML (HyperText Markup Language), CSS (Cascading Style Sheets), dan JavaScript sehingga sebuah URL bisa berfungsi dan menampilkan website yang baik. Orang yang bekerja di baliknya alias Front-End developer bertugas mengembangkan semua komponen visual pada situs web dan menjaga tampilan antarmuka. Front-End developer juga bertugas mengerjakan layout tampilan muka alias User Interface (UI) suatu website atau aplikasi untuk tampilan yang lebih menarik.

2. Back-end

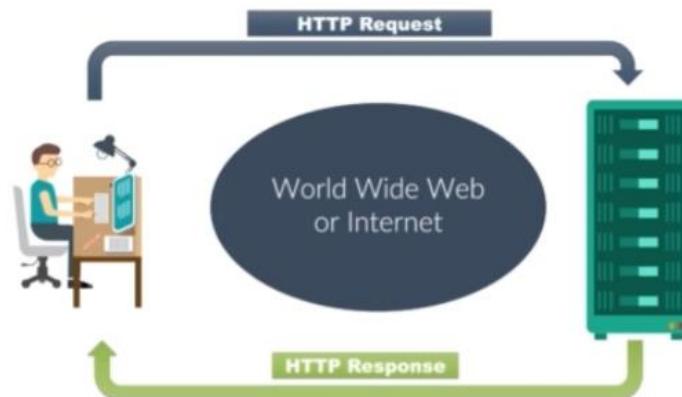
Di balik sebuah website yang bagus, ada program Back-End yang menopang semua fitur agar berjalan dengan semestinya. Jika Front-End fokus pada tampilan website, maka Back-End fokus pada fungsionalitas. Unsur-unsur yang ada dalam program Back-End biasanya berupa database, bahasa pemrograman, server website, dan service web. Sederhananya, pengembang Back-End

bertugas di belakang layar memastikan fungsi fitur berjalan lancar menuju sistem atau database yang digunakan oleh situs tersebut.

Perbedaan antara Client side dan Server Side

Pada client side semua proses terjadi di sisi pengguna. Jadi, ketika seseorang berkunjung ke sebuah web, komputernya akan mendownload data/script yang bersifat client-side di web tersebut. Client-side script mempengaruhi berat-tidaknya loading sebuah website yang tergantung pada kecepatan loading & spesifikasi komputer serta koneksi internet penggunanya. Client-side script meliputi kode-kode yang ditampilkan ketika anda mengklik kanan pada sebuah halaman web dan melihat sumber halaman (View Page Source). Contoh dari Client-side Script adalah HTML, CSS, JavaScript dan XML.

Sedangkan untuk server side, setiap kali sebuah web dikunjungi, server akan mengirimkan data-data yang diminta dari database yang kemudian akan ditampilkan di web. Server-side Script biasanya hanya ada pada web dinamis saja. Server-side Script memengaruhi berat-tidaknya loading sebuah website bergantung pada kecepatan & spesifikasi komputer server. Kode-kode Server-side tidak bisa kita lihat karena sifatnya yang rahasia untuk Client. Contoh dari Server-side Script adalah PHP, ASP, ASP.Net



Gambar 3. Contoh request dan response pada HTTP

Konsep Bahasa Pemrograman

Bahasa pemrograman adalah instruksi standar untuk memerintah komputer. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan/diteruskan, dan jenis langkah apa yang akan diambil dalam berbagai situasi secara persis. *Bahasa pemrograman web* adalah bahasa pemrograman yang digunakan untuk membangun aplikasi berbasis web. Ada 2 tipe bahasa pemrograman web, yaitu client side language dan server side language.

1. **Client side language**
 - a. HTML, CSS, Javascript
2. **Server side language**
 - a. PHP, Java, PythonPerl, Node.JS, Ruby

Selain client dan server side language maka ada beberapa istilah lain seperti Library, CMS, dan Framework yang bukan merupakan bahasa pemrograman. Contoh dari library, cms dan framework:

1. **Library**; jQuery, React, Vue, dan Angular
2. **Framework**; Laravel, Codeigniter, Lumen, ExpressJS, dan Spring
3. **CMS**; Wordpress, Drupal, dan Ghost

Web Browser

Web browser adalah software yang digunakan untuk menjelajahi internet atau untuk mencari informasi dari suatu laman website. Web browser juga digunakan untuk merepresentasikan dokumen HTML ke dalam bentuk Visual. Contoh; Google Chrome, Mozilla Firefox, Safari, dan Opera.

Web Standard (oleh W3C)

sebuah konsorsium yang bekerja untuk mengembangkan standar-standar untuk world wide web. Spesifikasi beberapa teknologi paling utama yang digunakan sebagai basis paling utama situs, seperti URL (Uniform Resource Locator), HTTP (Hypertext Transfer Protocol), serta HTML (HyperText Markup Language) diperkembangkan serta ditata oleh badan ini.

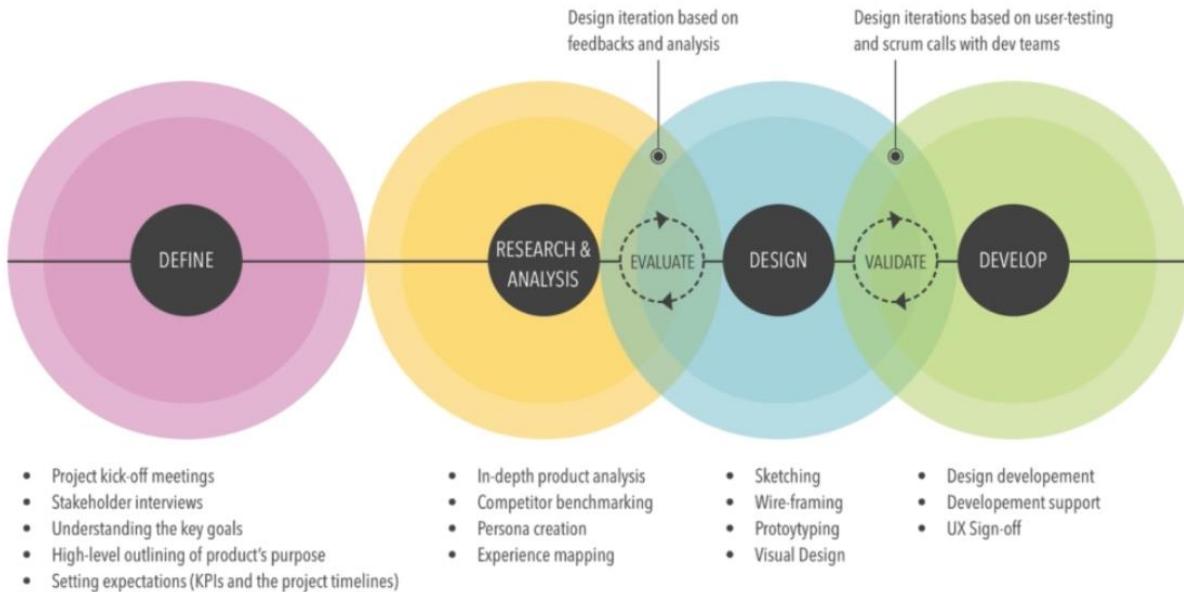
W3C merupakan suatu badan yang membuat standard untuk website agar sesuai dengan kualitas yang diharapkan.

1. Memiliki HTML, CSS, dan Javascript yang valid sehingga dapat dikenali oleh browser
2. Memenuhi standar aksesibilitas, sehingga website dapat diakses oleh orang dengan penyandang disabilitas
3. Memiliki metadata yang valid, sehingga dapat diindeks oleh mesin pencari
4. Memiliki karakter encoding yang sesuai, sehingga karakter spesial akan tampil dengan baik dan tidak merusak tampilan halaman.

Design Best Practice

Sebelum mendekvelop sebuah website, maka perlu untuk mendefinisikan dahulu beberapa hal seperti

1. Apa tujuan dari website tersebut?
2. Siapa Saja penggunanya?
3. Target bisnis yang akan dicapai?



Gambar 4. Contoh Best Practice (sumber: www.peepaldesign.com)

Design Best Practice yang lain nya terdiri dari 3 poin, yaitu:

1. Mobile First, bisa diakses langsung di mobile

2. User Centered, user tidak bingung ketika menggunakan aplikasi
3. Usability Testing, test dimana kita langsung meminta user untuk melakukan ujicoba

Target yang ingin dicapai, terdiri dari 4 poin yaitu:

1. Pengunjung betah di website
2. Meningkatkan intensitas percakapan
3. Meningkatkan nilai kepuasan
4. Membuat pengunjung untuk datang lagi.

Development Tools

1. **Laptop** atau komputer dengan OS seperti windows, linux, atau mac os
2. **Text Editor**, seperti sublime text, Brackets, Atom, atau Visual Studio Code.
3. **Domain dan Server atau Hosting**, Amazon web services, DigitalOcean, CloudHost, dan Dewaweb
4. **Deployment Tools**, GIT, FTP Client, atau Cloud Apps (Heroku, Now, Cloud Foundry)
5. **Local Server Tools**, Xampp, Ampps, atau Laragon.

II KONSEP DASAR HTML

Pengertian HTML

HTML adalah singkatan dari *Hypertext Markup Language*. HTML memungkinkan seorang user untuk membuat dan menyusun bagian paragraf, heading, link atau tautan, dan blockquote untuk halaman web dan aplikasi.

Ketika bekerja dengan HTML, kami menggunakan struktur kode yang sederhana (tag dan attribute) untuk mark up halaman website. Misalnya, kami membuat sebuah paragraf dengan menempatkan enclosed text di antara tag pembuka `<p>` dan tag penutup `</p>`.



```
1 <html>
2   <head>
3     <title>Belajar HTML Dasar</title>
4   </head>
5   <body>
6     <h1>Halo Mahasiswa</h1>
7     <p>Ini halaman pertama saya</p>
8   </body>
9 </html>
```

Gambar 5. Contoh HTML

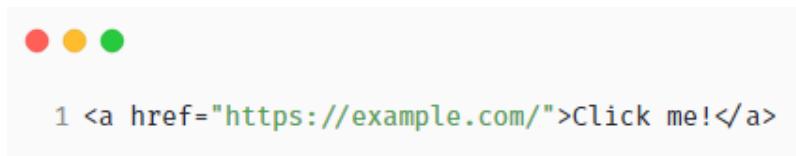
Tag HTML: block-level dan inline tags.

Elemen *block-level* memakai semua space yang tersedia dan selalu membuat line baru di dalam dokumen. Contoh dari tag block adalah heading dan paragraf.

Tiga tag *block-level* yang harus dimiliki oleh setiap dokumen HTML adalah <html>, <head>, dan <body>.

1. Tag <html></html> adalah elemen level tertinggi yang menyertakan setiap halaman HTML.
2. Tag <head></head> menyimpan informasi meta, seperti judul dan charset halaman.
3. Tag <body></body> melampirkan semua konten yang muncul pada suatu halaman.

Elemen *inline* hanya memakai space sesuai dengan kebutuhannya dan tidak membuat line baru di halaman. Biasanya elemen ini akan memformat isi konten dari elemen block-level. Contoh dari tag inline adalah link dan emphasized strings. Sebagian besar tag inline digunakan untuk memformat teks. Sebagai contoh, tag akan render elemen ke format bold, sedangkan tag akan ditampilkan dalam format italic.



Gambar 6. Contoh elemen tag inline

Kelebihan dan Kekurangan HTML

Sama seperti hal teknis lainnya dalam dunia web, HTML juga punya kelebihan dan kekurangannya.

Kelebihan:

1. Bahasa yang digunakan secara luas dan memiliki banyak sumber serta komunitas yang besar.
2. Dijalankan secara alami di setiap web browser.
3. Memiliki learning curve yang mudah.
4. Open-source dan sepenuhnya gratis.
5. Bahasa markup yang rapi dan konsisten.
6. Standard web yang resmi di-maintain oleh World Wide Web Consortium (W3C).
7. Mudah diintegrasikan dengan bahasa backend, seperti PHP dan Node.js.

Kekurangan:

1. Paling sering digunakan untuk halaman web statis. Untuk fitur dinamis, bisa menggunakan JavaScript atau bahasa backend, seperti PHP.
2. HTML tidak memungkinkan user untuk menjalankan logic. Alhasil, semua halaman web harus dibuat terpisah meskipun menggunakan elemen yang sama, seperti header dan footer.
3. Fitur-fitur baru tidak bisa digunakan secara cepat di sebagian browser.
4. Terkadang perilaku browser susah untuk diprediksi (misalnya, browser lama tidak selalu bisa render tag yang lebih baru).

Pengertian CSS

CSS adalah bahasa *Cascading Style Sheet* dan biasanya digunakan untuk mengatur tampilan elemen yang tertulis dalam bahasa markup, seperti HTML. CSS berfungsi untuk memisahkan konten dari tampilan visualnya di situs.

1. Format yang ditulis untuk memberi pengaturan tampilan
2. Tanpa menggunakan tag HTML
3. Memisahkan antara konten dan style

4. Satu halaman HTML dapat diganti dengan stylennya dengan mengganti file CSS

III DASAR-DASAR HTML DAN CSS

Element Pada HTML

Sebuah elemen pada HTML diawali oleh tag dan diakhiri oleh tag, contoh <tagname> Content goes here...</tagname>. TAG sederhananya adalah kode-kode tertentu yang menjelaskan bahwa kode tersebut adalah HTML dan bisa diterjemahkan oleh browser dengan tampilan tertentu. Misalnya tadi, untuk membuat sebuah tulisan tebal, maka kita bisa menggunakan TAG STRONG, atau TAG B. Untuk membuat paragraf kita bisa menggunakan TAG P. Dan masih banyak TAG HTML lainnya, tentunya dengan fungsi yang berbeda-beda.

Elemen HTML Bersarang

Elemen HTML sangat mungkin untuk bersarang (nested), yaitu memiliki elemen lain di dalamnya. Perhatikan kode berikut:



```
1 <!DOCTYPE html>
2 <html>
3   <body>
4     <h1>Sebuah Heading</h1>
5     <p>Sebuah Paragraf dengan <strong>tulisan tebal</strong> di dalamnya.</p>
6   </body>
7 </html>
```

Gambar 7. Contoh Element HTML Bersarang

Pada contoh kode HTML di atas, kita memiliki 4 buah elemen HTML, yaitu elemen HTML, BODY, H1 dan P. Di dalam elemen HTML terdapat elemen BODY. Dan di dalam elemen BODY terdapat 2 elemen lainnya yaitu elemen H1 dan P. Dan di dalam elemen P pun terdapat elemen lainnya yaitu elemen STRONG. Itulah yang disebut dengan elemen bersarang, artinya di dalam konten elemennya mengandung elemen lainnya.

HTML tidak case-sensitif, dalam artian penulisan <p> dianggap sama dengan <P>. Pada awal kemunculan HTML, programmer web umumnya menggunakan huruf besar untuk seluruh tag agar membedakan

dengan text yang berupa isi dari web. Namun varian HTML, xHTML mewajibkan huruf kecil untuk semua tag.

Terdapat pengecualian beberapa tag yang tidak berpasangan, seperti `
` untuk break (pindah baris) atau `<hr>` untuk horizontal line (garis horizontal). Tag ini dikenal juga dengan sebutan self closing tag atau void tag, untuk penulisannya bisa ditulis dengan `
`, maupun `
`.

Atribut Pada HTML

Atribut adalah karakteristik tambahan pada suatu TAG. Misalnya seperti yang sebelumnya di informasikan bahwa untuk membuat paragraf bisa menggunakan TAG P. Secara default paragraf yang akan ditampilkan oleh TAG P adalah rata kiri. Tapi jika kita ingin membuatnya rata tengah, kita bisa menambahkan atribut ALIGN dengan value CENTER ke dalam TAG P tersebut. Misalnya seperti berikut:

```
<p align="center">Untuk menampilkan tulisan rata tengah</p>
```

Heading HTML

Heading atau lebih tepatnya TAG Heading merupakan TAG khusus yang disediakan oleh HTML untuk membuat sebuah judul pada teks dalam sebuah halaman web. TAG Heading secara default ditampilkan oleh browser dengan ukuran lebih besar dan lebih tebal (bold) dari teks biasa. Tidak hanya sekedar untuk memberikan penampilan yang lebih tegas dan penekanan, TAG Heading juga sangat penting dalam kaitannya dengan teknik Search Engine Optimization (SEO).

TAG Heading dalam HTML terdiri dari 6 tingkatan berdasarkan tingkat penting dan ukurannya. TAG Heading tersebut adalah `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` dan `<h6>`. TAG H1 memiliki ukuran paling besar, sedangkan TAG h6 memiliki ukuran paling kecil.

Biasanya H1 hanya digunakan satu kali dalam sebuah halaman web, yaitu sebagai judul utama (judul besar). H2, H3 dan seterusnya bisa kita gunakan beberapa kali sebagai sub judul.



```

1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <title>Penggunaan TAG Heading HTML</title>
5   </head>
6   <body>
7     <h1>Judul Utama</h1>
8     <p>Isi dari Judul Utama</p>
9     <h2>Sub Judul</h2>
10    <p>Isi Sub Judul</p>
11    <h2>Sub Judul</h2>
12    <p>Isi dari Sub Judul</p>
13    <h3>Sub dari Sub Judul</h3>
14    <p>Isi dari Sub dari Sub Judul</p>
15  </body>
16 </html>

```

Isi dari Judul Utama

Sub Judul

Isi Sub Judul

Sub Judul

Isi dari Sub Judul

Sub dari Sub Judul

Isi dari Sub dari Sub Judul

Gambar 8. Contoh Penulisan Heading dan Hasilnya

Paragraf Pada HTML

Untuk membuat paragraf di dalam HTML tidak sesederhana ketika kita membuatnya di software pengolah kata seperti MS. Word, tidak cukup hanya dengan memberikan jarak dengan menekan ENTER. Karena pada dasarnya HTML akan mengabaikan spasi. Oleh karena itu untuk membuat paragraf di HTML kita membutuhkan tag khusus, yaitu TAG P. Berikut contoh penulisan yang disimpan dengan nama index.html:



```

1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <title>Paragraf di HTML</title>
5   </head>
6   <body>
7     <p>Tulisan di baris pertama.</p>
8     <p>Tulisan di baris kedua.</p>
9   </body>
10 </html>

```

Tulisan di baris pertama.

Tulisan di baris kedua.

Gambar 9. Contoh penulisan paragraf dan hasilnya.

HTML Line Breaks

Selain menggunakan TAG P, kita juga bisa membuat paragraf dengan TAG BREAK atau BR. Berikut contohnya:



```
1 <!DOCTYPE HTML>
2 <html>
3     <head>
4         <title>Paragraf di HTML</title>
5     </head>
6     <body>
7         Tulisan di baris pertama.<br>
8         Tulisan di baris kedua.<br><br>
9         Tulisan di baris ketiga.
10    </body>
11 </html>
```

Tulisan di baris pertama.
Tulisan di baris kedua.

Tulisan di baris ketiga.

Gambar 10. Contoh penulisan Line Breaks dan hasilnya

Membuat List

Dalam HTML elemen list terdiri dari 2 jenis yaitu Ordered List yaitu menggunakan urutan karakter seperti angka atau huruf dan menggunakan `` tag, sedangkan Unordered list menggunakan urutan simbol dan menggunakan `` tag. Letak perbedaan nya hanya pada tag yang digunakan.

An Unordered List:

- Item
- Item
- Item
- Item

An Ordered List:

1. First item
2. Second item
3. Third item
4. Fourth item

Gambar 12. Contoh Unordered List dan Ordered List



```
1 <ul>
2     <li>Coffee</li>
3     <li>Tea</li>
4     <li>Milk</li>
5 </ul>
```

```
1 <ol>
2     <li>Coffee</li>
3     <li>Tea</li>
4     <li>Milk</li>
5 </ol>
```

Gambar 11. Contoh penluisan Unordered List dan Ordered List

Membuat Link

Tujuan kata Hypertext dari HTML adalah membuat sebuah text yang ketika di-klik akan pindah ke halaman lainnya. HTML menggunakan tag `<a>` untuk keperluan ini. Link ditulis dengan `<a>` yang merupakan singkatan dari anchor (jangkar). Setiap tag `<a>` setidaknya memiliki sebuah atribut href. Dimana href berisi alamat yang dituju (href adalah singkatan dari hypertext reference).

Agar lebih jelas, kita akan lihat menggunakan contoh. Silahkan buka text editor dan buat kode seperti dibawah ini.

```
● ● ●  
1 <!DOCTYPE html>  
2 <html>  
3 <head>  
4   <title>Penggunaan Tag Link </title>  
5 </head>  
6 <body>  
7   <h1>Belajar Tag Link</h1>  
8   <p>Klik untuk ke Google</p>  
9 </body>  
10 </html>
```

Belajar Tag Link

Klik untuk ke [Google](https://www.google.com)

Gambar 12. Contoh penerapan tag link dan hasilnya

Menambahkan Gambar

Tag Image digunakan untuk menampilkan gambar kedalam halaman web, menggunakan . Atribut src adalah singkatan dari source, merupakan atribut yang berisi alamat dari gambar yang akan ditampilkan. Pada contoh dibawah ini menggunakan sebuah gambar koala.jpg yang berada dalam satu folder dengan halaman HTML saat ini, dan disimpan dengan nama img.html :

```
● ● ●  
1 <!DOCTYPE html>  
2 <html>  
3 <head>  
4   <title>Penggunaan Tag Image</title>  
5 </head>  
6 <body>  
7   <h1>Belajar Tag Gambar</h1>  
8     
10 </body>  
11 </html>
```

Belajar Tag Gambar



Gambar 13. Contoh menambahkan gambar dan hasilnya.

Atribut width dan height dalam tag

Atribut lainnya membolehkan kita untuk menentukan besar dari gambar yang ditampilkan, yaitu width dan height. Untuk mempertahankan proporsi gambar, namun tetap membuat gambar menjadi besar/kecil, cantumkan hanya salah satu atribut saja (width saja atau height saja, namun tidak keduanya). Misalkan jika kita menetapkan atribut width=300px (tanpa mencantumkan height), maka web browser akan menampilkan gambar dengan lebar 300px, dan menghitung secara otomatis tinggi gambar agar gambar tetap proporsional.

Membuat Table

Untuk membuat tabel pada html maka tag yang digunakan adalah <table> tag. Setiap baris pada tabel di definisikan dengan menggunakan <tr> tag, tabel header dengan menggunakan <th> dan table data dengan menggunakan <td> tag.

1. Tag <table> digunakan untuk memulai tabel
2. Tag <tr> adalah singkatan dari table row, digunakan untuk membuat baris dari tabel.
3. Tag <td> adalah singkatan dari table data, digunakan untuk menginput data ke tabel.

Sebelum standar CSS diimplementasikan ke dalam semua browser, pada sekitar tahun 2000-an kebanyakan programmer dan desainer web menggunakan tabel untuk mengatur tampilan web. Membuat menu ada di atas dan sisi kanan web bisa diakali dengan menjadikan halaman web sebagai sebuah tabel yang besar. Hal ini sebenarnya tidak salah, namun akan membuat pengkodean HTML menjadi rumit. Saat ini anda masih bisa menggunakan ide tersebut, namun sangat disarankan menggunakan CSS untuk mengatur tampilan halaman web.

The screenshot shows a code editor window with the title 'Basic HTML Table'. On the left, the HTML code is displayed:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h2>Basic HTML Table</h2>
5 <table border="1">
6   <tr>
7     <th>Firstname</th>
8     <th>Lastname</th>
9     <th>Age</th>
10    </tr>
11    <tr>
12      <td>Jill</td>
13      <td>Smith</td>
14      <td>50</td>
15    </tr>
16  </table>
17 </body>
18 </html>
```

On the right, the rendered output is shown as a table with three columns: Firstname, Lastname, and Age. The data row contains Jill, Smith, and 50 respectively.

Gambar 14. Contoh penulisan tabel dan hasil nya

Membuat Form di HTML (tag form)

Form biasanya digunakan untuk mengumpulkan data dari pengunjung web. Mulai dari form untuk login, form kontak, form untuk pendaftaran user, bahkan untuk mengirimkan data antar halaman web. Penggunaan form hanya menggunakan HTML saja tidak akan terlalu berguna. Form biasanya hanya berupa interface yang disediakan untuk mengumpulkan data dari user, dan akan diproses dengan bahasa pemrograman web seperti JavaScript atau PHP, dan disimpan di dalam tabel MySQL. Tag dasar yang akan digunakan untuk membuat form di HTML adalah tag *form*, *input*, *textarea*, *select* dan *option*.

Sebuah form dalam HTML harus berada di dalam tag *form*, yang diawali dengan <*form*> dan diakhiri dengan </*form*>. Tag *form* akan membutuhkan beberapa atribut untuk dapat berfungsi dengan seharusnya.

1. Atribut pertama adalah action, yang berfungsi untuk menjelaskan kemana data form akan dikirimkan. Biasanya nilai dari atribut action ini adalah alamat dari sebuah halaman PHP yang digunakan untuk memproses isi data form.
2. Atribut kedua adalah method, yang berfungsi untuk menjelaskan bagaimana data isian form akan dikirim oleh web browser. Nilai dari atribut method ini bisa berupa get atau post.



```

1 <form action="prosesdata.php" method="post">
2   ... isi form ...
3 </form>

```

Gambar 15. Contoh penulisan form

Mengenal tag <input>

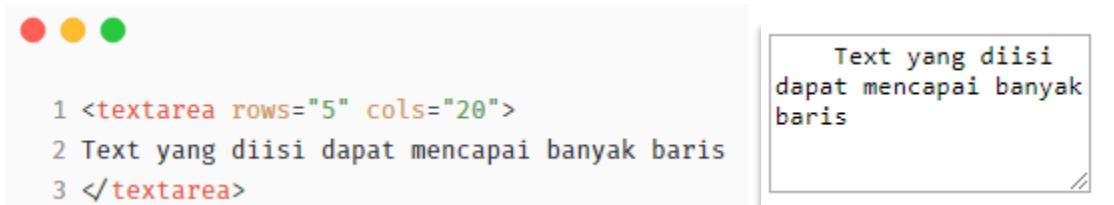
Tag input merupakan tag paling banyak digunakan di dalam form dan memiliki banyak bentuk, mulai dari isian text biasa, text password, checkbox, radio, sampai dengan tombol submit, semuanya dalam bentuk tag <input>.

Bentuk-bentuk dari keluarga tag input ini dibedakan berdasarkan atribut type:

1. <input type="text" /> atau bisa juga <input /> adalah textbox inputan biasa yang menerima input berupa text, contohnya digunakan untuk inputan nama, username, dan inputan yang berupa text pendek. Input type text ini juga bisa memiliki atribut value yang bisa diisi nilai tampilan awal dari text
2. <input type="password" /> dalam tampilannya sama dengan type text, namun teks yang diinput tidak akan terlihat, akan berupa bintang atau bulatan. Biasanya hanya digunakan untuk inputan yang sensitif seperti password.
3. <input type="checkbox" /> adalah inputan berupa checkbox yang dapat diceklist atau di centang oleh user. User dapat memilih atau tidak memilih checkbox ini. Type checkbox memiliki atribut checked yang jika ditulis atau diisi dengan nilai checked, akan membuat checkbox langsung terpilih pada saat pertama kali halaman ditampilkan. Contoh inputan checkbox berupa hobi, yang oleh user dapat dipilih beberapa hobi.
4. <input type="radio" /> mirip dengan checkbox, namun user hanya bisa memilih satu diantara pilihan group radio. Type radio ini berada dalam suatu grup dan user hanya bisa memilih salah satunya. Contoh inputan type radio adalah jenis kelamin.
5. <input type="submit" /> akan menampilkan tombol untuk memproses form. Biasanya diletakkan pada baris terakhir dari form. Atribut value jika diisi akan membuat text tombol submit berubah sesuai inputan nilai value.

Mengenal tag <textarea>

Tag textarea pada dasarnya sama dengan input type text, namun lebih besar dan dapat berisi banyak baris. Panjang dan banyak baris untuk text area di atur melalui atribut rows dan cols, atau melalui CSS.



```

1 <textarea rows="5" cols="20">
2 Text yang diisi dapat mencapai banyak baris
3 </textarea>

```

Text yang diisi
dapat mencapai banyak
baris

Gambar 16. Contoh penulisan text area dan hasilnya

Mengenal tag <select>

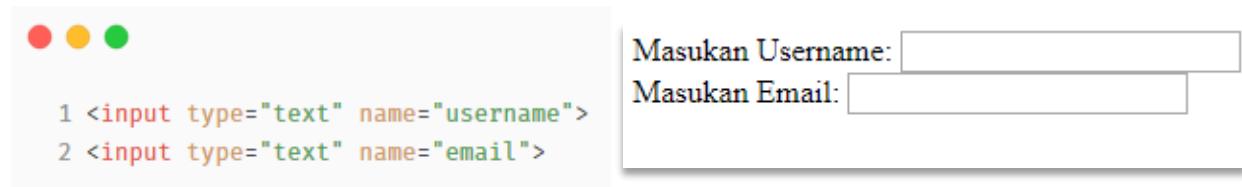
Tag select digunakan untuk inputan yang telah tersedia nilainya, dan user hanya dapat memilih dari nilai yang ada. Tag select digunakan bersama-sama dengan tag option untuk membuat box pilihan.



Gambar 17. Contoh penulisan select dan hasilnya

Mengenal Atribut: Name

Setiap tag inputan di dalam form harus ditambahkan atribut name agar dapat diproses oleh web server nantinya. Di dalam halaman proses (yang biasanya berupa bahasa PHP atau ASP), nilai dari atribut **name** inilah yang akan menjadi variabel form. Contoh pemakaiannya adalah sebagai berikut:



Gambar 18. Contoh penerapan atribute name di tag input

Membuat Layout

Halaman web pada umumnya terbagi menjadi beberapa kolom, seperti majalah atau berita dan hal ini disebut sebagai layout.



- <header> - Defines a header for a document or a section
- <nav> - Defines a container for navigation links
- <section> - Defines a section in a document
- <article> - Defines an independent self-contained article
- <aside> - Defines content aside from the content (like a sidebar)
- <footer> - Defines a footer for a document or a section
- <details> - Defines additional details
- <summary> - Defines a heading for the <details> element

Gambar 19. Pembagian layout pada halaman website

Pengertian CSS

Dalam bahasa bakunya, seperti di kutip dari wikipedia, CSS adalah “kumpulan kode yang digunakan untuk mendefenisikan desain dari bahasa markup”, dimana bahasa markup ini salah satunya adalah HTML.

Untuk pengertian bebasnya, CSS adalah kumpulan kode program yang digunakan untuk mendesain atau mempercantik tampilan halaman HTML. Dengan CSS dapat mengubah desain dari text, warna, gambar dan latar belakang dari (hampir) semua kode tag HTML.

CSS biasanya selalu dikaitkan dengan HTML, karena keduanya memang saling melengkapi. HTML ditujukan untuk membuat struktur, atau konten dari halaman web. Sedangkan CSS digunakan untuk tampilan dari halaman web tersebut. Istilahnya, “HTML for content, CSS for Presentation”.

The screenshot shows a code editor with two panes. The left pane contains the following HTML and CSS code:

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Test Background Color CSS</title>
5   <style type="text/css">
6     .warna {
7       color: red;
8     }
9   </style>
10 </head>
11 <body>
12   <p>
13     CSS merupakan bahasanya
14     <span class=warna>desainer web</span>.
15   </p>
16 </body>
17 </html>
  
```

The right pane shows the resulting rendered HTML with the text "CSS merupakan bahasanya" in red and the word "desainer web" in red.

Gambar 20. Contoh Penerapan CSS dan hasilnya

Tag sendiri merupakan tag yang tidak bermakna, namun bisa dikostumasi menggunakan CSS. Tag span ditambahkan dengan atribut class="warna". Atribut class berguna untuk memasukkan kode CSS pada tag <style> di bagian head.

Teknik Styling CSS: Inline, Internal, dan External

Ada beberapa contoh dari teknik styling pada CSS, yaitu

1. Inline Style

Style didefinisikan di dalam atribut style pada elemen, dan hanya satu elemen yang dikenai style tersebut, tidak berlaku untuk elemen lainnya.



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Contoh Inline CSS</title>
5 </head>
6 <body>
7   <h2 style="color:red;font-family:sans">Ini judul artikel</h2>
8   <p style="color:maroon">Ini adalah paragraf yang memuat isi artikel. Paragraf ini hanya
    untuk percobaan saja. Percobaan untuk mendemokan <i>internal css</i>.</p>
9 </body>
10 </html>
```

Gambar 21. Contoh penerapan inline style pada CSS

2. Internal Style

CSS Style Internal diload setiap kali website di-refresh, dan kekurangannya adalah waktu loading semakin lama. CSS style yang sama pun tidak dapat digunakan di halaman lain karena sudah aktif terlebih dulu di suatu halaman. Namun di balik kekurangannya, CSS Style Internal memiliki beberapa kelebihan. Salah satunya adalah kemudahan dalam sharing template untuk pratinjau (preview) karena CSS hanya ada di satu halaman.

Style didefinisikan di dalam tag <style>, dan semua elemen yang terpilih oleh selector (dan turunannya) akan dikenai style.

```
● ● ●
```

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Contoh Internal CSS</title>
5 </head>
6 <body>
7   <style type="text/css">
8     h2 {
9       font-family: sans;
10      color: #333;
11    }
12  </style>
13  <h2>Ini judul artikel</h2>
14  <p>Ini adalah paragraf yang memuat isi artikel. Paragraf ini hanya untuk percobaan saja. Percobaan untuk mendemokan
15    <i>internal css</i>. Seperti namanya, <i>inline CSS</i> adalah kode CSS yang ditulis langsung dalam file HTML.</p>
16 </body>
17 </html>
```

Gambar 22. Contoh penerapan internal style pada CSS

3. External Style

External merupakan CSS style yang paling mudah dan tidak menyulitkan. Semuanya dilakukan secara eksternal pada file .css. Styling dilakukan di file terpisah, lalu terapkan CSS ke halaman mana pun yang Anda inginkan. Sayangnya, CSS Style External juga memperlama waktu loading.

Sebagai contoh, kita akan membuat sebuah file bernama style-ku.css. Berikut ini cuplikan isi file style-ku.css

```
● ● ●
```

```
1 p {
2   font-family: serif; line-height: 1.75em;
3 }
4 i {
5   font-family: sans; color: orange;
6 }
7 h2 {
8   font-family: sans; color: #333;
9 }
```

Gambar 23. Isi file dari style-ku.css

Untuk menggunakan CSS tersebut dalam HTML, maka perlu mengimportnya. Ada beberapa cara memasukkan kode CSS dari berkas eksternal:

1. Menggunakan tag <link>

```
1 <link rel="stylesheet" type="text/css" href="style-ku.css">
```

2. Menggunakan Import

```
1 <style type="text/css">
2 @import "style-ku.css";
3 </style>
```

Penulisan pada HTML versi lengkapnya seperti ini:

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Contoh Eksternal CSS</title>
5   <link rel="stylesheet" type="text/css" href="style-ku.css">
6 </head>
7 <body>
8   <h2>Ini judul artikel</h2>
9   <p>Ini adalah paragraf yang memuat isi artikel. Paragraf ini hanya untuk percobaan saja.
10  Percobaan untuk mendemokan <i>internal css</i>. Seperti namanya, <i>inline CSS</i>
11  adalah kode CSS yang ditulis langsung dalam file HTML.</p>
12 </body>
13 </html>

```

Gambar 24. CSS external dan Penulisan di .html

Selector, Property dan Value pada CSS

Pengertian Selector CSS

Karena kode CSS digunakan untuk mengubah/memanipulasi tampilan dari tag HTML, CSS membutuhkan suatu cara untuk ‘mengaitkan’ atau menghubungkan kode CSS dengan tag HTML yang sesuai. Hal inilah yang dimaksud dengan Selector dalam CSS.

Sesuai dengan namanya, selector digunakan untuk mencari bagian web yang ingin dimanipulasi atau yang ingin di-style. Misalnya : “cari seluruh tag <p>”, atau “cari seluruh tag HTML yang memiliki atribut class=”warning”” atau “cari seluruh link yang ada di dalam tag <p>”.

Selector paling dasar dari CSS adalah tag dari HTML itu sendiri, misalnya: tag p, i, h1, li, dll. Selector didalam CSS dapat menjadi kompleks tergantung kebutuhannya.

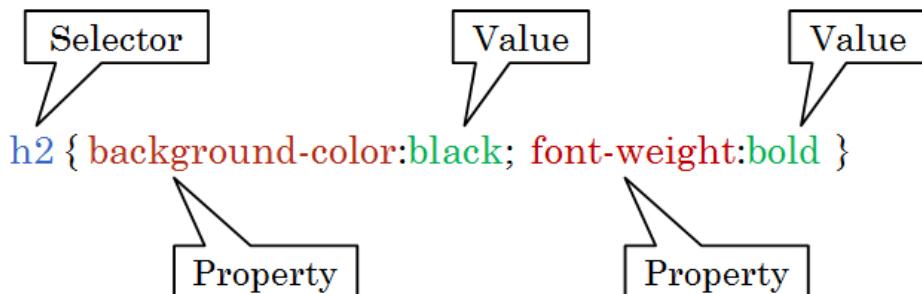
Pengertian Property CSS

Property CSS adalah jenis style, atau elemen apa yang akan diubah dari sebuah tag HTML. CSS memiliki puluhan property yang dapat digunakan agar menampilkan hasil akhir yang di inginkan. Hampir semua property dalam CSS dapat dipakai untuk seluruh selector.

Jika selector digunakan misalnya untuk “mencari seluruh tag <p>”, maka property adalah “efek apa yang ingin dimanipulasi dari tag p tersebut“, seperti ukuran text, warna text, jenis fontnya, dll.

Pengertian Value CSS

Value CSS adalah nilai dari property. Misalkan untuk property background-color yang digunakan untuk mengubah warna latar belakang dari sebuah selector, value atau nilainya dapat berupa red, blue, black, atau white



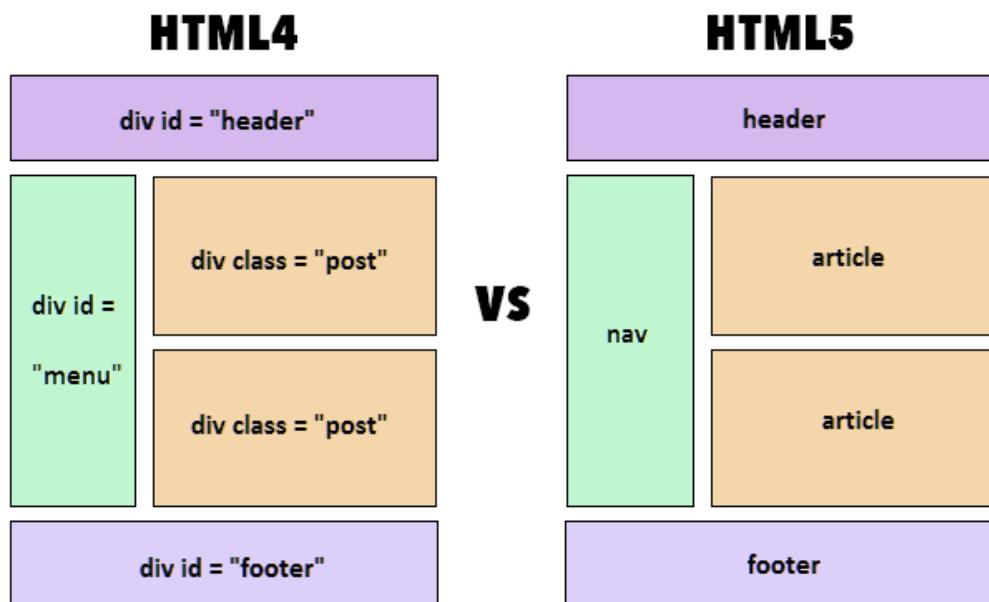
Gambar 25. Selector, property dan value pada CSS

IV HTML5

Perbedaan HTML & HTML5

Tidak ada bahasa yang tidak pernah di-upgrade atau memiliki rilis baru. Begitu juga dengan HTML, tidak terkecuali. HTML5 ini dirilis dengan tujuan utama meningkatkan pengalaman ber-internet untuk pengembang dan pengguna. Ada beberapa element baru yang terdapat di HTML5, seperti:

1. New semantic elements <header>, <footer>, <article>, and <section>.
2. New attributes of form elements number, date, time, calendar, and range.
3. New graphic elements: <svg> & <canvas>.
4. New multimedia elements: <audio> & <video>.

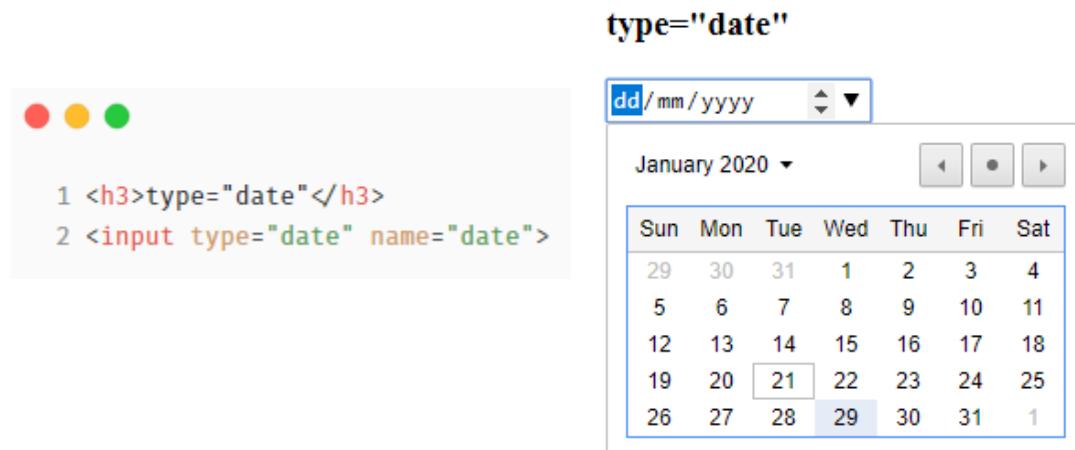


Gambar 26. HTML dan HTML5

Penambahan input type & Validasi Form

Sebelum HTML5, sangat susah memvalidasi Form. Misalkan sebuah input tidak boleh kosong, input harus angka dan input yang valid diantara 1 sampai 100. Untuk membuat validasi tersebut dapat menggunakan javascript atau PHP di server yang mana masing masing programmer dan framework mempunyai caranya sendiri sendiri. Dengan adanya HTML5 maka validasi sudah langsung ditangani oleh HTML5.

Form tidak terbatas pada type text, password, file dan Hidden, HTML5 menambahkan input tipe telefon, url, email,number, range, color, date dan image. Dengan berbagai range input yang luas, maka form di HTML5 menjadi cikal bakal aplikasi berbasis web yang lebih populer dengan nama HTML5 App atau WebApp.



Gambar 26. Contoh penerapan input date dan hasilnya

Native Audio dan Video

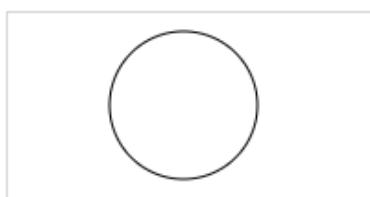
Sejak era 2000-an, audio dan video di internet tidak mempunyai standar. Kebanyakan audio menggunakan plugin itunes atau realplayer sedangkan video player menggunakan Flash atau silverLight. HTML5 menghadirkan Native audio dan Video Codec langsung didalam browser. Tidak ada lagi namanya update plugin atau browser crash karena plugin error.



Gambar 27. Contoh penerapan audio dan Hasilnya

Canvas

Canvas adalah fitur yang menarik. Bahasa mudahnya, dengan canvas dapat menggambar atau menampilkan animasi langsung di browser.



```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <canvas id="myCanvas" width="200" height="100"
5      style="border:1px solid #d3d3d3;">
6 Your browser does not support the HTML5 canvas tag.</canvas>
7 <script>
8 var c = document.getElementById("myCanvas");
9 var ctx = c.getContext("2d");
10 ctx.beginPath();
11 ctx.arc(95,50,40,0,2*Math.PI);
12 ctx.stroke();
13 </script>
14
15 </body>
16 </html>
```

Gambar 28. Contoh penerapan canvas untuk menggambar lingkaran

Ada banyak elemen baru yang diperkenalkan dalam HTML. Beberapa yang paling penting adalah: summary, time, aside, audio, command, data, datalist, details, embed, wbr, figcaption, figure, footer, header, article, hgroup, bdi, canvas, keygen, mark, meter, nav, output, progress, rp, rt, ruby, section, source, track, video.

V JAVASCRIPT

JavaScript adalah bahasa pemrograman web yang bersifat Client Side Programming Language. Client Side Programming Language adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh client. Aplikasi client yang dimaksud merujuk kepada web browser seperti Google Chrome dan Mozilla Firefox.

Bahasa pemrograman Client Side berbeda dengan bahasa pemrograman Server Side seperti PHP, dimana untuk server side seluruh kode program dijalankan di sisi server. Untuk menjalankan JavaScript, kita hanya membutuhkan aplikasi text editor dan web browser. JavaScript memiliki fitur: high-level programming language, client-side, loosely typed dan berorientasi objek.

Fungsi JavaScript Dalam Pemograman Web

JavaScript pada awal perkembangannya berfungsi untuk membuat interaksi antara user dengan situs web menjadi lebih cepat tanpa harus menunggu pemrosesan di web server. Sebelum javascript, setiap interaksi dari user harus diproses oleh web server.

Bayangkan ketika kita mengisi form registrasi untuk pendaftaran sebuah situs web, lalu men-klik tombol submit, menunggu sekitar 20 detik untuk website memproses isian form tersebut, dan mendapatkan halaman yang menyatakan bahwa terdapat kolom form yang masih belum diisi. Untuk keperluan seperti inilah JavaScript dikembangkan. Pemrosesan untuk mengecek apakah seluruh form telah terisi atau tidak, bisa dipindahkan dari web server ke dalam web browser.

Dalam perkembangan selanjutnya, JavaScript tidak hanya berguna untuk validasi form, namun untuk berbagai keperluan yang lebih modern. Berbagai animasi untuk mempercantik halaman web, fitur chatting, efek-efek modern, games, semuanya bisa dibuat menggunakan JavaScript.

Akan tetapi karena sifatnya yang dijalankan di sisi client yakni di dalam web browser yang digunakan oleh pengunjung situs, user sepenuhnya dapat mengontrol eksekusi JavaScript. Hampir semua web browser menyediakan fasilitas untuk mematikan JavaScript, atau bahkan mengubah kode JavaScript yang ada. Sehingga kita tidak bisa bergantung sepenuhnya kepada JavaScript.

Perkembangan JavaScript Saat Ini

Dalam perkembangannya, JavaScript mengalami permasalahan yang sama seperti kode pemograman web yang bersifat client side seperti CSS, yakni bergantung kepada implementasi web browser. Kode JavaScript yang kita buat, bisa saja tidak bekerja di Internet Explorer, karena web browser tersebut tidak mendukungnya. Sehingga programmer harus bekerja extra untuk membuat kode program agar bisa "mengakali" dukungan dari web browser.

Karena hal tersebut, JavaScript pada awalnya termasuk bahasa pemograman yang rumit, karena harus membuat beberapa kode program untuk berbagai web browser. Namun, beberapa tahun belakangan ini, JavaScript kembali bersinar berkat kemudahan yang ditawari oleh komunitas programmer yang membuat library JavaScript seperti jQuery. Library ini memudahkan kita membuat program JavaScript untuk semua web browser, dan membuat fitur-fitur canggih yang sebelumnya membutuhkan ribuan baris kode program menjadi sederhana. Kedepannya, JavaScript akan tetap menjadi kebutuhan programmer, apalagi untuk situs saat ini yang mengharuskan punya banyak fitur modern sebagai standar.

Menjalankan kode JavaScript

Cara penulisan JavaScript mirip dengan penulisan bahasa pemograman web lainnya seperti PHP dan CSS, yakni dengan menyisipkan kode JavaScript di dalam HTML.



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Belajar JavaScript di DuniaIlkom</title>
5
6 <script type="text/javascript">
7 function tambah_semangat()
8 {
9     var a=document.getElementById("div_semangat");
10    a.innerHTML+="<p>Sedang Belajar JavaScript, Semangat ... !!!</p>";
11 }
12 </script>
13
14 </head>
15 <body>
16 <h1>Belajar JavaScript</h1>
17 <p> Saya sedang belajar JavaScript di duniaIlkom.com </p>
18 Klik tombol ini untuk menambahkan kalimat baru:
19 <button id="tambah" onclick="tambah_semangat()">Semangaat .. !!</button>
20 <div id="div_semangat"></div>
21 </body>
22 </html>
```

Belajar JavaScript

Saya sedang belajar JavaScript di duniaIlkom.com

Klik tombol ini untuk menambahkan kalimat baru:

Sedang Belajar JavaScript, Semangat...!!!

Gambar 29. Contoh penulisan JavaScript dan Hasil nya

Memasukkan kode JavaScript ke dalam HTML

JavaScript termasuk jenis bahasa script, yang digunakan di dalam file HTML. Untuk menginput, atau memasukkan kode JavaScript ke dalam HTML, JavaScript menyediakan 4 alternatif, yaitu:

Menggunakan tag <script> (internal JavaScript)

Cara pertama untuk menginput kode JavaScript ke dalam halaman HTML adalah dengan menggunakan tag <script> secara internal. Internal disini berarti bahwa kode JavaScript ditulis pada halaman yang sama dengan HTML, atau di dalam satu file HTML. Kode JavaScript yang akan diinput diletakkan diantara tag pembuka <script> dan tag penutup </script> seperti berikut ini:

```
1 <script>
2 //kode javascript diletakkan disini
3 </script>
```

```
1 <script type="text/javascript">
2 //kode javascript diletakkan disini
3 </script>
```

Menggunakan tag <script scr=""> (external JavaScript)

Cara atau metode kedua untuk menginput kode JavaScript ke dalam halaman HTML adalah dengan memindahkan kode JavaScript ke dalam sebuah file terpisah, lalu ‘memanggilnya’ dari HTML. Cara ini sangat disarankan karena akan memberikan banyak keuntungan dan fleksibilitas dalam membuat program JavaScript.

Sebuah file JavaScript disimpan dalam ekstensi .js, seperti: kode.js, register.js, atau kodeku.js. Dari halaman HTML, kita memanggilnya menggunakan tag <script> dengan atribut src. Atribut src berisi alamat dari file javascript tersebut, seperti berikut ini:

```
1 <script src="kode_javascript.js"></script>
```

Menggunakan Event Handler (Inline JavaScript)

Konsep Event Handler akan kita pelajari secara khusus pada tutorial terpisah, namun secara sederhananya, event handler adalah pemanggilan kode javascript ketika ‘sesuatu’ terjadi dalam tag HTML.

Sesuatu disini maksudnya ketika sebuah element dalam HTML di klik, di klik kanan, di arahkan mouse, dan lain-lain. Event handler di dalam JavaScript ditulis dengan penambahan kata on. Sehingga jika sebuah tombol di-klik, maka disebut sebagai **onclick**, jika mouse berada diatas element disebut sebagai **onmouseover**, dan lain-lain.

```
1 <button onclick="alert('Hello World!! ')>Klik Saya|
```



Varibel JavaScript

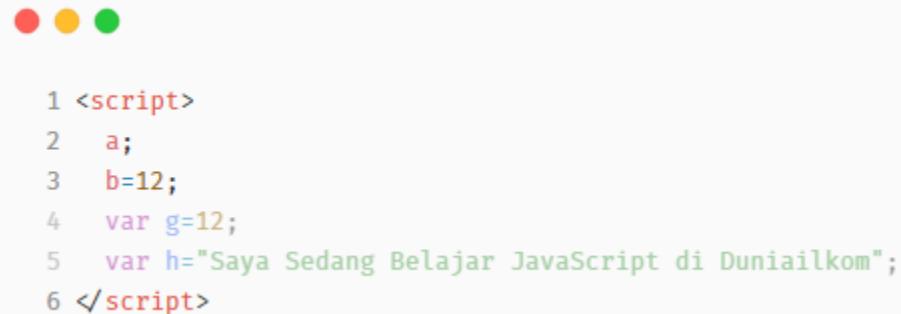
Dalam bahasa pemrograman, variabel adalah ‘penampung’ sebuah nilai. Tergantung dengan ‘nilai’ dari variabel tersebut, sebuah variabel di dalam JavaScript dapat bertipe Angka (Number), String, Boolean, atau yang lainnya. Aturan penamaan variabel pernah kita bahas pada tutorial Aturan Dasar Penulisan Kode Program JavaScript yakni sama dengan aturan pembuatan identifier:

1. Karakter pertama harus diawali dengan huruf, underscore (_) atau tanda dollar (\$)

2. Karakter kedua dan seterusnya bisa ditambahkan dengan huruf, angka, underscore (_) atau tanda dollar (\$).

Membuat Variabel JavaScript

Walaupun kita tidak perlu menyebutkan jenis tipe data dari suatu variabel, namun kita tetap harus mendeklarasikan variabel di dalam JavaScript. Cara membuat variabel di dalam JavaScript di bedakan menjadi 2, yakni dengan menggunakan keyword var, dan tanpa var.



```
1 <script>
2   a;
3   b=12;
4   var g=12;
5   var h="Saya Sedang Belajar JavaScript di DuniaIlkom";
6 </script>
```

Gambar 30. Contoh penulisan Variabel JavaScript

Tipe data dalam JavaScript

Tipe data dalam JavaScript dibedakan menjadi 2 kelompok, yakni tipe data dasar (primitif) dan tipe data objek. Tipe data dasar terdiri dari tipe data angka, tipe data text (string), dan tipe data boolean. Tipe data null dan undefined juga merupakan tipe data dasar, namun memiliki jenis tersendiri. Selain ke-5 tipe data dasar tersebut, tipe data lain yang ada di dalam JavaScript adalah tipe data objek. Contoh tipe data objek adalah tipe data tanggal (date), array, dan function.

Menampilkan Output pada Javascript

Output adalah sebuah tampilan program yang biasanya digunakan untuk memperlihatkan hasil akhir. Output biasanya ditampilkan dalam bentuk teks dengan fungsi print(). Ada 4 cara menampilkan output pada Javascript:

1. Menggunakan Fungsi console.log();
2. Menggunakan Fungsi alert();
3. Menggunakan Fungsi document.write();
4. Menggunakan innerHTML.

Menggunakan Fungsi console.log()

Fungsi console.log() adalah fungsi untuk menampilkan teks ke console Javascript. Contoh penggunaan:



```
1 console.log("Hello World!");
```

Console

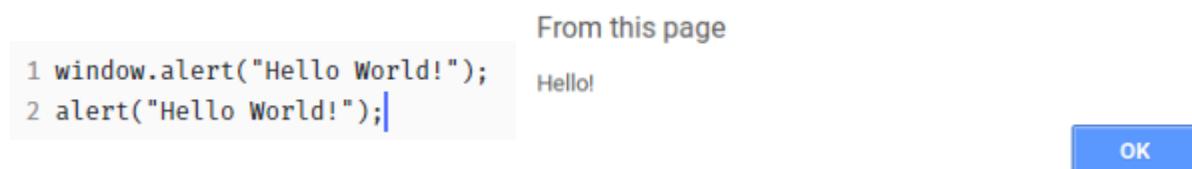
```
> console.log("Hello World!");
Hello World!
< undefined
```

Gambar 31. Contoh menampilkan menggunakan console.log dan hasilnya

Fungsi console.log() biasanya digunakan untuk debugging. Karena setiap pesan error di Javascript selalu ditampilkan di dalam Console. Selain console.log(), terdapat juga beberapa fungsi untuk debugging seperti console.debug(), console.info(), console.error(), console.dir(), dsb.

Menggunakan Fungsi alert()

Fungsi alert() adalah fungsi untuk menampilkan jendela dialog. Fungsi sebenarnya berada pada objek window. Fungsi alert(), hanya bisa digunakan di dalam browser saja. Secara lengkap bisa ditulis seperti ini:



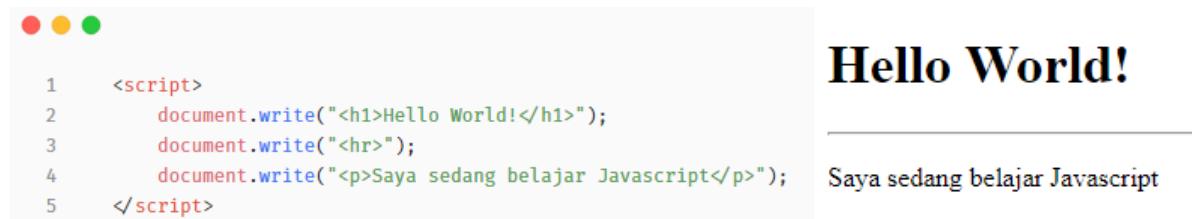
The screenshot shows a browser window with an alert dialog. The dialog has a title bar 'From this page' and a message area containing 'Hello'. In the bottom right corner of the message area is a blue 'OK' button. The background of the browser window is visible, showing some code snippets.

```
1 window.alert("Hello World!");
2 alert("Hello World!");
```

Gambar 32. Contoh menampilkan menggunakan alert dan hasilnya

Menggunakan Fungsi document.write()

Objek document adalah objek yang mewakili dokumen HTML di dalam Javascript. Dalam objek document, terdapat fungsi write() untuk menulis sesuatu ke dokumen HTML.



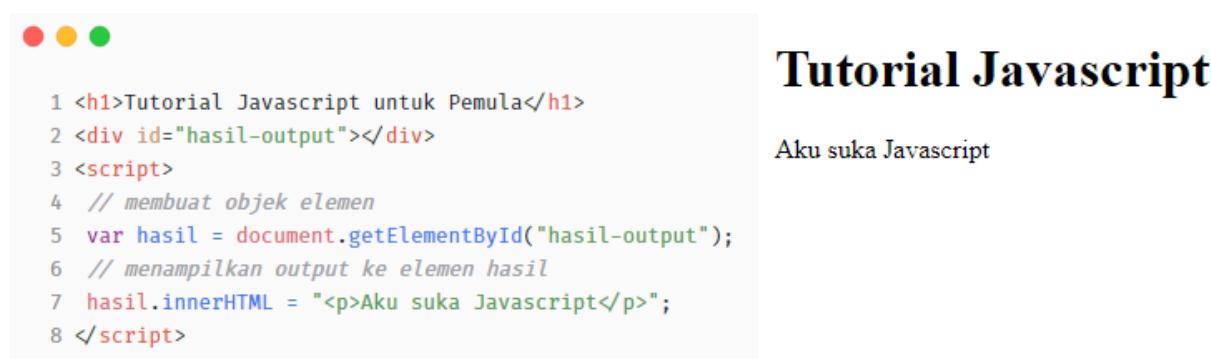
The screenshot shows a browser window with the text 'Hello World!' displayed prominently in large bold letters. Below it is a horizontal line and the text 'Saya sedang belajar Javascript'. The background shows some code snippets.

```
1 <script>
2   document.write("<h1>Hello World!</h1>");
3   document.write("<hr>");
4   document.write("<p>Saya sedang belajar Javascript</p>");
5 </script>
```

Gambar 33. Contoh menampilkan menggunakan document.write() dan hasilnya

Menggunakan innerHTML

innerHTML adalah sebuah atribut di dalam (objek) elemen HTML yang berisi string HTML. Dengan innerHTML, kita dapat menampilkan output ke elemen yang lebih spesifik. Contoh:



The screenshot shows a browser window with the text 'Tutorial Javascript' in a large font. Below it is a smaller text 'Aku suka Javascript'. The background shows some code snippets.

```
1 <h1>Tutorial Javascript untuk Pemula</h1>
2 <div id="hasil-output"></div>
3 <script>
4 // membuat objek elemen
5 var hasil = document.getElementById("hasil-output");
6 // menampilkan output ke elemen hasil
7 hasil.innerHTML = "<p>Aku suka Javascript</p>";
8 </script>
```

Gambar 34. Contoh menampilkan menggunakan innerHTML dan hasilnya

Operator di JavaScript

Operator merupakan hal dasar yang harus dipahami dalam pemrograman. Karena kita akan banyak menggunakan untuk melakukan berbagai macam operasi di dalam program. Operator adalah simbol yang digunakan untuk melakukan operasi pada suatu nilai dan variabel. Operator dalam pemrograman terbagi dalam 6 jenis:

1. **Operator aritmatika;** Operator aritmatika merupakan operator untuk melakukan operasi aritmatika seperti penjumlahan, pengurangan, pembagian, perkalian, dsb.
2. **Operator Penugasan (Assignment);** Operator penugasan adalah operator yang digunakan untuk memberikan tugas kepada variabel. Biasanya digunakan untuk mengisi variabel.
3. **Operator relasi atau perbandingan;** Operator relasi atau perbandingan adalah operator yang digunakan untuk membandingkan dua nilai. Operator perbandingan akan menghasilkan sebuah nilai boolean true dan false.
4. **Operator Logika;** Operator logika digunakan untuk melakukan operasi terhadap dua nilai boolean.
5. **Operator Bitwise;** Operator bitwise merupakan operator yang digunakan untuk operasi berdasarkan bit (biner).
6. **Operator Ternary;** Operator-operator sebelumnya hanya dua bagian saja, yaitu: bagian kiri dan kanan. Ini disebut operator binary. Sementara operator trinary ada bagian kiri, tengah, dan kanan.



Gambar 35. Contoh penggunaan operator ternary

Percabangan pada Javascript

Percabangan akan mampu membuat program berpikir dan menentukan tindakan sesuai dengan logika/kondisi yang kita berikan.

Percabangan if

Percabangan if merupakan percabangan yang hanya memiliki satu blok pilihan saat kondisi bernilai benar. Contoh: "Jika total belanja lebih besar dari Rp 100.000, Maka tampilkan pesan Selamat, Anda dapat hadiah"

```
1 <script>
2 var totalBelanja = prompt("Total belanja?", 0);
3 if(totalBelanja > 100000){
4   document.write("<h2>Selamat Anda dapat hadiah</h2>");
5 }
6 document.write("<p>Terimakasih sudah berbelanja di toko kami</p>");
7 </script>
```

Gambar 36. Contoh penerapan percabangan IF

Percabangan if/else

Percabangan if/else merupakan percabangan yang memiliki dua blok pilihan. Pilihan pertama untuk kondisi benar, dan pilihan kedua untuk kondisi salah (else). Contoh: Apabila password benar, pesan yang akan ditampilkan: "Selamat datang bos!". Tapi kalau salah, maka pesan yang akan ditampilkan: "Password salah, coba lagi!"

```
1 <script>
2 var password = prompt("Password:");
3 if(password == "kopi"){
4   document.write("<h2>Selamat datang bos!</h2>");
5 } else {
6   document.write("<p>Password salah, coba lagi!</p>");
7 }
8 document.write("<p>Terima kasih!</p>");
9 </script>
```

Gambar 37. Contoh penerapan percabangan IF-ELSE

Perulangan pada Javasc

Perulangan akan membantu kita mengeksekusi kode yang berulang-ulang, berapapun yang kita mau. Ada lima macam bentuk perulangan di Javascript. Secara umum, perulangan ini dibagi dua. Yaitu: **counted loop** dan **uncounted loop**. Perbedaannya: **Counted Loop** merupakan perulangan yang jelas dan sudah tentu banyak perulangannya. Sedangkan **Uncounted Loop**, merupakan perulangan yang tidak jelas berapa kali ia harus mengulang.

- Perulangan yang termasuk dalam Counted Loop: Perulangan For, Perulangan Foreach, Perulangan Repeat
- Perulangan yang termasuk dalam Uncounted Loop: Perulangan While, Perulangan Do/While

Perulangan For di Javascript

Perulangan for merupakan perulangan yang termasuk dalam counted loop, karena sudah jelas berapa kali ia akan mengulang.



```
1 for(let i = 0; i < 10; i++){
2     document.write("<p>Perulangan ke-" + i + "</p>")
3 }
```

Gambar 38. Contoh penerapan perulangan FOR

Perulangan While di Javascript

Perulangan while merupakan perulangan yang termasuk dalam perulangan uncounted loop. Perulangan while juga dapat menjadi perulangan yang counted loop dengan memberikan counter di dalamnya.



```
1 var ulangi = confirm("Apakah anda mau mengulang?");
2 var counter = 0;
3 while(ulangi){
4     counter++;
5     ulangi = confirm("Apakah anda mau mengulang?");
6 }
```

Gambar 39. Contoh penerapan perulangan WHILE

Perulangan akan terjadi selama variabel ulangi bernilai true. Lalu kita menggunakan fungsi confirm() untuk menampilkan dialog konfirmasi. Selama kita memilih Ok pada dialog konfirmasi, maka variabel ulangi akan terus bernilai true. Tapi kalau kita pilih Cancel, maka variabel ulangi akan bernilai false. Saat variabel ulangi bernilai false, maka perulangan akan dihentikan.

Fungsi di Javascript

Dalam pemrograman, fungsi sering digunakan untuk membungkus program menjadi bagian-bagian kecil. Logika program yang ada di dalam fungsi dapat kita gunakan kembali dengan memanggilnya. Sehingga tidak perlu menulis ulang.

Fungsi adalah sub-program yang bisa digunakan kembali baik di dalam program itu sendiri, maupun di program yang lain. Fungsi di dalam Javascript adalah sebuah objek. Karena memiliki properti dan juga method.

Membuat Fungsi

Cara ini paling sering digunakan, terutama buat yang baru belajar Javascript.



```
1 function namaFungsi(){
2     console.log("Hello World!");
3 }
```

Gambar 40. Contoh penerapan fungsi di javascript

Memanggil/Eksekusi Fungsi

Setelah mengetahui cara membuat fungsi, lalu bagaimana cara memanggilnya?



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <script>
5         // membuat fungsi
6         var sayHello = () => alert("Hello World!");
7     </script>
8 </head>
9 <body>
10    <!-- Memanggil fungsi saat link diklik -->
11    <a href="#" onclick="sayHello()">Klik Aku!</a>
12 </body>
13 </html>
```

Gambar 41. Contoh pemanggilan fungsi javascript di HTML

Fungsi dengan Parameter

Parameter adalah variabel yang menyimpan nilai untuk diproses di dalam fungsi.



```
1 function kali(a, b){
2     hasilKali = a * b;
3     console.log("Hasil kali a*b = " + hasilKali);
4 }
5 kali(3, 2); // → Hasil kali a*b = 6
```

Gambar 42. Contoh pemanggilan fungsi dengan parameter di javascript

Fungsi yang Mengembalikan Nilai

Agar hasil pengolahan nilai di dalam fungsi dapat digunakan untuk proses berikutnya, maka fungsi harus mengembalikan nilai. Pengembalian nilai pada fungsi menggunakan kata kunci return kemudian diikuti dengan nilai atau variabel yang akan dikembalikan. Contoh:

```
 1 function bagi(a,b){  
 2     hasilBagi = a / b;  
 3     return hasilBagi;  
 4 }  
 5  
 6 // memanggil fungsi  
 7 var nilai1 = 20;  
 8 var nilai2 = 5;  
 9 var hasilPembagian = bagi(nilai1, nilai2);  
10  
11 console.log(hasilPembagian); //→ 4
```

Gambar 42. Contoh pemanggilan fungsi dengan return di javascript

Menerapkan Hasil Form HTML dengan JavaScript

Mengkobinasikan form html dengan javascript akan membuat website menjadi lebih interaktif, Berikut adalah contoh dari penerapan html dan javascript:

```
● ○ ●

1 <script>
2 function tampilkan(){
3
4     var nama_kota=document.getElementById("form1").select1.value;
5     var p_kontainer=document.getElementById("container");
6
7     if (nama_kota=="jakarta")
8     {
9         p_kontainer.innerHTML="Jakarta Ibu kota Republik Indonesia";
10    }
11   else if (nama_kota=="bandung")
12   {
13       p_kontainer.innerHTML="Bandung kota kembang";
14   }
15   else if (nama_kota=="bogor")
16   {
17       p_kontainer.innerHTML="Bogor kota hujan";
18   }
19 }
20 </script>
```

Gambar 42. Contoh kode program javascript

```
● ○ ●

1 <form id="form1" name="form1" onsubmit="return false">
2     <label for="select1">Pilih Kota: </label>
3     <select id="select1" name="select1">
4         <option value="jakarta">Jakarta</option>
5         <option value="bandung">Bandung</option>
6         <option value="bogor">Bogor</option>
7     </select>
8     <input type="submit" value="Tampilkan Keterangan" onclick="tampilkan()">
9 </form>
10
11 <p id="container"></p>
```

Gambar 43. Contoh kode program html

Penjelasan:

1. Form dengan atribut id="form1" dan name="form1". Form ini hanya memiliki 1 objek form, yakni select dan 1 buah tombol submit untuk mengirimkan hasil form.

2. Objek form <select> di beri atribut id="select1". dan name="select1". Setiap nilai dari select (tag <option>) memiliki value masing-masing, yakni 3 buah nama kota: Jakarta, Bandung, dan Bogor.
3. Atribut HTML yang berfungsi untuk ‘mengaitkan’ JavaScript, yakni onclick="tampilkan()". Atribut ini berfungsi sebagai event JavaScript yang akan dipanggil ketika tombol submit di klik.
4. Sebuah fungsi tampilkan() yang akan dieksekusi ketika tombol submit di tekan.
5. Perintah document.getElementById untuk mendapatkan suatu objek HTML dengan menggunakan atribut id. Sehingga document.getElementById("form1") akan menghasilkan objek HTML dengan id="form1"
6. Untuk mendapatkan nilai dari tag <select> dari form1, menggunakan perintah: document.getElementById("form1").select1.value
7. Perintah select1.value akan berisi nilai dari tag <select> yang saat ini sedang dipilih, dan disimpan dengan nama variabel nama_kota.
8. Variabel kedua adalah p_kontainer. Variabel ini berisi ‘objek’ dari kontainer dimana keterangan tentang kota akan kita tampilkan.
9. Perhatikan bahwa id dari tag <p> dalam kode HTML kita adalah: container.
10. Jika variabel nama_kota adalah Jakarta, maka tampilkan kata “Jakarta Ibu kota Republik Indonesia” di dalam “container”.

Menampilkan Hasil Form HTML dengan JavaScript

Pilih Kota:

Bandung kota kembang

Gambar 44. Hasil Eksekusi Kode Program

VI JQUERY

Pengertian jQuery

jQuery adalah sebuah library JavaScript. Dalam dunia pemrograman, library adalah kumpulan dari berbagai fungsi ‘siap pakai’ untuk memudahkan pembuatan sebuah aplikasi. Dengan demikian, jQuery adalah kumpulan fungsi-fungsi JavaScript yang memudahkan penulisan kode JavaScript.



Gambar 45. Logo jQuery (sumber: jquery.com)

jQuery dikembangkan pertama kali oleh John Resig di tahun 2006. Sejak saat itu, jQuery berkembang menjadi proyek opensource dan menjadi library JavaScript paling populer di dunia. Situs resmi jQuery beralamat di jquery.com.

Mengapa harus menggunakan jQuery?

JavaScript sangat powerfull untuk memanipulasi element HTML, tapi penggunaannya juga tidak mudah dan ‘sedikit ribet’. Sebagai contoh, untuk mencari sebuah element HTML, dengan JavaScript kita menggunakan fungsi getElementById():

```
1 var x = document.getElementById("tombol");
```

Dengan jQuery, penulisannya jauh lebih singkat:

```
1 var x = $("#tombol")
```

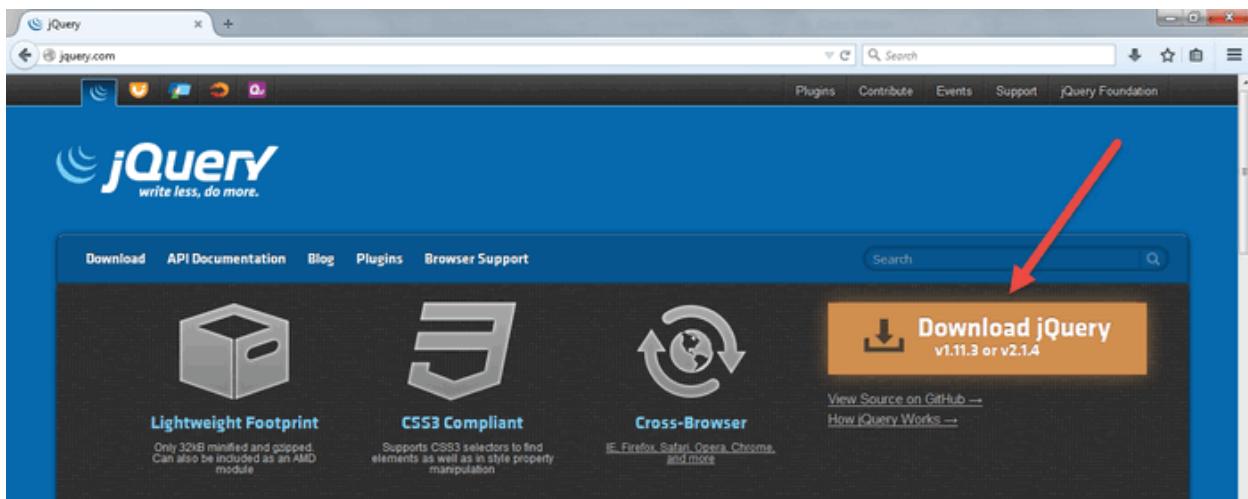
jQuery memiliki fungsi bawaan yang mendeteksi web browser dan menyediakan fungsi pengganti. Dengan demikian, tidak perlu memikirkan perbedaan implementasi JavaScript dari setiap web browser. Fungsi JavaScript lanjutan seperti *AJAX (Asynchronous JavaScript and XML)*, juga disederhanakan oleh jQuery. Untuk menggunakan AJAX, kita hanya butuh beberapa baris perintah.

Fitur lain dari jQuery (dan mungkin menjadi alasan paling penting), jQuery menyediakan berbagai efek animasi menarik. Beberapa diantaranya sangat mudah digunakan dan hanya membutuhkan 1 atau 2 baris kode program. Khusus untuk membuat efek animasi ini, jQuery juga memiliki jQuery UI yang fokus ke animasi dan interaksi user.

Menginput File jQuery ke HTML

Pada dasarnya, jQuery hanyalah sebuah file javascript external. Agar bisa menggunakan jQuery, maka terlebih dahulu mendownload file jQuery, kemudian menghubungkannya menggunakan tag <script>. Terdapat 2 cara menginput file jQuery: mendownload dan mengakses jQuery secara lokal, atau menggunakan CDN (Content Delivery Network).

Untuk menggunakan jQuery secara lokal (offline), terlebih dahulu mendownload file jQuery, kemudian mengaksesnya melalui tag <script>. File jQuery tersedia di situs resmi jQuery di jquery.com. Pada bagian kanan tengah, terdapat tombol “Download jQuery”.



Gambar 46. Download file jQuery

Menghubungkan File jQuery dengan HTML

Untuk menghubungkan file jQuery dengan HTML, cukup menambahkan tag <script> dengan alamat file **jquery-2.1.4.js**, seperti berikut ini:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Belajar jQuery</title>
6   <script src="jquery-2.1.4.js"></script>
7 </head>
8 <body>
9 </body>
10 </html>
```

A screenshot of a code editor window showing a basic HTML file. The code consists of a single document type declaration, an opening html tag, a head section containing a meta tag for UTF-8 encoding and a title 'Belajar jQuery', a script tag linking to 'jquery-2.1.4.js', a closing head tag, a body section, and a closing html tag. The code is numbered from 1 to 10 on the left side.

Gambar 47. Contoh menghubungkan file jQuery di HTML

Kode HTML diatas harus ditempatkan di dalam 1 folder dengan file **jquery-2.1.4.js**.

Menjalankan Kode jQuery (event ready)

jQuery sebenarnya adalah JavaScript. Dengan demikian, kode yang ditulis pun adalah kode JavaScript. jQuery hanya 'memudahkan' penulisan JavaScript dengan berbagai fungsi-fungsi bawaan. Untuk menginput kode JavaScript dengan 'rasa' jQuery, cara paling umum adalah menempatkannya setelah tag <script> yang digunakan untuk memanggil file jQuery, seperti contoh berikut:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Belajar jQuery</title>
6   <script src="jquery-2.1.4.js"></script>
7   <script>
8     // kode JavaScript dan jQuery disini
9   </script>
10 </head>
11 <body>
12 </body>
13 </html>
```

Gambar 48. Contoh penempatan kode jQuery

Dengan ‘template’ diatas, sebenarnya sudah bisa langsung menjalankan fungsi-fungsi jQuery. Namun terdapat 1 fungsi yang selalu disertakan dalam setiap kode jQuery, yakni event **ready()**.

Pengertian Event ready() atau The Document Ready Event

Secara sederhana, JavaScript digunakan untuk memanipulasi objek HTML seperti paragraf, tombol, form, gambar, dll. Jika menempatkan kode JavaScript di awal (pada bagian **<head>**), besar kemungkinan objek HTML ini belum tersedia. Ini terjadi karena bagian **<body>** baru diproses setelah bagian **<head>**. Salah satu solusi untuk hal ini adalah dengan meletakkan kode JavaScript di bagian penutup halaman, yakni sebelum tag penutup **</body>**. Namun jQuery memiliki solusi yang lebih elegant, yakni event **ready()**.

Event **ready()** jQuery men-simulasi efek dari event **Onload JavaScript**, tapi lebih efisien. Dengan menggunakan fungsi **ready()** jQuery, kode program JavaScript baru akan dijalankan setelah web browser selesai memproses tag HTML, tapi sebelum file external seperti gambar. Dengan demikian, kode JavaScript diproses dengan lebih awal daripada event **Onload** bawaan JavaScript. Penulisan fungsi **ready()** jQuery adalah sebagai berikut:

```
1 <script src="jquery-2.1.4.js"></script>
2   <script>
3     $(document).ready(function() {
4       // Kode jquery dan JavaScript disini
5     });
6   </script>
```

Gambar 49. Contoh penulisan event **ready()**.

Kode JavaScript yang berada didalam fungsi **\$(document).ready(function())** akan dijalankan setelah seluruh tag HTML diproses oleh web browser. Dengan demikian, bisa menghindari error yang terjadi kerena objek yang ingin dimanipulasi belum tersedia.

Sebagai contoh, terdapat sebuah kode program sederhana, dimana ketika tombol HTML di klik, akan tampil sebuah kalimat. Berikut kode programnya:

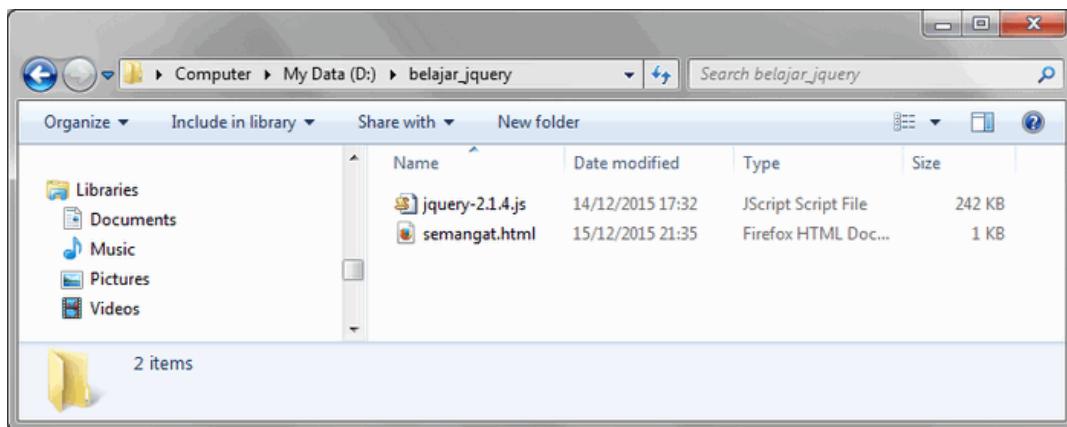
```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Belajar jQuery</title>
6 <script src="jquery-2.1.4.js"></script>
7 <script>
8   $(document).ready(function() {
9     $("#tombol").click(function() {
10       $(this).after('<p>Sedang Belajar jQuery</p>');
11     });
12   });
13 </script>
14 </head>
15 <body>
16 <h1>Belajar jQuery</h1>
17 <button id="tombol">Click Me!</button>
18 </body>
19 </html>

```

Gambar 50. Contoh kode sederhana dengan jQuery dan Hasilnya.

Agar bisa berjalan, simpanlah kode program diatas di folder yang sama dengan file **jquery-2.1.4.js**. Sebagai contoh, file tadi akan disimpan ke dalam file **semangat.html** di folder **belajar_jquery**, seperti berikut ini:



Gambar 51. Penempatan File HTML dan jQuery di dalam satu folder.

Aturan Dasar Penulisan Kode jQuery

Sepanjang penggunaan jQuery, hampir selalu didahului tanda \$. Tanda dollar '\$' (dollar sign) adalah sebuah shortcut untuk mengakses jQuery Object. Di dalam jQuery Object inilah seluruh fungsi-fungsi jQuery berada. Sebagai contoh, untuk mencari element HTML yang memiliki atribut **id="tombol"**. Seperti contoh berikut:

```
1 $("#tombol")
```

Jika karena sesuatu hal tidak bisa menggunakan tanda \$ (biasanya karena bentrok dengan library lain), bisa menggunakan **jQuery()**, seperti berikut:

```
1 jQuery("#tombol")
```

Pengertian jQuery Action / jQuery Event

Untuk bisa melakukan ‘sesuatu’ terhadap element HTML, kita tinggal menyambung penulisan jQuery Object dengan jQuery Action, atau dikenal juga dengan jQuery Event. Format dasarnya **\$(selector).action()**. Selector digunakan untuk mencari element mana dari HTML yang akan dimanipulasi. Bisa berupa sebuah paragraf **<p>**, sebuah gambar ****, atau sesuatu yang lebih kompleks, seperti element **<tr>** pertama dari tabel kedua.

Action adalah sesuatu yang bisa kita lakukan dengan element HTML tersebut, apakah menyembunyikannya, menampilkannya, mengubah warna, menambahkan element baru, dll. Action mirip seperti Event dalam JavaScript, tetapi dengan penambahan berbagai fitur lain. Di dalam jQuery, sebuah action bisa disambung dengan action lain, atau dikenal dengan istilah **chaining**.

Berikut contoh penulisan jQuery Object, selector dan action:

```
● ● ●  
18 </head>  
19 <body>  
20   <h1>Belajar jQuery</h1>  
21   <button id="tombol_hide">Sembunyikan</button>  
22   <button id="tombol_show">Tampilkan</button>  
23 </body>  
24 </html>  
  
1  <!DOCTYPE html>  
2  <html>  
3  <head>  
4  <meta charset="UTF-8">  
5  <title>Belajar jQuery</title>  
6  <script src="jquery-2.1.4.js"></script>  
7  <script>  
8    $( document ).ready(function() {  
9      $( "#tombol_hide" ).click(function() {  
10        $("h1").hide();  
11      });  
12  
13      $( "#tombol_show" ).click(function() {  
14        $("h1").show();  
15      });  
16    });  
17  </script>
```

Gambar 52. Contoh penulisan jQuery lengkap dengan object, selector dan action

Apabila button Tampilkan di klik maka tulisan pada tag **<h1>** akan tampil, dan bila button Sembunyikan di klik, maka tag **<h1>** akan di sembunyikan.

Mencari Elemen HTML dengan jQuery Selector

Sebagaimana fungsi utama JavaScript, kita memerlukan JavaScript untuk memanipulasi element HTML. Langkah pertama adalah mencari element mana yang akan diubah, apakah itu tag **<p>**, tag **<p>** dengan atribut **id="penting"**, tag **<p>** dengan atribut **class="warning"**, atau sebuah tombol **<button>**.

Di dalam JavaScript, untuk mencari element HTML ini kita biasa menggunakan method **getElementById()**. Method atau fungsi ini berfungsi untuk mencari element HTML berdasarkan atribut id-nya. Sebagai contoh, jika terdapat sebuah kode HTML berikut:

```
1 <button id="tombol">Click Me!</button>
```

Maka untuk mengakses element ini bisa menggunakan:

```
1 var x = document.getElementById("tombol");
```

Dengan menggunakan jQuery Selector, kode programnya jauh lebih singkat dan lebih powerfull.

Pengertian jQuery Selector

jQuery memudahkan proses pencarian element HTML dengan menyediakan jQuery Selector. jQuery Selector adalah sebuah cara yang disediakan jQuery untuk mencari element HTML. Tidak hanya dengan atribut id saja, kita juga bisa menggunakan class, atribut lain, dan kombinasi diantaranya.

Dengan jQuery, untuk mencari sebuah element HTML yang memiliki id="tombol", penulisannya jauh lebih singkat:

```
1 var x = $("#tombol");
```

Karena sama seperti selector CSS, jQuery selector mendukung hampir semua penulisan selector dari CSS, berikut contohnya:

- Mencari seluruh element dengan tag tertentu. Misalkan kita ingin mencari seluruh tag **<p>**, penulisan jQuery Selectornya adalah: **\$(“p”)**.
- Mencari element dengan id tertentu. Misalkan ingin mencari sebuah tag HTML yang memiliki **id="belajar"**, maka kita bisa menulis: **\$(“# belajar”)**.
- Mencari element dengan class tertentu. Misalkan kita ingin mencari seluruh tag HTML yang menggunakan atribut **class="warna"**, maka bisa ditulis dengan: **\$(“.warna”)**.

Sama seperti CSS, kita juga bisa mengkombinasikan selector untuk mencari element yang lebih spesifik, misalnya untuk mencari tag **<p>** yang memiliki **class="warna"** dan berada di dalam tag **<div>**, selectornya adalah: **\$(“div p.warna”)**.

jQuery Event dan Cara Penggunaannya

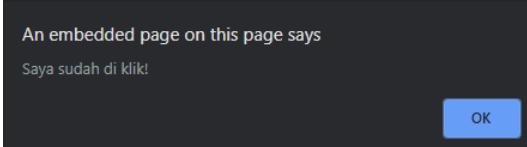
Di dalam JavaScript, *Event* adalah suatu hal yang bisa dilakukan user ke sebuah element HTML, misalnya di click, double click, mouseover (cursor mouse berada diatas element), mouseout (cursor mouse sudah keluar dari atas element), dll. jQuery menyediakan jQuery Event dengan penulisan yang lebih praktis, berikut contohnya:



```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Belajar jQuery</title>
6 <script src="jquery-2.1.4.js"></script>
7 <script>
8     $(document).ready(function() {
9         $("#paragraf").click(function() {
10            alert('Saya sudah di klik!');
11        });
12    });
13 </script>
14 </head>
15 <body>
16 <p id="paragraf">
17   Silahkan klik saya
18 </p>
19 </body>
20 </html>

```



Silahkan klik saya

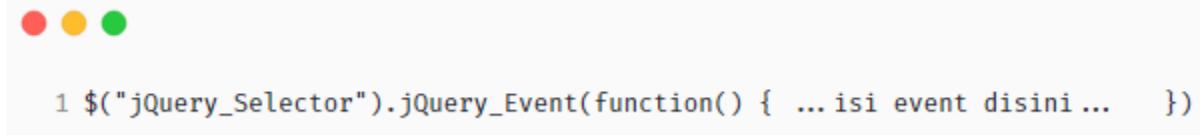
An embedded page on this page says

Saya sudah di klik!

OK

Gambar 53. Contoh penulisan jQuery Event dan Hasilnya

Perhatikan bagaimana jQuery memudahkan penulisan event ini, sangat sederhana. Format penulisannya adalah:



```
1 $("jQuery_Selector").jQuery_Event(function() { ... isi event disini ... })
```

Selain event click, juga tersedia berbagai event lain, seperti dblclick, mouseenter, mouseover, mouseleave, dll.

Mengambil Nilai HTML dengan jQuery

Mengambil nilai dari sebuah element HTML cukup sering dilakukan. Misalkan terdapat membuat kotak isian form `<input type="text">`. Bagaimana cara mengambil nilai yang sudah diisi menggunakan JavaScript?. Dengan JavaScript 'murni' kita bisa menggunakan property `innerHTML`. Namun caranya cukup panjang karena harus digabung dengan fungsi `document.getElementById()`.

Mengambil nilai HTML dengan Method `text()` jQuery

Metode pertama untuk mengambil nilai sebuah elemen HTML dengan jQuery adalah menggunakan method **text()**. Caranya cukup menambahkan method/fungsi **text()** ke jQuery Selector.

The screenshot shows a browser window with the following content:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Belajar jQuery </title>
6 <script src="jquery-2.1.4.js"></script>
7 <script>
8     $(document).ready(function() {
9         $("#tombol").click(function() {
10            var nilai = $("#paragraf").text();
11            alert(nilai);
12        })
13    });
14 </script>
15 </head>
16 <body>
17 <p id="paragraf">
18     Sedang belajar <em>jQuery</em>
19 </p>
20 <button id="tombol">Ambil Nilai</button>
21 </body>
22 </html>
```

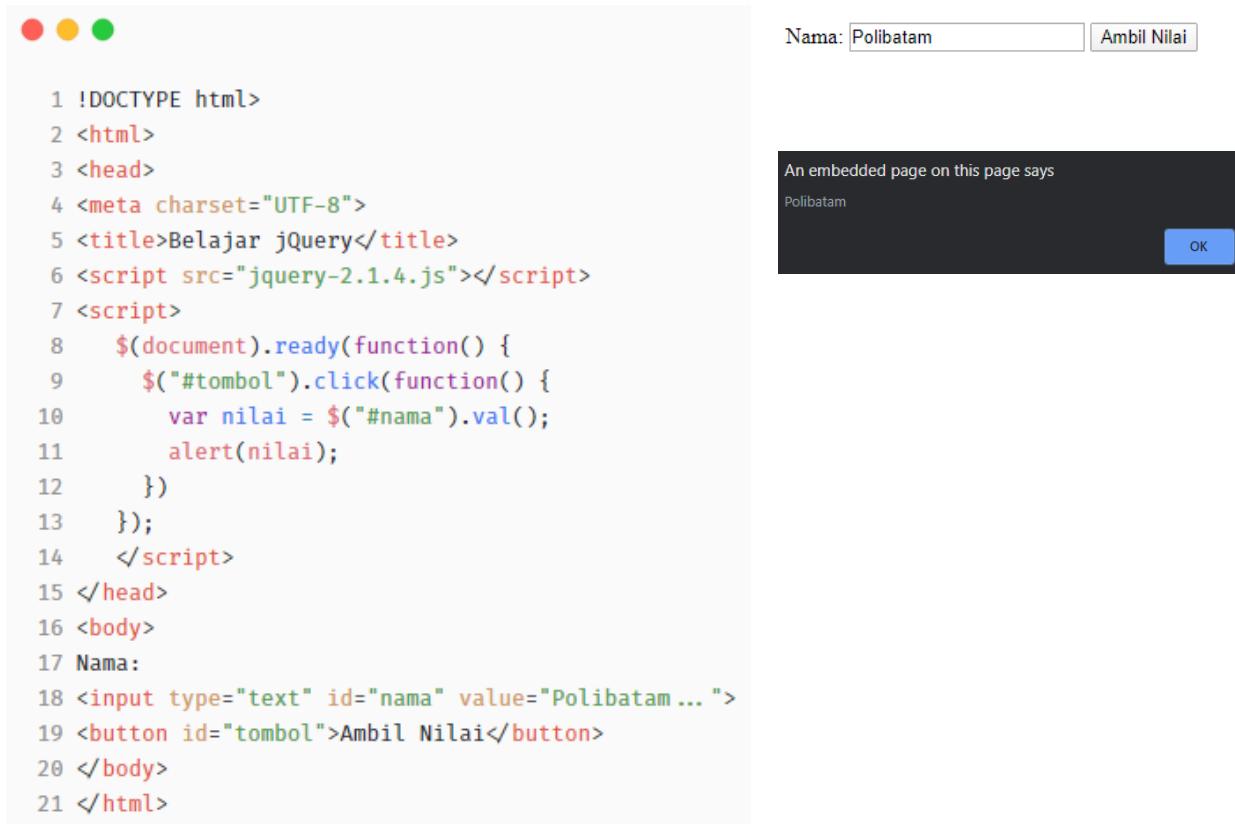
The page contains a button labeled "Ambil Nilai". When clicked, an alert box appears with the text "Sedang belajar jQuery".

Gambar 54. Contoh Mengambil nilai HTML dengan Method `text()` jQuery dan Hasilnya

Untuk mengambil nilai teks yang ada didalam tag `<p id="paragraf">`, kita bisa menggunakan perintah `$("#paragraf").text()`. Selanjutnya nilai ini disimpan ke dalam variabel nilai, lalu ditampilkan dengan fungsi `alert(nilai)`.

Mengambil nilai form HTML dengan method `val()` jQuery

Baik method **text()** maupun **html()** sudah mencukupi untuk mengambil nilai yang ada disetiap element HTML. Tapi khusus untuk form, jQuery menyediakan method **val()**. Langsung saja kita lihat contoh penggunaannya:



```

1 !DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Belajar jQuery</title>
6 <script src="jquery-2.1.4.js"></script>
7 <script>
8     $(document).ready(function() {
9         $("#tombol").click(function() {
10            var nilai = $("#nama").val();
11            alert(nilai);
12        })
13    });
14 </script>
15 </head>
16 <body>
17 Nama:
18 <input type="text" id="nama" value="Polibatam ... ">
19 <button id="tombol">Ambil Nilai</button>
20 </body>
21 </html>

```

Gambar 55. Contoh Mengambil nilai HTML dengan Method val() jQuery dan Hasilnya

Kali ini terdapat sebuah tag `<input type="text">`. Apabila kita klik tombol “Ambil Nilai”, atau ubah teks yang ada di dalam inputan form tersebut. Perintah `$("#nama").val()` akan menampilkan nilai apapun yang di input ke dalam isian form tersebut.

Untuk checkbox perlu penyeleksian lebih lanjut, karena jika menggunakan method val() secara langsung, nilai form akan dikembalikan apapun kondisi checkbox (baik sudah dipilih maupun belum). Untuk mengatasinya ini, kita bisa menggunakan selector khusus jQuery, yakni `:checked`.

```

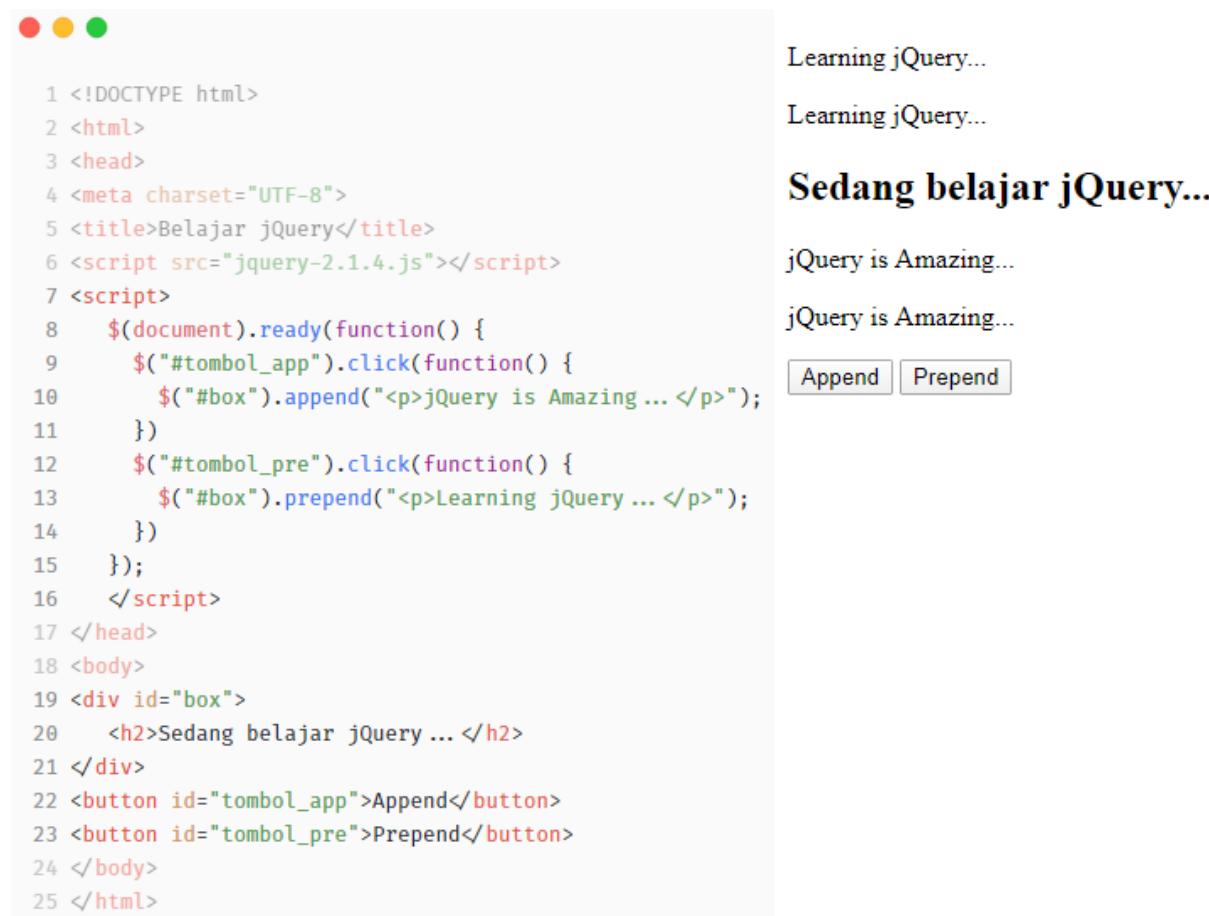
1 <script>
2     $(document).ready(function() {
3         $("#tombol").click(function() {
4             var nilai1 = $("#keren:checked").val();
5             var nilai2 = $("#ganteng:checked").val();
6             var nilai3 = $("#cool:checked").val();
7             alert(nilai1+ ' '+nilai2+ ' '+nilai3);
8         })
9     });
10    </script>

```

Menambah Elemen HTML dengan jQuery

Method `append()` dan `prepend()` bisa digunakan untuk menambah elemen HTML baru ke dalam kode HTML yang sudah ada. Method `append()` akan menambahkan elemen baru ke bagian akhir, sedangkan `prepend()` akan menambahkan elemen HTML baru ke bagian awal.

Yang perlu dicatat, kedua fungsi ini menambahkan element baru ke dalam selector yang dicari. Berikut contohnya:



The screenshot shows a browser window with three tabs. The active tab is titled "Sedang belajar jQuery...". Inside the tab, there is a heading "Sedang belajar jQuery..." and two paragraphs: "jQuery is Amazing..." and "Learning jQuery...". Below the heading are two buttons: "Append" and "Prepend". To the left of the browser window, the source code for this page is displayed in a code editor.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Belajar jQuery</title>
6 <script src="jquery-2.1.4.js"></script>
7 <script>
8     $(document).ready(function() {
9         $("#tombol_app").click(function() {
10             $("#box").append("<p>jQuery is Amazing ... </p>");
11         })
12         $("#tombol_pre").click(function() {
13             $("#box").prepend("<p>Learning jQuery ... </p>");
14         })
15     });
16 </script>
17 </head>
18 <body>
19 <div id="box">
20     <h2>Sedang belajar jQuery ... </h2>
21 </div>
22 <button id="tombol_app">Append</button>
23 <button id="tombol_pre">Prepend</button>
24 </body>
25 </html>
```

Gambar 56. Contoh Menambah Elemen HTML dengan jQuery dan Hasilnya

Pada kode program diatas, terdapat sebuah tag `<div>` dengan `id="box"`. Di dalamnya terdapat sebuah tag `<h2>`. Ketika tombol “Append” di klik, akan dijalankan perintah berikut:

```
1 $("#box").append("<p>jQuery is Amazing ... </p>");
```

ini maksudnya, tambahkan kode “`<p>jQuery is Amazing...</p>`” ke bagian akhir tag `<div id="box">`. Begitu juga ketika tombol “Prepend” di klik, perintah:

```
1 $("#box").prepend("<p>Learning jQuery ... </p>");
```

Akan menambah sebuah paragraf baru ke dalam tag `<div id="box">`, pada posisi paling awal.

Menghapus Elemen HTML dengan jQuery

Kita bisa menggunakan method `remove()` jQuery untuk menghapus element HTML yang sudah tampil di halaman saat ini. Method `remove()` akan menghapus tag saat ini, beserta seluruh isinya.



[Hapus HTML](#) [Hapus Semua!](#)

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Belajar jQuery</title>
6 <script src="jquery-2.1.4.js"></script>
7 <script>
8     $(document).ready(function() {
9         $("#hapus_html").click(function() {
10            $("#list_html").remove();
11        })
12        $("#hapus_all").click(function() {
13            $("#belajar").remove();
14        })
15    });
16 </script>
17 </head>
18 <body>
19 <button id="hapus_html">Hapus HTML</button>
20 <button id="hapus_all">Hapus Semua!</button>
21 <ul id="belajar">
22     <li id="list_html">Belajar HTML</li>
23     <li id="list_css">Belajar CSS</li>
24     <li id="list_php">Belajar PHP</li>
25 </ul>
26 </body>
```

- Belajar HTML
- Belajar CSS
- Belajar PHP

Gambar 57. Contoh Menghapus Elemen HTML dengan jQuery dan Hasilnya

Disini terdapat 2 tombol dan sebuah list dengan 3 tag ``. Baik tag `` maupun tag `` dari list memiliki atribut id masing-masing.

Ketika tombol Hapus HTML di klik, perintah: `$("#list_html").remove()` akan dijalankan. Akibatnya, list `<li id="list_html">Belajar HTML` akan terhapus. List yang tersisa akan mengisi element yang sudah hilang ini (naik ke atas).

Tombol Hapus Semua akan menjalankan perintah `$("#belajar").remove()`. Elemen HTML yang memiliki `id="belajar"` adalah tag ``. Element ini merupakan ‘parent element’ dari seluruh tag ``. Dengan demikian, semua list akan ikut terhapus.

VII USER INTERFACE

Dalam beberapa tahun terakhir istilah UI/UX mulai ramai diperbincangkan. UI/UX sendiri mempunyai satu kesatuan yang tidak bisa lepas, karena User Interface akan bepengaruh pada User Experience itu sendiri .

banyak diantara pengembang mulai melirik dan memperhatikan istilah ini karena sejatinya kepuasan dan memenuhi kebutuhan pengguna adalah prioritas utama bagi pengembang sebuah aplikasi.

Pengertian User Interface

User Interface adalah hubungan manusia dan komputer dalam suatu perangkat baik layar tampilan, keyboard, mouse, dan desktop. Namun, menurut ahli, User Interface adalah cara program dan pengguna berinteraksi. Dalam istilah User Interface terkadang digunakan sebagai pengganti istilah Hubungan manusia dan Komputer atau Human Computer Interaction (HCI) yang mana semua aspek saling berhubungan.

User interface adalah bagian visual dari website, aplikasi software atau device hardware yang memastikan bagaimana seorang user berinteraksi dengan aplikasi atau website tersebut serta bagaimana informasi ditampilkan di layarnya. User interface sendiri menggabungkan konsep desain visual, desain interasi, dan infrastruktur informasi. Tujuan dari user interface adalah untuk meningkatkan usability dan tentunya user experience.



Gambar 58. Contoh User interface pada berbagai device

Seperti namanya, (UI) User Interface. Maka ada 2 hal yang harus diperhatikan oleh developer saat membuat UI, yaitu User dan Interface. berikut gambarannya



Gambar 59. User dan Interface

Untuk menghasilkan UI (User Interface) yang baik, harus memperhatikan 2 hal diatas. Kata User ditempatkan di depan kata Interface, karena UI yang baik selalu memperhatikan dan mengutamakan user. UI yang baik akan membantu user. Dan, UI yang baik akan membuat user nyaman menggunakananya.

Mengapa User Interface itu Penting?

User interface adalah salah satu faktor yang menentukan peningkatan traffic website. Desain user interface sangat penting karena akan menentukan bagaimana seseorang berinteraksi dengan website dan aplikasi tersebut dengan mudah.

UI bisa menjadi faktor apakah seseorang tertarik untuk mengunjungi dan meng-explore sebuah website. Kalau mereka senang dengan tampilan interface website tersebut dan experience saat menggunakan website tersebut, maka kemungkinan besar mereka akan mengunjungi website tersebut kembali. Jadi saat me-develop sebuah website harus selalu memperhatikan UI dan UX. Berikut beberapa alasan mengapa user interface itu penting:

- **Penghubung Langsung**
User interface memiliki peran penting pada sebuah aplikasi. Karena user interface adalah penghubung langsung antara pengguna (kita) dengan sistem aplikasi itu sendiri. Yang membuat sebuah sistem bisa berjalan dan dikendalikan oleh pengguna karena adanya user interface.
- **Berdampak Pada Faktor Kesuksesan**
Bagus tidaknya user interface pada sebuah aplikasi, bisa berdampak pada kesuksesan aplikasi tersebut. Jika user interfacenya mudah digunakan akan ada kemungkinan aplikasi banyak dan sering digunakan, jika sebaliknya user interfacenya menyulitkan maka besar kemungkinan aplikasi itu akan ditinggalkan.
- **Identitas Atau Ciri Khas**
User interface bisa dijadikan sebagai identitas atau ciri khas dari sebuah produk. Jika Kamu pernah menggunakan smartphone android, IOS (Apple dan windows phone) Kamu pasti tau bedanya. Ketiga sistem operasi tersebut memiliki ciri khas desain user interface masing-masing. Dan desain tersebut dibuat secara konsisten dari mulai versi lama sampai versi terbaru.
- **Berpengaruh Pada SEO (Pada Website)**

Apakah user interface itu bisa berpengaruh pada seo? Benarkah? User interface tidak hanya membahas soal indah dan mudahnya digunakan sebuah aplikasi, pada website kecepatan juga masuk dalam bagian user interface. Kecepatan sebuah website saat dibuka akan berpengaruh pada pengunjung yang datang. Makin cepat dibuka makin besar kemungkinan pengunjung sering datang, makin lama dibuka makin besar kemungkinan ditinggalkan.

- **Menarik Minat Pengguna**

Dulu sebuah desain user interface haruslah mudah digunakan itu sudah cukup. Sekarang mudah digunakan saja tidaklah cukup, desain harus bisa menarik minat penggunanya. Menarik dalam artian tidak harus yang wah, namun cukup memberi kesan menyenangkan saat digunakan ke pengguna.

Karakteristik User Interface yang Baik

Di luar sana ada berbagai informasi yang menyarankan teknik dan pola desain interface yang dianggap baik dan benar. Memang kalau kita mengikuti saran-saran tersebut, kemungkinan besar kita akan membuat website atau aplikasi dengan UI yang baik. Tetapi ada beberapa hal yang perlu kita checklist karakteristik apa saja yang sebenarnya membuat sebuah UI dibilang bagus, yaitu:

- **Jelas**

Memiliki UI yang jelas adalah salah satu elemen penting dalam desain user interface. Tentunya tujuan dari desain UI adalah agar orang-orang bisa menggunakan dan berinteraksi dengan sistem kita dengan mudah. Jika orang-orang tidak bisa mengerti bagaimana cara menggunakan dan menavigasi website kita mereka pasti akan bingung.

- **Singkat**

Tentunya memiliki UI yang jelas akan membuat UX kita bagus, tetapi kita juga harus hati-hati agar penjelasan tidak terlalu panjang. Kalau kita meletakkan definisi dan penjelasan pada setiap bagian website, maka akan terlihat lebih berantakan. UI boleh jelas, tapi juga harus singkat. Kalau memang diperlukan penjelasan, usahakan agar kita bisa menjelaskannya dalam satu kalimat.

- **Familiar**

Familiar yang dimaksud disini adalah sesuatu yang sudah pernah dilihat sebelumnya. Kalau kita familiar dengan sesuatu, kita pasti tahu kan apa yang harus dilakukan? Coba pikirkan hal-hal apa yang sudah pasti familiar bagi user website kita dan masukkan hal-hal ini ke desain web.

- **Responsif**

Arti lain dari responsive pada UI juga harus bisa memberitahu user apa yang sedang terjadi di halaman itu. Misalnya, jika seorang user mengklik suatu tombol di website, apakah mereka sudah berhasil menekan tombol tersebut? Mungkin text pada tombol bisa menjadi kata “loading” jika mereka sudah berhasil menekan tombolnya.

- **Konsisten**

Dalam mengembangkan user interface, konsistensi pada interface dapat membantu user untuk mengerti pola. Dari satu interface, mereka bisa mempelajari apa kegunaan tombol, tabs, icons, dan berbagai elemen yang ada pada interface tersebut.

- **Menarik**

Yang dimaksud dengan menarik disini adalah interface pada website menarik untuk digunakan. Tentunya apa yang dianggap menarik untuk website atau aplikasi Anda juga perlu disesuaikan dengan pasar dan audience website. Jadi tampilan harus menarik sesuai dengan audience. Tetapi juga harus tetap memerhatikan fungsi website agar mereka tetap berfungsi dengan baik.

- **Efisien**

User interface yang baik harus memastikan bahwa website dan aplikasinya bisa digunakan dengan efisien. Agar kita bisa membuat UI yang efisien, maka perlu tahu dulu apa yang ingin user capai dan biarkan mereka melakukan langkah-langkahnya tanpa banyak masalah

Rancangan User Interface (UI)

Rancangan UI adalah proses untuk membuat antarmuka dalam sebuah perangkat lunak atau perangkat terkomputerisasi dengan fokus pada penampilan atau gaya. Dalam siklus-hidup pengembangan sistem/aplikasi, merancang User Interface merupakan bagian dari tahap “Desain”. Output dari proses merancang User Interface adalah dokumen spesifikasi desain. Dokumen Spesifikasi Desain terdiri dari: *Iktisar naratif, Contoh Desain, dan Pengujian dan evaluasi kegunaan*.

Dokumen spesifikasi ini merupakan dokumen spesifikasi yang harus ada dalam setiap pembuatan aplikasi/perangkat lunak.

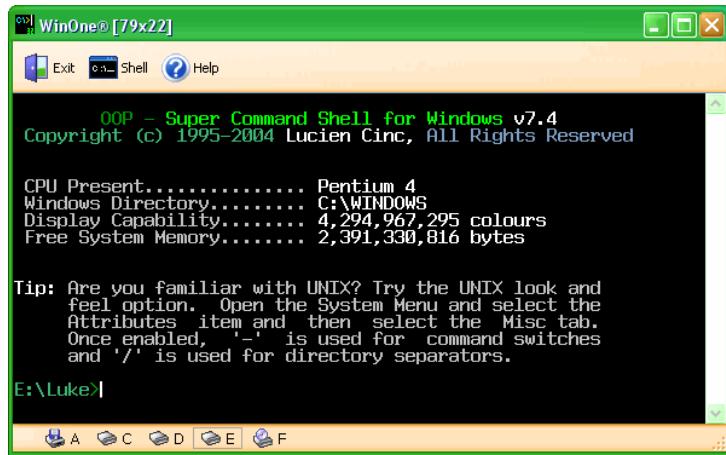
Design Specification
<ol style="list-style-type: none">1. Narrative Overview<ol style="list-style-type: none">a. Interface/Dialogue Nameb. User Characteristicsc. Task Characteristicsd. System Characteristicse. Environmental Characteristics2. Interface/Dialogue Designs<ol style="list-style-type: none">a. Form/Report Designsb. Dialogue Sequence Diagram(s) and Narrative Description3. Testing and Usability Assessment<ol style="list-style-type: none">a. Testing Objectivesb. Testing Proceduresc. Testing Results<ol style="list-style-type: none">i) Time to Learnii) Speed of Performanceiii) Rate of Errorsiv) Retention over Timev) User Satisfaction and Other Perceptions

Gambar 60. Joseph Valacich, Joey George; Modern Systems Analysis and Design

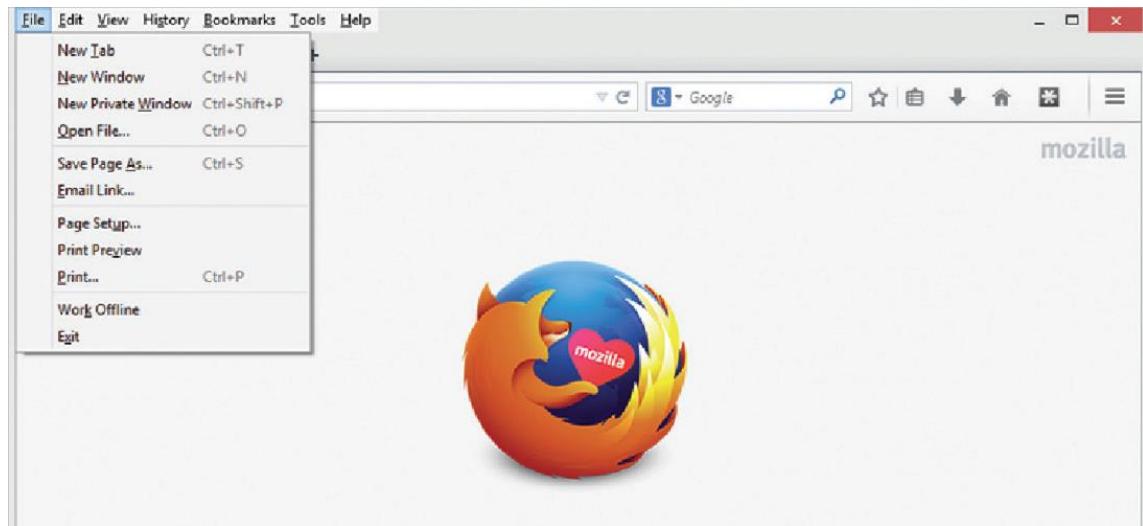
Beberapa Bentuk Interaksi Pengguna dan Aplikasi

Secara umum terdapat 5 bentuk interaksi antara pengguna dan aplikasi:

- **Command language**, tipe antarmuka dimana user berinteraksi dengan sistem operasi melalui text-terminal. Pengguna menjalankan perintah dan program di sistem operasi tersebut dengan cara mengetikkan perintah di tiap baris-baris tertentu.



- **Menu Interaction** memberikan beberapa pilihan kepada user berbentuk daftar menu yang isinya adalah perintah kepada komputer sesuai dengan menu yang terdaftar, nama perintah yang dijalankan harus sesuai dengan nama pilihan menu.



Menus adalah sebuah cara dari sistem untuk mempresentasikan pilihan-pilihan instruksi kepada user dimana tiap instruksi dapat menyebabkan perubahan kondisi interface. Seleksi bisa dilakukan dengan Cursor key, mouse click, maupun Alternate (Alt) yang ditambah dengan kode huruf atau angka.

- **Form interaction**, cara sebuah sistem untuk meminta data atau informasi dari user dengan member petunjuk kepada user untuk mengisi area area dan melengkapi data pada kategori dan bagian bagian yang telah disediakan oleh form.

Advanced Search

Find pages with...

all these words:

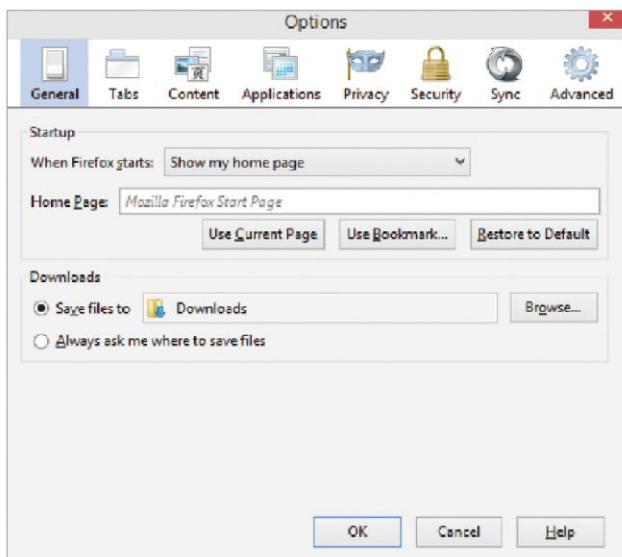
this exact word or phrase:

any of these words:

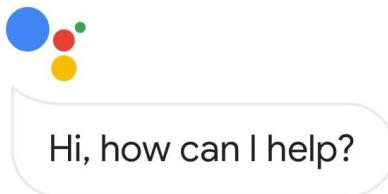
none of these words:

numbers ranging from: to

- **Interaksi langsung dengan objek** pada layar sehingga aktivitas akan dikerjakan oleh komputer ketika pengguna memberikan instruksi langsung yang ada pada layar komputer. Mewakili suatu objek dalam antarmuka, dapat merupakan suatu lambang dari sebuah aplikasi atau tindakan.



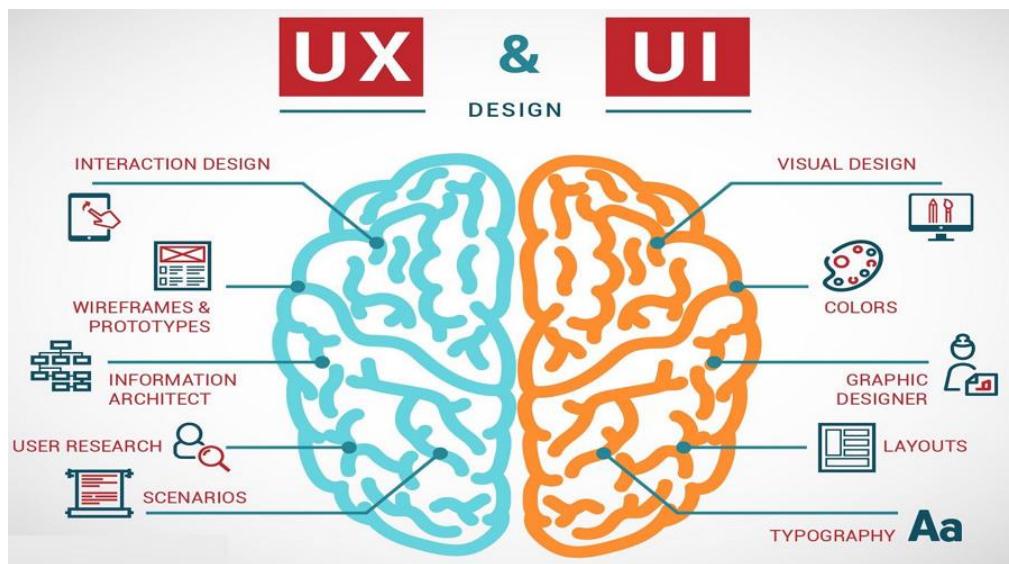
- **Perintah dengan menggunakan Bahasa manusia** yang digunakan manusia untuk berkomunikasi sehari-hari. Program yang dapat menerima perintah menggunakan natural language dapat disebut sebagai artificial intelligence.



User Interface (UI) dan User Experience (UX)

UI adalah desain antarmuka yang lebih memfokuskan pada keindahan dari sebuah tampilan, pemilihan warna yang baik dan pas dan hal-hal lainnya yang membuat tampilan web ecommerce yang kita miliki lebih enak dipandang mata dan membuat pengujung betah berlama-lama.

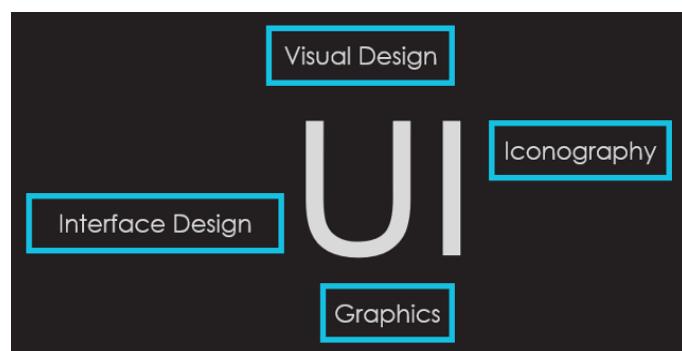
UX merupakan desain yang digunakan untuk meningkatkan kepuasan dari pengguna website melalui kesenangan dan kegunaan yang diberikan dalam interaksi antara pengguna internet atau pengunjung dan produk. UX inilah yang berfungsi untuk membuat website ecommerce anda menjadi lebih mudah dan tidak membingungkan pengguna.



Gambar 61. UI vs UX

UI lebih mementingkan tampilan permukaan dan keseluruhan nuansa desain. UX lebih mencakup kepada seluruh spektrum pengalaman pengguna. Jika kita ingin agar pengguna menyukai perangkat lunak kita, kita harus mendesainnya agar berperilaku seperti yang disukai orang: penuh respek, murah hati, dan bermanfaat". (Alan Cooper, software designer and programmer)

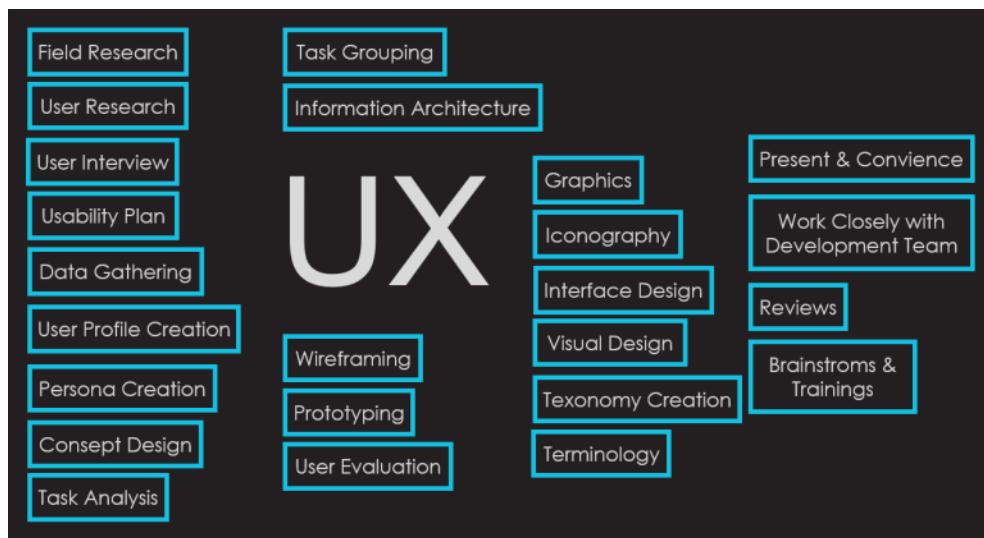
Ranah utama dari **user interface** yang terdiri dari *interface design, grafis, icon, dan visual design*. User interface designer akan mampu memproduksi 4 hal diatas dengan sangat baik. Tujuan utama dari UI adalah menampilkan interface yang memiliki keseragaman yang baik dari segi warna, font, gambar dll.



Gambar 62. Ranah pada UI

UX Design adalah proses untuk membuat product yang telah kita buat menjadi mudah untuk digunakan dan tidak membingungkan ketika digunakan oleh user. User experience memiliki ranah yang lebih luas dari

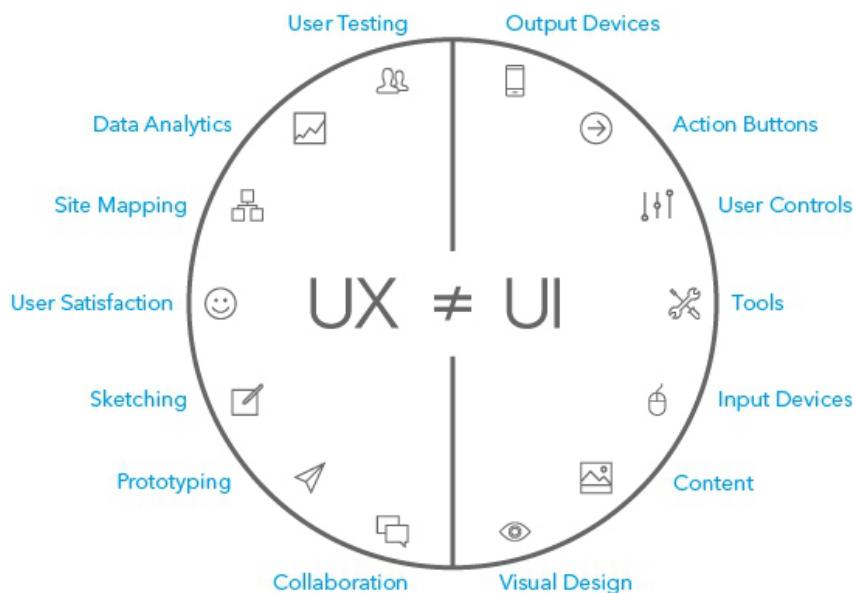
UI, karena ranah UX ini dimulai dengan research pasar sampai kemudian diimplementasi kedalam sebuah interface.



Gambar 63. Ranah pada UX

Perbedaan UI dan UX

Beda UI dan UX adalah dari fungsinya, jika UI lebih berfungsi untuk membuat desain web ecommerce menjadi lebih indah. UX merupakan desain yang digunakan untuk meningkatkan kepuasan dari pengguna website melalui kesenangan dan kegunaan yang diberikan dalam interaksi antara pengguna internet atau pengunjung dan produk.



Gambar 63. Perbedaan UX dan UI

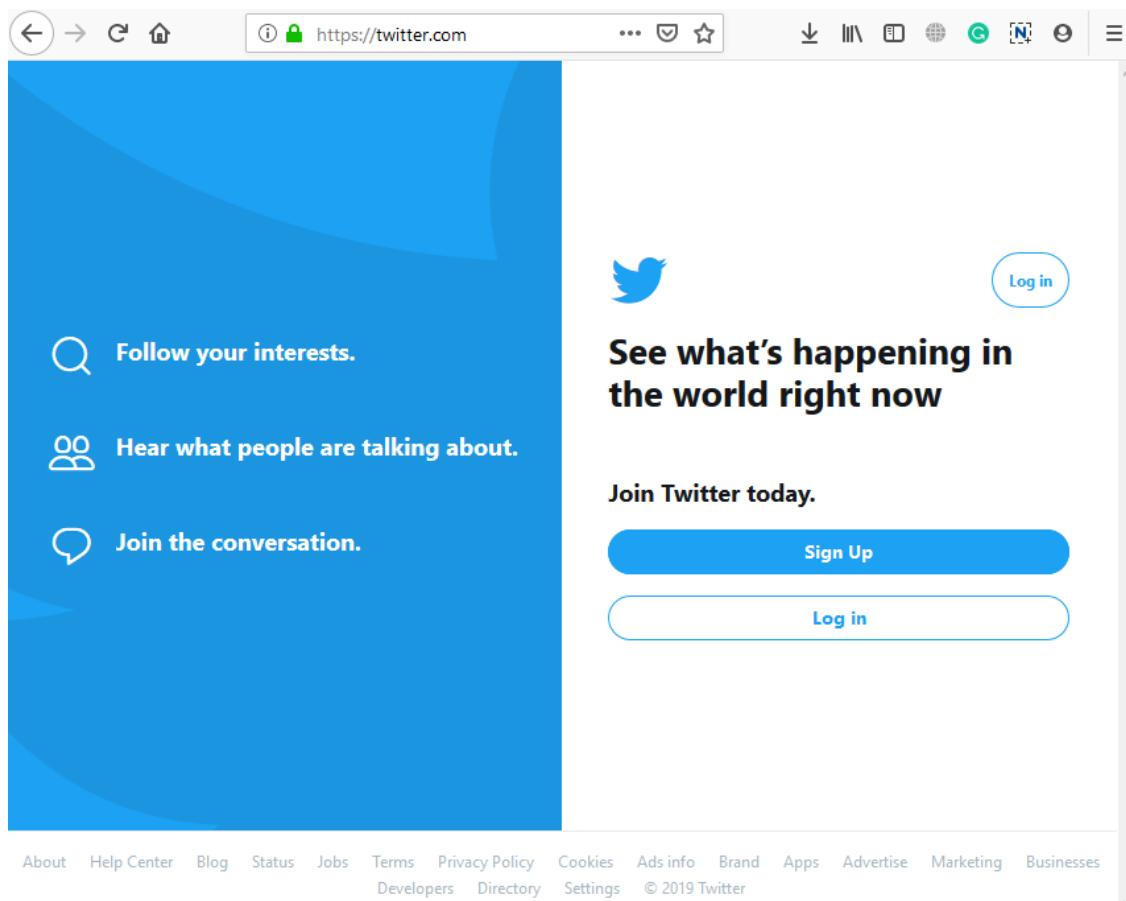
Prinsip Utama Desain UI

1. **User compatibility:** Antarmuka merupakan cara masuk ke sebuah sistem.

2. **Product compatibility:** Aplikasi yang sesuai dengan sistemnya pada interface berbeda dengan sistem yang manual yang menghasilkan hasil yang lainnya.
3. **Task compatibility:** Aplikasi yang membantu para pemakai dalam menyelesaikan pekerjaannya.
4. **Work flow compatibility:** Sistem manual yang terdapat langkah kerjanya dalam hal penyelesaian tugas.
5. **Consistency:** Sistem yang konsisten dengan sistem yang ada sesuai barangnya yang di dasarkan kepada setiap sifat dari semua orang yang berbeda.
6. **Familiarity:** Memberikan tampilan yang terkesan kepada pemakai.
7. **Simplicity:** Memberikan sistem sederhana yang tidak membuat si pemakai kebingungan.

Contoh Familiarity

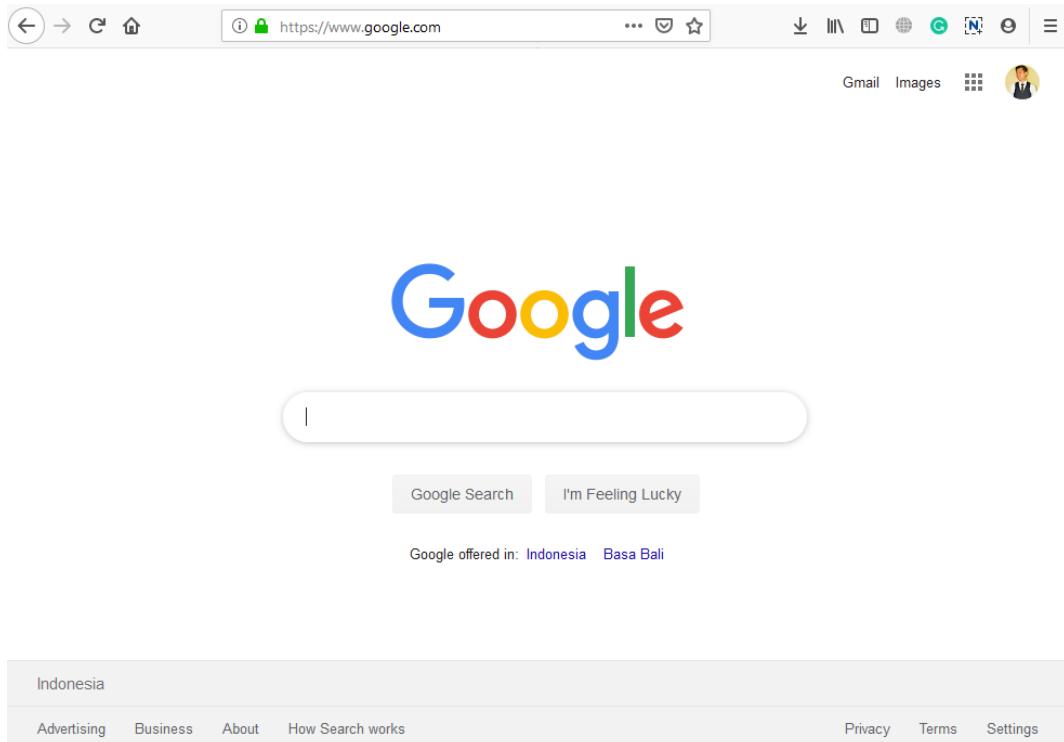
Memberikan tampilan yang familiar terhadap user. Tampilan awal twitter.com sangatlah familiar di mata user dan tidak membingungkan.



Gambar 63. Contoh Familiarity

Contoh Simplicity

Kesederhanaan perlu diperhatikan pada saat membangun antarmuka. Tidak selamanya antarmuka yang memiliki menu banyak adalah antarmuka yang baik. Kesederhanaan di sini lebih berarti sebagai hal yang ringkas dan tidak terlalu berbelit. User akan merasa jengah dan bosan jika pernyataan, pertanyaan dan menu bahkan informasi yang dihasilkan terlalu panjang dan berbelit. User lebih menyukai hal-hal yang bersifat sederhana tetapi mempunyai kekuatan/ bobot.



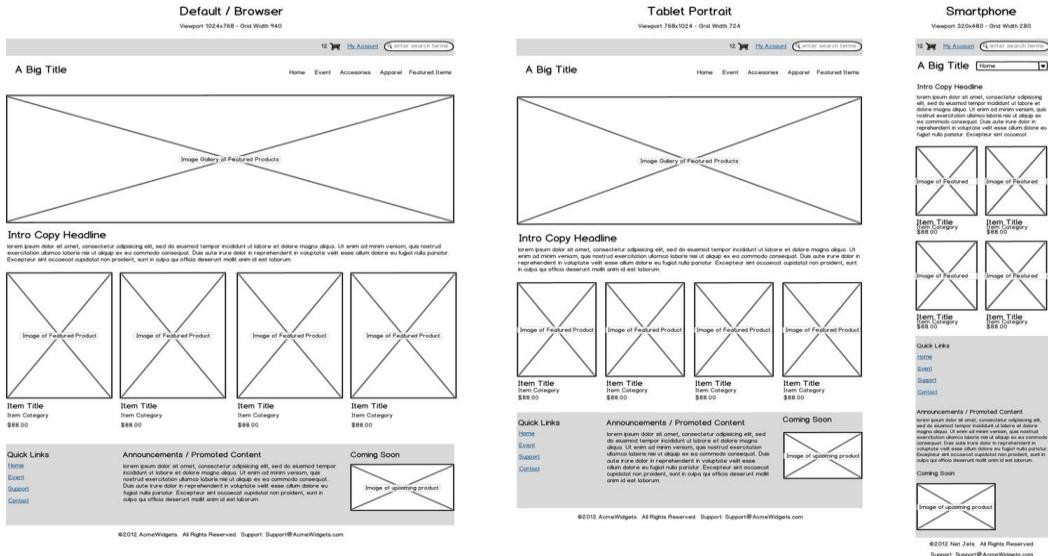
Gambar 63. Contoh Simplicity

Wireframing

Mockup/Wireframe adalah rancangan awal sebuah desain web yang dibuat secara manual menggunakan photoshop, atau software pengedit gambar lainnya. Mockup merupakan gambar model atau prototif halaman web secara full dan detail.

Wireframe adalah kerangka dasar/blueprint dari halaman web yang akan kita bangun. Secara garis besar di dalam wireframe ini kita menempatkan elemen-elemen penting dari halaman web tersebut pada posisinya masing-masing seperti banner, body content, menu link, kolom, footer maupun fitur-fitur lainnya yang ada dalam web nantinya. Pertimbangan Pemanfaatan Wireframe:

- Dengan membuat wireframe kita membantu client untuk fokus pada kerangka utama dari membangun halaman web tersebut.
- Dengan wireframe yang hanya berupa kotak hitam-putih akan lebih mudah bagi kita untuk mendeteksi apa yang tidak bekerja dari sisi usability dan fungsionalitas.
- Minimalisasi revisi



Gambar 63. Contoh wireframe

Tugas pertama web designer itu adalah membuat rancangan layout website, misalnya sekedar membuat wireframe dengan coret-coret di kertas atau dengan tools atau mungkin langsung membuat di photoshop.

Penerapan Prinsip Design pada Wireframe

Terdapat empat prinsip tersebut yang menjadi dasar dari sebuah desain akan menjadi suatu desain yang baik ataupun tidak, yaitu: *Contrast (Kontras)*, *Repetition (Pengulangan)*, *Alignment (Penajaran)* dan *Proximity (Kedekatan)*.

Contrast

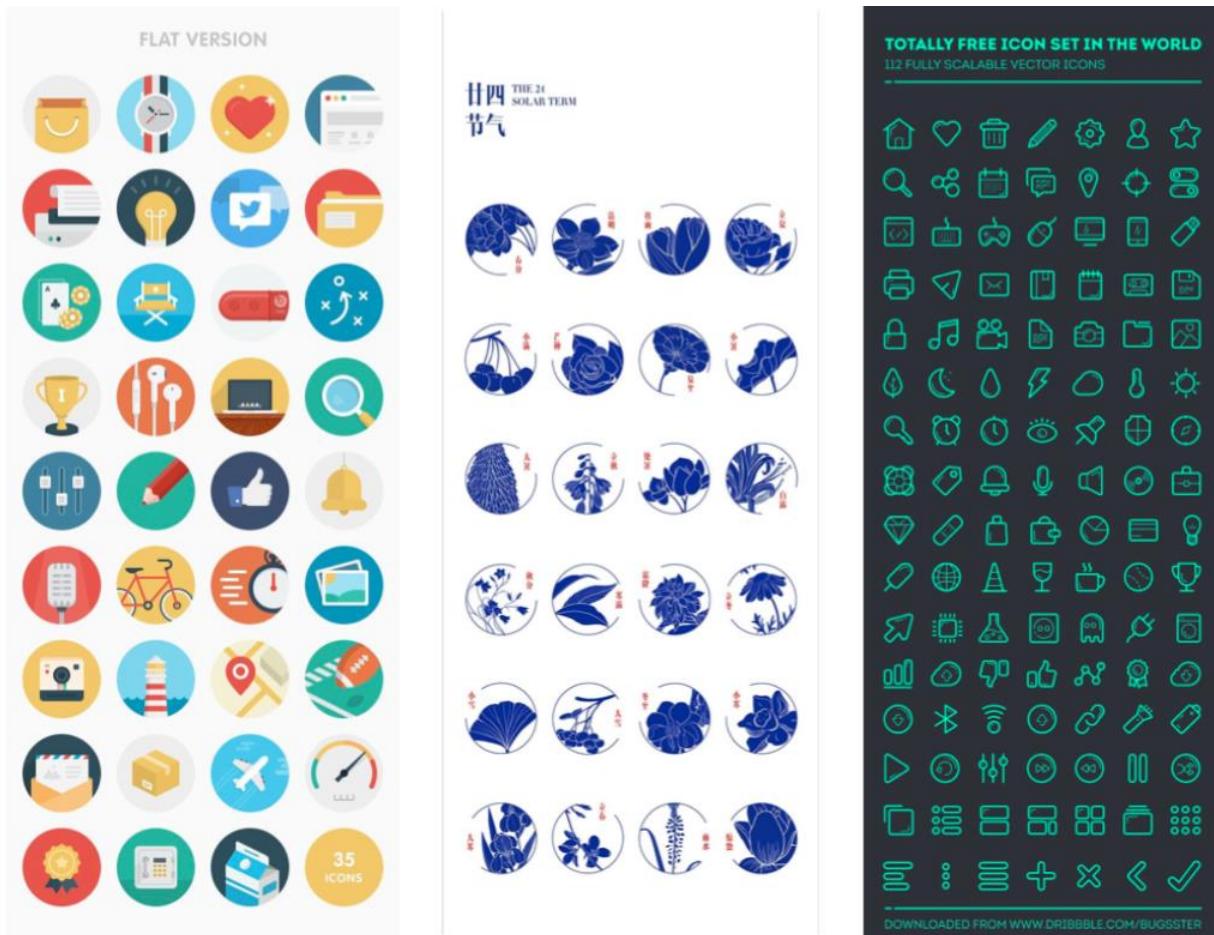
Kontras yang baik dalam desain; yang mana dapat dicapai dengan menggunakan elemen-elemen seperti warna, corak, ukuran dan lain sebagainya; yang memberikan pandangan pengguna mengalir secara natural. *When color used to code redundant information, performance enhanced.*



Gambar 64. Contoh contrast

Repetition

Repetisi dalam desain adalah melakukan pengulangan elemen yang sama secara bertautan atau saling mempunyai hubungan atau kesamaan. Singkatnya, Bentuklah suatu gaya/corak pada tiap elemen dalam desain dan gunakan pada elemen-elemen yang serupa.



Style + Repetition + Consistency = Identification > Branding

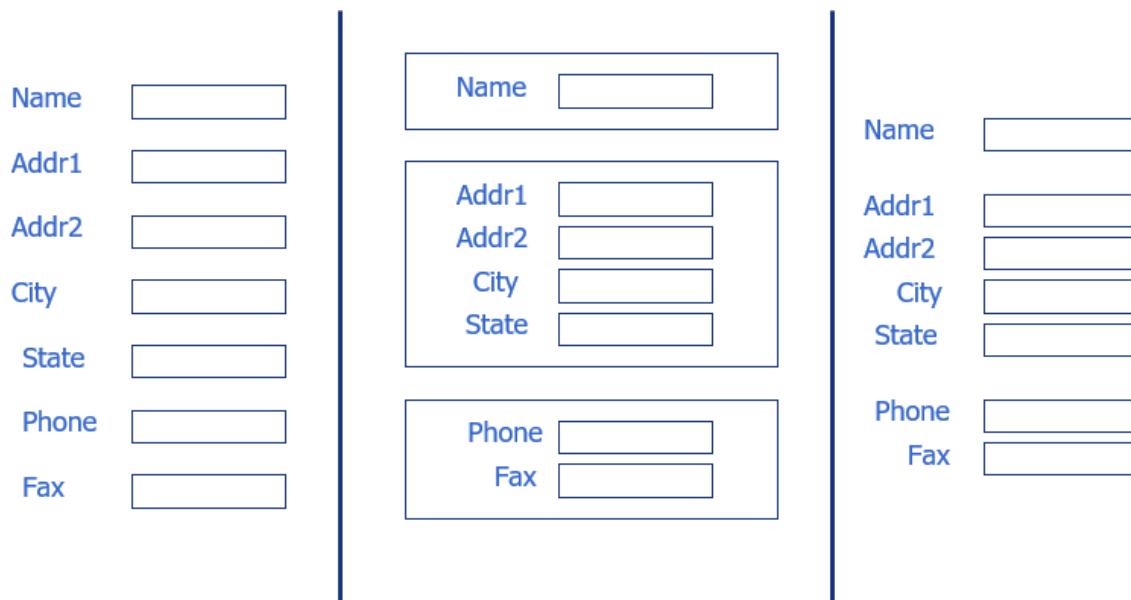
Gambar 64. Contoh Repetition

Alignment

Penjajaran yang baik dalam sebuah desain yaitu dimana setiap elemen saling terhubung secara visual dengan elemen lainnya. Penjajaran memberikan kesan saling terkait. Penjajaran yang diaplikasikan dengan baik akan membawa orang yang melihat desain tidak perlu banyak mempelajari elemen-elemen dalam desain tersebut seharusnya dilihat.

Proximity

Kedekatan memberikan kesatuan visual dalam sebuah desain. Jika ada dua elemen yang saling berhubungan, seharusnya diletakkan berdekatan. Minimalisir kesemrawutan visual, pengorganisasian yang tegas dan tingkatkan pemahaman orang yang melihat desain.



Gambar 64. Contoh Proximity

7 Kebiasaan Desainer Interface yang Sukses

1. Pahami bisnisnya
2. Maksimalkan efektivitas grafikal
3. Berpikir seperti seorang pengguna
4. Gunakan model dan prototype (atau mock-up)
5. Fokus pada kegunaan
6. Sediakan feedback
7. Dokumentasikan segalanya

Beberapa Komponen UI (best practice)

- Tipografi
- Warna
- Icon
- Grid
- Tombol
- Label
- Table
- Alert
- Form Control & Form Template
- Kolom pencarian
- Navigasi
- Headers
- Footers
- Dialog

Beberapa Perangkat Lunak untuk Implementasi UI

- Adobe Dreamweaver (Commercial)
- Adobe Xd (Online)
- Pingendo (Freemium)
- Silex.me (Open Source)

- Google Web Designer (Free)
- Dan lain-lain

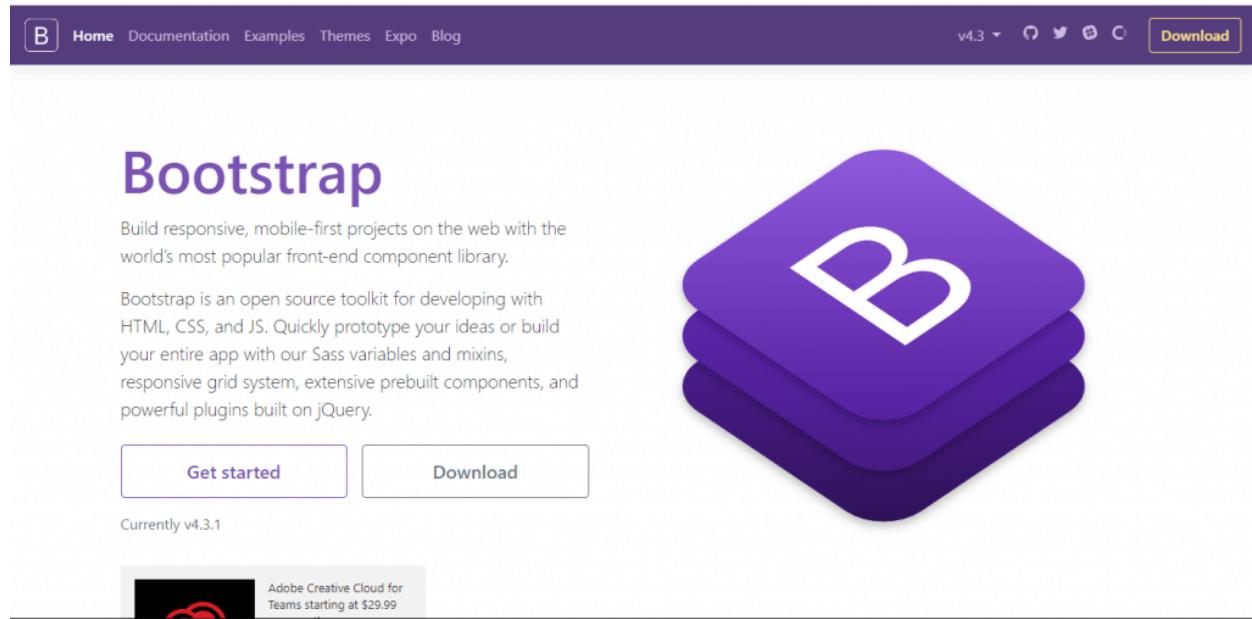
CSS Framework untuk Design Website

Untuk memudahkan dan mempercepat proses mendesain sebuah website, ada banyak tool yang dapat dimanfaatkan di antaranya adalah menggunakan CSS Framework. CSS Framework adalah pustaka CSS yang dimana sudah dibuat dan siap untuk digunakan. Dengan CSS Framework proses design website nantinya hanya tinggal menggunakan class-class yang sudah disediakan masing-masing CSS Framework.

Berikut CSS Framework populer yang bisa digunakan untuk membantu proses design sebuah website:

Bootstrap

Salah satu CSS Framework yang cukup populer adalah Bootstrap. Bootstrap adalah salah satu css framework open source untuk membuat design website dan design aplikasi web. Slogan dari framework ini adalah “*Sleek, intuitive, and powerful front-end framework for faster and easier web development*”, yang berarti kita dapat mendesain sebuah website dengan lebih rapi, cepat dan mudah. Bootstrap menyediakan komponen berupa layout tampilan, form, button dan typography.

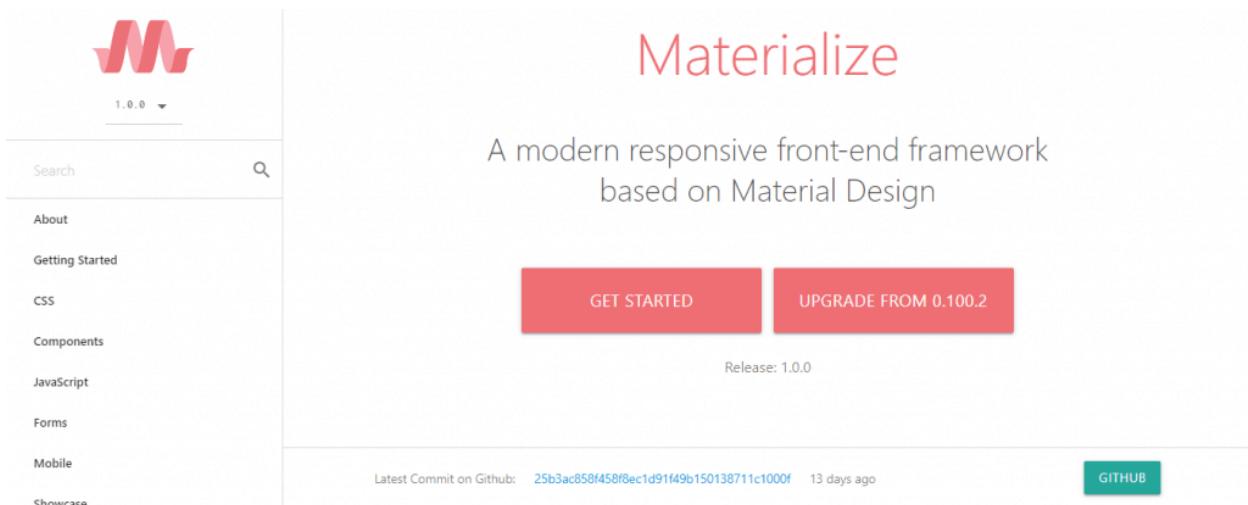


Gambar 65. Boostrap

Materialize

Framework besutan Google yang mengimplementasikan Google Material Design, salah satu yang populer dalam desain pemrograman. Framework ini mempunyai tampilan elegan dan sangat konsisten dan hampir terasa seperti tampilan android. Salah satu kelebihan dari framework ini adalah komponen-komponen serta class yang mudah untuk dipelajari.

Yang menarik dari framework ini adalah, Materialize diciptakan oleh empat mahasiswa berbeda di Carnegie Mellon University.



Gambar 66. Materialize

SemanticUI

SemanticUI pertama kali dirilis pada tahun 2013 dan diciptakan oleh Jack Lukic. Yang membedakan framework ini dari framework lain adalah banyaknya tema serta kustomisasi yang bisa digunakan. Ada sekitar 3000 lebih variasi tema untuk framework ini.



Gambar 66. SemanticUI

VIII BOOTSTRAP

Pengertian Bootstrap

Bootstrap adalah sebuah library framework CSS yang di buat khusus untuk bagian pengembangan front-end website. bootstrap merupakan salah satu framework HTML, CSS dan javascript yang paling populer di kalangan web developer. pada saat ini hampir semua web developer telah menggunakan bootstrap untuk membuat tampilan front-end menjadi lebih mudah dan sangat cepat. karena anda hanya perlu menambahkan class-class tertentu untuk misalnya membuat tombol, grid, navigasi dan lainnya.

Bootstrap telah menyediakan kumpulan komponen class interface dasar yang telah di rancang sedemikian rupa untuk menciptakan tampilan yang menarik, bersih dan ringan. selain komponen class interface, bootstrap juga memiliki fitur grid yang berfungsi untuk mengatur layout pada halaman website yang bisa digunakan dengan sangat mudah dan cepat. dengan menggunakan bootstrap kita juga di beri keleluasaan dalam mengembangkan tampilan website yang menggunakan bootstrap yaitu dengan cara mengubah tampilan bootstrap dengan menambahkan class dan CSS sendiri. Salah satu contoh website besar yang menggunakan framework bootstrap adalah Twitter

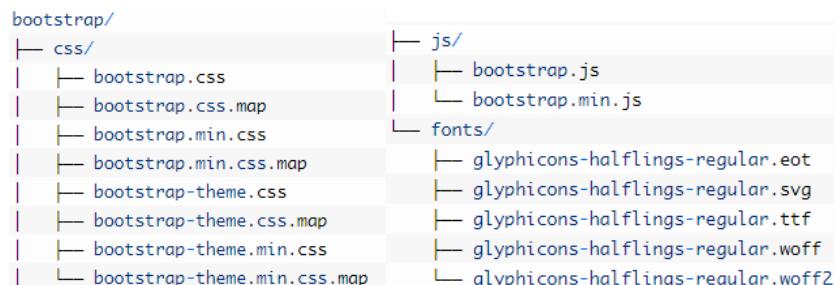
Kelebihan Bootstrap

Adapun beberapa kelebihan bootstrap adalah:

1. beberapa kelebihan bootstrap adalah sebagai berikut.
2. Penggunaan bootstrap sangat menghemat waktu.
3. Tampilan bootstrap yang sudah cukup terlihat modern.
4. Mobile Friendly yang maksudnya tampilan bootstrap sudah sangat responsive, yaitu tampilan bootstrap sudah mendukung segala jenis resolusi, baik itu pc, laptop, tablet dan smartphone.
5. Sangat ringan karena bootstrap di buat dengan sangat terstruktur.
6. Dan masih banyak lagi kelebihan dan kegunaan dari bootstrap yang akan anda temukan sendiri setelah mencoba membangun sebuah aplikasi berbasis web dengan menggunakan bantuan framework css bootstrap.

Memulai Menggunakan Bootstrap

Untuk cara menggunakan bootstrap yang pertama kali yang harus anda lakukan adalah melakukan instalasi bootstrap. anda dapat mendownload langsung bootstrap pada website resminya di <http://getbootstrap.com/getting-started/>. Untuk sampai saat ini bootstrap telah di rilis sampai versi 4. Berikut ini adalah struktur isi dari bootstrap setelah anda mendownloadnya.



Gambar 67. Struktur Isi Bootstrap

Cara menginstall bootstrap secara online

Untuk menginstall bootstrap secara online. syntax berikut bisa anda letakkan pada bagian <head> pada halaman HTML anda sama seperti menghubungkan HTML dengan CSS dan Javascript, atau copy script dari link berikut: <https://www.bootstrapcdncdn.com/>



```
1 <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js" integrity="sha384-J6qa484bIE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYf0RSJoZ+n" crossorigin="anonymous"></script>
2 <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvbIYzFJof+2mJbHaEWldlvI9IOY5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
3 <script src="https://stackpath.bootstrapcdncdn.com/bootstrap/4.4.1/js/bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uDj003uMBJuUD4Ih7YwaYd1iqfktj0Uod8GCEx130g8ifwB6" crossorigin="anonymous"></script>
```

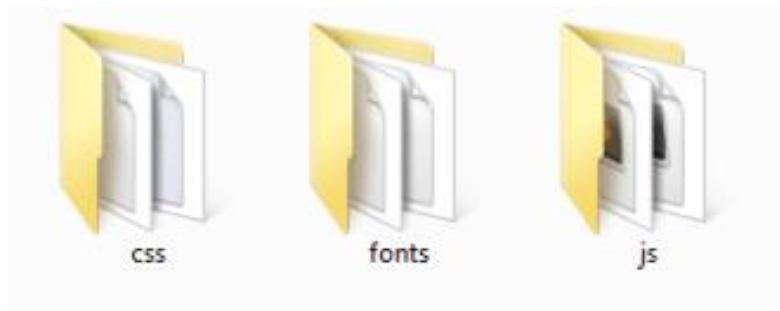
Gambar 68. CDN Boostrap

Cara menginstall bootstrap secara offline

Cara menginstall bootstrap secara offline ini sangat banyak di gunakan oleh pada developer, karena kita dapat bekerja menggunakan bootstrap tanpa harus terkoneksi dengan internet lagi, karena file framework bootstrap sudah terletak pada pc kita sendiri. Iain dengan penginstalan bootstrap secara online yang file-file bootstrap nya terletak pada server luar, setelah kita mendownload framework css bootstrap kemudian ekstrak pada project.

Kemudian untuk menghubungkan atau menginstall bootstrap buat sebuah file HTML atau PHP yang akan kita hubungkan dengan bootstrap. di sini kita membuat sebuah file dengan nama index.html.

Bootstrap juga memerlukan jquery karena untuk bagian javascript nya bootstrap menggunakan jquery, jadi jangan lupa untuk menambahkan juga file jquery pada folder js bootstrap atau letakkan sesuai keinginan anda asalkan direktori saat menghubungkan file css dan js nanti sesuai.



Perhatikan letak direktori file css dan js bootstrap dan jquery pada saat kita menghubungkannya. Kita juga tidak perlu menghubungkan seluruh isi bootstrap, yang hanya perlu di hubungkan adalah file bootstrap.css dan bootstrap.js

Membuat Halaman Pertama dengan Boostrap

setelah kita mendownload framework css bootstrap kemudian ekstrak pada project, lalu kita membuat sebuah file dengan nama index.html, kemudian ketikan baris kode berikut:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="css/bootstrap.min.css">
8   <script src="js/jquery.min.js"></script>
9   <script src="js/bootstrap.min.js"></script>
10 </head>
11 <body>
12
13 <div class="jumbotron text-center">
14   <h1>My First Bootstrap Page</h1>
15   <p>Resize this responsive page to see the effect!</p>
16 </div>
17
18 </body>
19 </html>
```

Hasilnya sebagai berikut:

My First Bootstrap Page

Resize this responsive page to see the effect!

Sebenarnya yang wajib dalam Bootstrap adalah yang dua ini:

```
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="css/bootstrap.min.css">
```

Tag `<meta name="viewport">` berfungsi untuk menyesuaikan tampilan web terhadap layar pengguna. Jadi jika dibuka di layar yang lebih kecil, ia akan melakukan sekala sesuai ukuran lebar layarnya. Tag ini wajib, bila kita ingin membuat web yang responsif. Selanjutnya tag `<link rel="...">` berfungsi untuk menyisipkan file CSS bootstrap ke dalam dokumen HTML.

Containers Pada Bootstrap 4

Containers adalah element layout yang paling dasar pada bootstrap. kita wajib menggunakan containers jika ingin menggunakan sistem grid bootstrap. Container bisa kita anggap sebagai wadah yang menampung semua element grid sistem atau element lain nya dari bootstrap.



Gambar 69. Container pada bootstrap

Sekarang kita akan membuat contoh untuk penggunaan container, ketik coding berikut:

```
● ● ●

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="css/bootstrap.min.css">
8   <script src="js/jquery.min.js"></script>
9   <script src="js/bootstrap.min.js"></script>
10 </head>
11 <body>
12
13 <div class="container">
14   <h1>My First Bootstrap Page</h1>
15   <p>This part is inside a .container class.</p>
16   <p>The .container class provides a responsive fixed width container.</p>
17 </div>
18
19 </body>
20 </html>
```

My First Bootstrap Page

This part is inside a .container class.

The .container class provides a responsive fixed width container.

Gambar 70. Contoh Container dan hasilnya pada bootstrap

Container-fluid

Jika container berukuran tidak penuh, dan letaknya di tengah, maka container-fluid adalah kebalikannya, container-fluid memiliki lebar penuh.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="css/bootstrap.min.css">
8   <script src="js/jquery.min.js"></script>
9   <script src="js/bootstrap.min.js"></script>
10 </head>
11 <body>
12
13 <div class="container-fluid">
14   <h1>My First Bootstrap Page</h1>
15   <p>This part is inside a .container-fluid class.</p>
16   <p>The .container-fluid class provides a full width container,
17     spanning the entire width of the viewport.</p>
18 </div>
19
20 </body>
21 </html>
```

My First Bootstrap Page

This part is inside a .container-fluid class.

The .container-fluid class provides a full width container, spanning the entire width of the viewport.

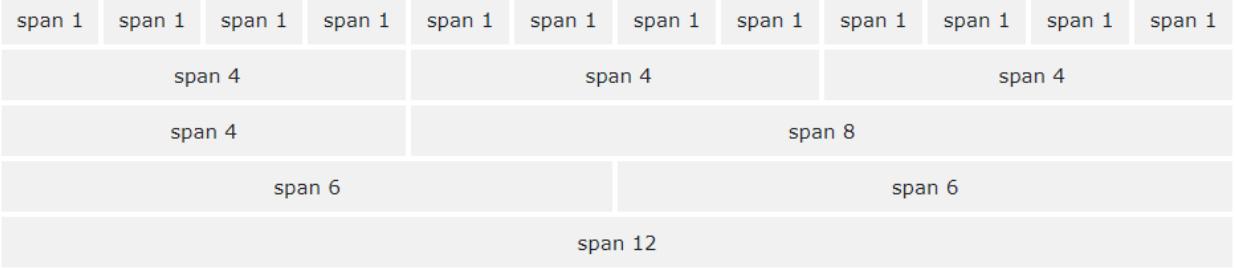
Gambar 71. Contoh Container fluid dan hasilnya pada bootstrap

Jadi perbedaan diantara kedua container tersebut adalah, *.container class provides a responsive fixed width container, .container-fluid class provides a full width container, spanning the entire width of the viewport.*

Grid System Bootstrap 4

Grid system adalah sistem pengaturan lebar untuk masing-masing komponen yang ingin kita buat. pengaturan ini bertujuan untuk mengatur ke-responsive-an website yang kita buat. Jadi kita bisa

mengaturnya sesuai yang kita inginkan dengan sangat mudah. tanpa harus menulis coding css secara manual untuk membuat halaman yang responsive.



	Extra small ≤576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
Max container width	None (auto)	540px	720px	960px	1140px
Class prefix	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
# of columns	12				
Gutter width	30px (15px on each side of a column)				
Nestable	Yes				
Column ordering	Yes				

Gambar 72. Grid System di Boostrap

Ada 12 grid pada bootstrap. yang bisa kita analogikan sebagai 12 kotak yang bisa kita atur lebarnya dengan mudah. Ada beberapa jenis class dasar dari bootstrap 4 untuk mengatur grid, di antaranya adalah:

- .col-*
digunakan untuk mengatur grid secara otomatis
- .col-sm-*
digunakan untuk mengatur grid pada ukuran layar monitor yang berukuran kecil (≥576px)
- .col-md-*
digunakan untuk mengatur grid pada ukuran layar monitor yang berukuran sedang (≥720px)
- .col-lg-*
digunakan untuk mengatur grid pada ukuran layar monitor yang berukuran besar (≥992px)
- .col-xl-*
digunakan untuk mengatur grid pada ukuran layar monitor yang berukuran ekstra besar (≥1200px)

Pada urutan pertama ada class col-* , col-* digunakan untuk mengatur grid secara default. misalnya jika di akses dari layar yang berukuran kecil, besar ataupun ekstra besar, ukuran gridnya akan tetap dan tidak berubah. jadi mau dibuka dari ukuran layar berapa pun, hasilnya akan tetap mengikuti pengaturan grid col-* yang sudah ditetapkan.

col-sm-* digunakan untuk mengatur grid jika halaman website diakses dari ukuran layar yang kecil seperti smartphone, maka pengaturan class col-sm-* lah yang akan digunakan. Untuk mempelajari lebih lanjut mengenai grid silahkan kunjungi halaman berikut: <https://getbootstrap.com/docs/4.4/layout/grid/>



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="css/bootstrap.min.css">
8   <script src="js/jquery.min.js"></script>
9   <script src="js/bootstrap.min.js"></script>
10 </head>
11 <body>
12
13 <div class="container-fluid">
14   <h1>Responsive Columns</h1>
15   <p>Resize the browser window to see the effect.</p>
16   <p>The columns will automatically stack on top of each other when the screen is less than 576px wide.</p>
17   <div class="row">
18     <div class="col-sm-3" style="background-color:lavender;".col-sm-3</div>
19     <div class="col-sm-3" style="background-color:lavenderblush;".col-sm-3</div>
20     <div class="col-sm-3" style="background-color:lavender;".col-sm-3</div>
21     <div class="col-sm-3" style="background-color:lavenderblush;".col-sm-3</div>
22   </div>
23 </div>
24 </body>
25 </html>
```

Responsive Columns

Resize the browser window to see the effect.

The columns will automatically stack on top of each other when the screen is less than 576px wide.

.col-sm-3

.col-sm-3

.col-sm-3

.col-sm-3

Gambar 73. Contoh grid dengan col-sm-3 dan hasilnya

Heading Bootstrap 4

Seperti yang sudah kita ketahui, heading merupakan element website untuk membuat judul pada website. heading terdiri dari h1 sampai h6. Bootstrap sudah menyediakan class untuk membuat heading, tanpa menggunakan tag heading (<h1> ... <h6>). Yaitu kita bisa menggunakan class h1, h2, h3, h4, h5 dan h6.

```
1 <p class="h1">Paragraf menggunakan class .h1</p>
```

Membuat Kustom Heading

Kita juga bisa membuat kustomisasi dengan menambahkan element lain, agar tampilan heading website menjadi lebih menarik. Jadi pada saat kita membuat halaman website kita bisa sekaligus membuat deskripsi halaman website tersebut. perhatikan contoh berikut ini.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="css/bootstrap.min.css">
8   <script src="js/jquery.min.js"></script>
9   <script src="js/bootstrap.min.js"></script>
10 </head>
11 <body>
12   <div class="container">
13     <h3>
14       Tutorial Bootstrap 4
15       <small class="text-muted">Politeknik Negeri Batam </small>
16     </h3>
17   </div>
18 </body>
19 </html>
```

Tutorial Bootstrap 4 Politeknik Negeri Batam

Gambar 74. Contoh heading custom dan hasilnya

Pada contoh di atas kita membuat heading 3 dengan text “Tutorial Bootstrap 4”. dan di dalamnya kita buat semacam tulisan deskripsinya. dengan menggunakan **class text-muted**. **class text-muted** adalah class untuk membuat text berwarna abu-abu.

Text Bootstrap 4

Ada beberapa komponen yang akan di bahas pada Text Bootstrap 4 ini, di antaranya adalah tulisan dalam bentuk paragraf, kemudian tulisan dalam bentuk quote. Berikut contoh:



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="css/bootstrap.min.css">
8   <script src="js/jquery.min.js"></script>
9   <script src="js/bootstrap.min.js"></script>
10 </head>
11 <body>
12   <div class="container">
13     <p>Ini contoh text yang <mark>ditebalkan</mark>.</p>
14     <p><del>Ini contoh text yang di coret.</del></p>
15     <p><u>Ini contoh text yang digaris bawahi</u></p>
16     <p><small>Ini contoh text berukuran kecil.</small></p>
17     <p><strong>Ini contoh text tebal.</strong></p>
18     <p><em>Ini contoh text miring.</em></p>
19   </div>
20 </body>
21 </html>
```

Ini contoh text yang **ditebalkan**.

Ini contoh text yang **di coret**.

Ini contoh text yang digaris bawahi

Ini contoh text berukuran kecil.

Ini contoh text tebal.

Ini contoh text miring.

Gambar 75. Contoh paragraf dan hasilnya

Sebenarnya untuk membuat paragraf seperti ini kita tidak harus menggunakan bootstrap. tapi di sini kita hanya ingin melihat bahwa jika menggunakan bootstrap, tampilan tulisan nya menjadi lebih bersih, karena secara default bootstrap sudah menggunakan jenis font yang berbeda (bukan Arial seperti default nya). Tampilan text yang terhubung dengan bootstrap lebih rapi dibanding tampilan text yang halamnnya tidak terhubung dengan bootstrap.

Text Untuk Quote

Bootstrap 4 juga sudah menyediakan class-class untuk membuat tampilan text yang khusus seperti kutipan quote. perhatikan contoh berikut.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="css/bootstrap.min.css">
8   <script src="js/jquery.min.js"></script>
9   <script src="js/bootstrap.min.js"></script>
10 </head>
11 <body>
12 <div class="container">
13 <blockquote class="blockquote">
14   <p>Jangan menjelaskan tentang dirimu kepada siapapun,
15     karena yang menyukaimu tidak butuh itu. Dan yang membencimu
16     tidak percaya itu.</p>
17   <footer class="blockquote-footer">Ali Bin Abi Thalib</footer>
18 </blockquote>
19 </div>
20 </body>
21 </html>
```

Jangan menjelaskan tentang dirimu kepada siapapun,
karena yang menyukaimu tidak butuh itu. Dan yang
membencimu tidak percaya itu.

— Ali Bin Abi Thalib

Gambar 76. Contoh Quote dan hasilnya

Untuk membuat tulisan quote, kita bisa menggunakan class **blockquote**. dan untuk tulisan keterangan nama pemilik quote nya kita bisa gunakan class **blockquote-footer**. seperti pada contoh di atas.

Menampilkan Gambar Dengan Bootstrap 4

Sebuah website atau aplikasi pasti memiliki gambar yang ditampilkan. tentu gambar tersebut harus diberi sentuhan CSS dulu agar responsive, memiliki bentuk yang sesuai dengan keinginan, atau bahkan mengatur ukuran dari gambar tersebut.

Membuat Gambar Menjadi Responsive

Untuk membuat gambar menjadi responsive, kita bisa menggunakan class .img-fluid dari bootstrap 4.

Letakkan class tersebut pada tag gambar yang ingin dibuat responsive. class img-fluid akan membuat gambar memiliki width 100%, dan height nya menjadi auto.

```
1 
```

Membuat Gambar Thumbnail

Gambar thumbnail di bootstrap berbentuk gambar yang memiliki bingkai putih di pinggirannya. untuk membuat gambar thumbnail dengan bootstrap, kita bisa menggunakan class img-thumbnail.

```
1 
```

Membuat Efek Melengkung Pada Gambar

Efek melengkung yang dimaksud adalah memberi efek lengkungan pada sudut gambar. untuk membuat efek lengkungan ini kita bisa menggunakan class rounded.

```
1 
```

Mengatur Posisi Gambar

Bootstrap juga memungkinkan kita untuk mengatur posisi suatu element atau komponen, misal menempatkan suatu element ke sebelah kiri. atau sebelah kanan. Untuk menempatkan element ke sebelah kiri, kita bisa menggunakan class float-left. sedangkan untuk menampilkan posisi element ke sebelah kanan, kita bisa gunakan class float-right.

```
1 
2 
```



float-left



float-right

Membuat Table Dengan Bootstrap 4

Dengan menggunakan bootstrap, kita tidak hanya bisa membuat satu model desain table. tapi kita bisa membuat banyak model desain sesuai keinginan. Table merupakan salah satu elemen paling penting pada sebuah aplikasi. karena biasanya data yang ada pada aplikasi atau sistem informasi ditampilkan dalam sebuah table. Sehingga desain table yang menarik dan rapi akan membuat data-data yang ditampilkan pada table menjadi lebih enak dilihat. karena terstruktur dengan rapi dan tampil menggunakan warna yang bersih.

Untuk membuat table menggunakan bootstrap 4, kita bisa menggunakan class .table pada tag <table>. Berikut ini adalah contoh syntax membuat table sederhana dengan HTML.



```
1 <div class="container">
2   <h2>Basic Table</h2>
3   <p>The .table class adds basic styling (light padding and horizontal dividers)
4   to a table:</p>
5   <table class="table">
6     <thead>
7       <tr>
8         <th>Firstname</th>
9         <th>Lastname</th>
10        <th>Email</th>
11      </tr>
12    </thead>
13    <tbody>
14      <tr>
15        <td>John</td>
16        <td>Doe</td>
17        <td>john@example.com</td>
18      </tr>
19    </tbody>
20  </table>
21 </div>
```

Basic Table

The .table class adds basic styling (light padding and horizontal dividers) to a table:

Firstname	Lastname	Email
John	Doe	john@example.com

Gambar 77. Contoh Table dan hasilnya

Style table pada bootstrap memiliki beberapa jenis style yang dapat digunakan untuk berbagai keperluan, berikut beberapa style table:

Class	Description
.table-primary	Blue: Indicates an important action
.table-success	Green: Indicates a successful or positive action
.table-danger	Red: Indicates a dangerous or potentially negative action
.table-info	Light blue: Indicates a neutral informative change or action
.table-warning	Orange: Indicates a warning that might need attention
.table-active	Grey: Applies the hover color to the table row or table cell
.table-secondary	Grey: Indicates a slightly less important action
.table-light	Light grey table or table row background
.table-dark	Dark grey table or table row background

Membuat Table Responsive

Untuk membuat tabel menjadi responsive maka kita perlu menambahkan class .table-responsive. kita tinggal membuat sebuah elemen `<div> .. </div>` di luar elemen table. dan dalam tag `<div>` tersebut tulis class .table-responsive.

```
1 <div class="table-responsive">
2 tabel diletakan disini
3 </div>
```

Maka ketika tabel di akses melalui layar berukuran kecil daripada ukuran panjang tabel maka akan menampilkan button scroll untuk ke kiri dan kanan.

Basic Table

The .table class adds basic styling (light padding and horizontal dividers) to a table:

Firstname	Lastname	Email
John	Doe	john@example.com

Komponen Lainnya pada Bootstrap 4

Bootstrap 4 masih memiliki banyak komponen lainnya, beberapa contoh diatas hanya beberapa dari komponen yang sering digunakan pada bootstrap. Komponen bootstrap yang lain dapat anda lihat pada link berikut: <https://getbootstrap.com/docs/4.4/components/>. Diantara beberapa komponen tersebut adalah:

1. Komponen Dasar

Berikut adalah komponen-komponenn dasar yang tersedia di dalam framework bootstrap :

- Glyphicon
- Dropdown
- Button Group
- Button Dropdown
- Input Group
- Navs
- Navbar
- Breadcrumbs
- Pagination
- Label
- Badge
- Jumbotron
- Page Header
- Thumbnails
- Alerts
- Progress Bar
- Panels
- Responsive Embed

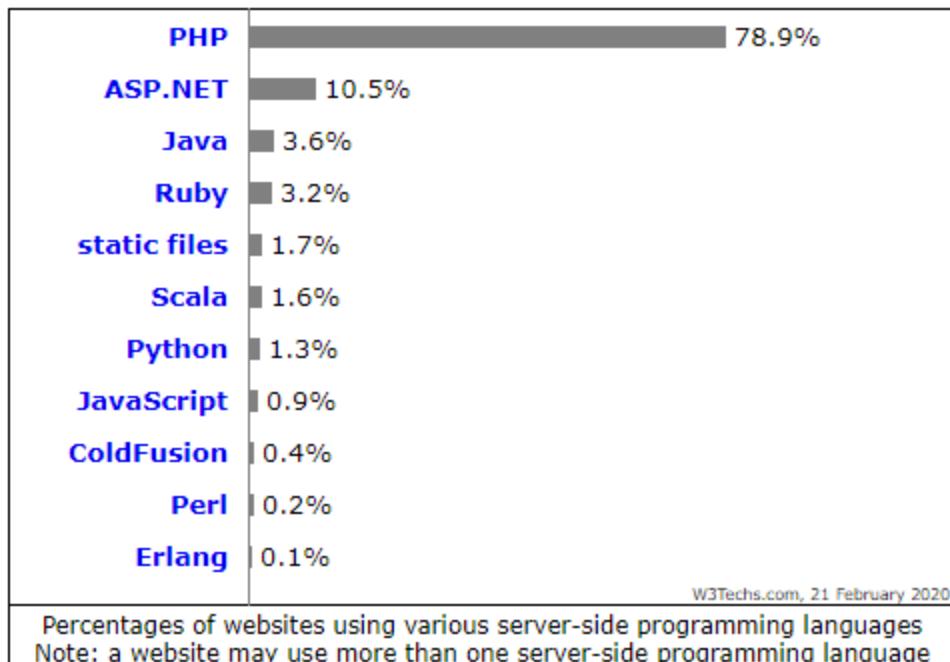
2. Komponen Javascript

Selain komponen CSS dan komponen dasar, Bootstrap juga menyediakan banyak pilihan plugin jQuery yang dapat memberikan efek animasi pada komponen Bootstrap. Hal ini akan membuat website menjadi lebih hidup. Komponen javascript pada Bootstrap sebagai berikut:

- Transitions
- Modal
- Dropdown
- Scrollspy
- Tab
- Tooltip
- Popover
- Alert
- Button
- Collapse
- Carousel
- Affix

IX PHP: HYPERTEXT PREPROCESSOR

PHP merupakan bahasa wajib programmer web. Sampai saat ini PHP masih menjadi standar bahasa pemrograman server side untuk pembuatan website. Berdasarkan data w3techs.com pada Februari 2020, PHP dipakai oleh lebih dari 78% web server di seluruh dunia.



Gambar 78. Jumlah penggunaan PHP

Pengertian PHP

PHP adalah bahasa pemrograman script server-side yang didesain untuk pengembangan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum (wikipedia). PHP di kembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh The PHP Group. Situs resmi PHP beralamat di <http://www.php.net>.

PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client). Saat ini PHP adalah singkatan dari *PHP: Hypertext Preprocessor*, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: *PHP: Hypertext Preprocessor*.

PHP dapat digunakan dengan gratis (free) dan bersifat Open Source. PHP dirilis dalam lisensi PHP License, sedikit berbeda dengan lisensi GNU General Public License (GPL) yang biasa digunakan untuk proyek Open Source.

Untuk membuat halaman web, sebenarnya PHP bukanlah bahasa pemrograman yang wajib digunakan. Kita bisa saja membuat website hanya menggunakan HTML saja. Web yang dihasilkan dengan HTML (dan CSS) ini dikenal dengan website statis, dimana konten dan halaman web bersifat tetap.

Sebagai perbandingan, website dinamis yang bisa dibuat menggunakan PHP adalah situs web yang bisa menyesuaikan tampilan konten tergantung situasi. Website dinamis juga bisa menyimpan data ke dalam database, membuat halaman yang berubah-ubah sesuai input dari user, memproses form, dll. Untuk pembuatan web, kode PHP biasanya di sisipkan ke dalam dokumen HTML. Karena fitur inilah PHP disebut juga sebagai Scripting Language atau bahasa pemrograman script.

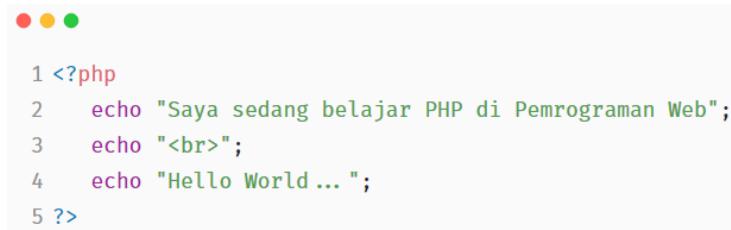
Banyak hal yang bisa kita lakukan dengan PHP, seperti menginput data ke database, menghasilkan gambar, menkonversi halaman text menjadi PDF, management cookie dan session, dll

Agar dapat mengakses halaman PHP dari web browser, maka kita harus menempatkan file **PHP** dalam folder khusus yang merupakan folder **home** dari **web server**. Pada aplikasi **XAMPP**, folder tersebut adalah

folder **htdocs** yang berada di **C:\xampp\htdocs**. Seluruh file PHP harus diletakkan dalam folder **htdocs** ini.

Menulis File PHP

File PHP pada dasarnya merupakan file text biasa yang bisa dibuat menggunakan aplikasi notepad bawaan windows, namun untuk penggunaan yang lebih baik, kita bisa pakai aplikasi text editor khusus programming seperti Notepad++, Atom, VS Code atau Sublime Text. Berikut contoh baris program php:



```
1 <?php
2 echo "Saya sedang belajar PHP di Pemrograman Web";
3 echo "<br>";
4 echo "Hello World ... ";
5 ?>
```

Gambar 79. Contoh baris program PHP

Lalu save sebagai **coba.php**. File tersebut akan berada pada alamat **C:\xampp\htdocs\belajar\coba.php**. Untuk menjalankannya, harus menggunakan web browser. Jika **Web Server Apache** belum berjalan, silahkan jalankan terlebih dahulu. Lalu buka browser, dan ketik alamat berikut: **localhost/belajar/coba.php** pada address bar dan tekan **Enter**.

Kesimpulan:

1. File php yang akan dijalankan harus berada di dalam folder **C:\xampp\htdocs**, baik itu di dalam folder tersebut, atau folder-folder dibawahnya. Untuk menjalankan di dalam browser, tinggal mengganti alamat **C:\xampp\htdocs** menjadi localhost. Jika file php berada di luar folder **htdocs**, maka web server tidak bisa mengaksesnya.
2. Setiap file PHP harus ditulis menggunakan akhiran **.php** (meskipun ini bisa diubah pada pengaturan modul php).
3. Penamaan file PHP harus tanpa spasi dan sebaiknya menggunakan huruf kecil. Anda dapat menggunakan underscore (**_**) sebagai pengganti spasi.
4. PHP yang ditulis di dalam HTML, filenya harus disimpan dengan ekstensi **.php** bukan **.html** meskipun isinya HTML dan PHP.

Cara Kerja Web Server Menjalankan Kode PHP

Sebuah file PHP selain berisi kode PHP itu sendiri, juga dapat berisi kode HTML, JavaScript dan CSS yang semuanya berada dalam satu file PHP. Web server Apache akan memisahkan bagaimana cara menjalankan kode script tersebut, sehingga butuh suatu cara untuk memberitahu web server bahwa kode “ini” adalah PHP, dan kode “itu” adalah HTML. Contoh:

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar PHP</title>
5 </head>
6 <body>
7   <p>Kalimat ini dibuat menggunakan HTML saja </p>
8   <?php
9     echo "<p>Kalimat ini dibuat menggunakan PHP </p>";
10  ?>
11 </body>
12 </html>

```

Walaupun kode di atas terdiri dari HTML dan PHP, agar web server “mau” mengeksekusi kode PHP yang terdapat di dalamnya, file tersebut harus di save sebagai file PHP, yakni dengan akhiran file *.php. Berikut cara kerja web server ketika menangani kode program di atas:

1. Web Server akan melihat extensi (jenis) file yang diakses. Jika file yang dijalankan adalah cara_kerja.html, maka web server tidak akan memanggil modul PHP untuk menjalankan kode PHP, karena web server menganggap itu adalah file HTML biasa dan menampilkannya langsung tanpa diproses. Namun jika file tersebut adalah cara_kerja.php, maka web server akan menjalankan modul PHP dan mengeksekusi kode PHP yang terdapat dalam file tersebut.
2. Web Server Apache akan memulai memproses file cara_kerja.php dari baris pertama sampai baris akhir secara berurutan.
3. Pada baris 1, web server mendapatkan tag pembuka HTML, yakni <!DOCTYPE html> kemudian diikuti dengan tag-tag HTML lainnya. Karena tidak menemukan instruksi untuk masuk ke mode PHP, seluruh tag HTML ini tidak memerlukan proses dan langsung dikirim ke web browser.
4. Begitu web server menemukan tag <?php pada baris ke-7, maka tag tersebut menginstruksikan kepada web server bahwa kode selanjutnya terdiri dari bahasa PHP, sehingga apa pun yang kita tulis setelah tag ini akan diproses mengikuti aturan bahasa pemrograman PHP. Kita sebut saja proses ini sebagai PHP mode.
5. Mulai dari tag <?php adalah PHP mode. Dalam PHP mode inilah kita menulis kode pemrograman PHP. Untuk contoh di atas, saya membuat sebuah perintah sederhana PHP, yakni echo. Echo adalah perintah PHP untuk menampilkan text yang terdapat di antara kedua tanda kutip (“”). Dalam contoh di atas echo dipakai untuk menampilkan teks: “<p>Kalimat ini dibuat menggunakan PHP</p>” ke dalam web browser.
6. Web server akan terus berada di PHP mode sampai menemukan tag penutup PHP, yakni tag ?> yang menandakan akhir dari PHP.
7. Tag ?> selain tanda keluar dari mode PHP, juga menginstruksikan kepada web server bahwa text berikutnya akan diproses sebagai text HTML biasa.

Namun kecepatan eksekusi PHP ini tidak akan dapat terasa terutama untuk file PHP sederhana seperti di atas. PHP sudah dioptimasi sedemikian rupa agar bisa di eksekusi dengan cepat layaknya file HTML biasa.

Sintaks Dasar PHP

Sintak adalah aturan penulisan kode program. Pada dasarnya setiap bahasa pemrograman itu sama, yang membedakan adalah sintak dan fitur. Dibawah ini adalah bentuk paling dasar program PHP:

```

1 <?php
2 echo "Saya sedang belajar PHP di Pemrograman Web";
3 echo "<br>";
4 echo "Hello World ... ";
5 ?>

```

Keterangan:

- <?php ini adalah pembuka program PHP. Pembuka ini wajib ada di setiap program PHP.
- echo "Hello World!"; adalah sebuah statement atau perintah untuk menampilkan teks.

Penulisan Statement dan Ekspresi

Statement dan ekspresi adalah intruksi yang akan diberikan ke komputer. Setiap statement dan ekspresi di PHP harus diakhiri dengan titik koma (;). Apabila kita lupa mengetikan titik koma, maka program akan mengalami error.

```
1 <?php
2 echo "ini statement 1";
3 echo "ini statement 2";
4 $a = $b + $c;
```

Aturan Dasar Penulisan Kode PHP

PHP juga memiliki aturan penulisan seperti case sensitivity (perbedaan antara huruf besar dan kecil), cara mengakhiri sebuah baris perintah, serta pengaruh penggunaan spasi dalam membuat kode program PHP. Berikut adalah aturan dasar penulisan kode PHP:

1. Case Sensitivity (perbedaan huruf besar dan kecil) dalam PHP

PHP tidak membedakan huruf besar dan kecil (case insensitive) untuk penamaan fungsi (function), nama class, maupun keyword bawaan PHP seperti echo, while, dan class. Ketiga baris berikut akan dianggap sama dalam PHP:

```
1 <?php
2 Echo "Hello World"; ECHO "Hello World"; EcHo "Hello World";
3 ?>
```

Akan tetapi, PHP membedakan huruf besar dan huruf kecil (case sensitive) untuk penamaan variabel, sehingga \$nama, \$Nama dan \$NAMA akan dianggap sebagai 3 variabel yang berbeda. Sering kali error terjadi karena salah menuliskan nama variabel, yang seharusnya menggunakan huruf kecil ditulis dengan huruf besar.

```
1 <?php
2 $andi="Andi"; echo $Andi; // Notice: Undefined variable:
   Andi
3 ?>
```

Untuk mengatasi perbedaan ini, disarankan menggunakan huruf kecil untuk seluruh kode PHP, termasuk variabel, fungsi maupun class. Jika perlu membuat nama variabel yang terdiri dari 2 kata, karakter spasi bisa digantikan dengan underscore (_).

2. Penulisan Baris Perintah dalam PHP

Statement (baris perintah) di dalam PHP adalah kumpulan perintah PHP yang menginstruksikan PHP untuk melakukan sesuatu. Baris perintah ini bisa terdiri dari satu baris singkat (seperti perintah echo untuk menampilkan text di layar) atau bisa sesuatu yang lebih rumit dan terdiri dari beberapa baris, seperti kondisi if, atau kode perulangan (loop). Berikut adalah contoh beberapa baris perintah dalam PHP:

```
1 <?php
2 echo "Hello, world";
3 sebuah_fungsi(21, "supardianto");
4 $a = 1;
5 $nama = "supardianto";
6 $b = $a / 25.0;
7 if ($y == $z) {
8     echo "Tampilkan Tabel";
9 }
10 ?>
```

Terlihat dari beberapa contoh baris perintah di atas, PHP menggunakan tanda semicolon (titik koma) „;“ sebagai tanda akhir baris perintah. Kumpulan baris perintah yang menggunakan tanda kurung kurawal seperti kodisi IF atau perulangan (loop) tidak butuh tanda titik koma setelah kurung penutup.

3. Karakter Spasi dan Tab dalam PHP

Secara umum, karakter spasi dan tab diabaikan ketika mengeksekusi kode program PHP. Kita dapat boleh memecah sebuah statement menjadi beberapa baris atau menyatukan beberapa statement dalam sebuah baris yang panjang.

Pengertian Variabel dan Cara Penulisan Variabel PHP

Dalam pemrograman, **variabel** adalah suatu lokasi penyimpanan (*di dalam memori komputer*) yang berisikan nilai atau informasi yang nilainya tidak diketahui maupun telah diketahui ([wikipedia](#)). Dalam definisi bebasnya, **variabel** adalah kode program yang digunakan untuk menampung nilai tertentu. Nilai yang disimpan di dalam variabel selanjutnya dapat dipindahkan ke dalam database, atau ditampilkan kembali ke pengguna.

Nilai dari variabel dapat diisi dengan informasi yang diinginkan dan dapat diubah nilainya pada saat kode program sedang berjalan. Sebuah variabel memiliki nama yang digunakan untuk mengakses nilai dari variabel itu. Sama seperti variabel dalam bahasa pemrograman lainnya, variabel dalam PHP digunakan untuk menampung nilai inputan dari user, atau nilai yang kita definisikan sendiri. Namun PHP memiliki beberapa aturan tentang cara penggunaan dan penulisan variabel.

Aturan Penulisan Variabel dalam PHP

1. Penulisan variabel harus diawali dengan tanda \$

Variabel di dalam PHP harus diawali dengan dollar sign atau tanda dollar (\$). Setelah tanda \$, sebuah variabel PHP harus diikuti dengan karakter pertama berupa huruf atau underscore (_), kemudian untuk karakter kedua dan seterusnya bisa menggunakan huruf, angka atau underscore (_).

Dengan aturan tersebut, variabel di dalam PHP tidak bisa diawali dengan angka. Minimal panjang variabel adalah 1 karakter setelah tanda \$. Berikut adalah contoh penulisan variabel yang benar dalam PHP:

```
1 <?php
2   $i;
3   $nama;
4   $_lokasi_memori;
5   $ANGKA_MAKSIMUM;
6 ?>
```

Dan Berikut contoh yang salah:

```
1 <?php
2   $4ever; //variabel tidak boleh diawali dengan angka
3   $_salah_satu; //variabel tidak boleh mengandung spasi
4   $nama*^; //variabel tidak boleh mengandung karakter
  khusus: * dan ^
5 ?>
```

2. Variabel dalam PHP bersifat case sensitif

PHP membedakan variabel yang ditulis dengan huruf besar dan kecil (bersifat case sensitif) , sehingga \$belajar tidak sama dengan \$Belajar dan \$BELAJAR, ketiganya akan dianggap sebagai variabel yang berbeda. Untuk menghindari kesalahan program yang dikarenakan salah merujuk variabel, disarankan menggunakan huruf kecil untuk seluruh nama variabel.

3. Cara Memberikan Nilai kepada Variabel

Sama seperti sebagian besar bahasa pemrograman lainnya, untuk memberikan nilai kepada sebuah variabel, PHP menggunakan tanda sama dengan (=). Operator 'sama dengan' ini dikenal dengan istilah Assignment Operators.

Perintah pemberian nilai kepada sebuah variabel disebut dengan assignment. Jika variabel tersebut belum pernah digunakan, dan langsung diberikan nilai awal, maka disebut juga dengan proses inisialisasi. Berikut contoh cara memberikan nilai awal (inisialisasi) kepada variabel:

```
1 <?php
2   $nama = "andi";
3   $umur = 17;
4   $pesan = "Saya sedang belajar PHP di Prog Web";
5 ?>
```

4. Variabel dalam PHP tidak memerlukan deklarasi terlebih dahulu

Jika anda pernah mempelajari bahasa pemrograman desktop seperti Pascal, C, C++, dan Visual Basic, di dalam bahasa pemrograman tersebut, sebuah variabel harus dideklarasikan terlebih dahulu sebelum digunakan. Namun di dalam PHP, variabel tidak perlu dideklarasikan terlebih dahulu.

5. Variabel dalam PHP tidak bertipe

Dalam kelompok bahasa pemrograman, PHP termasuk *Loosely Type Language*, yaitu jenis bahasa pemrograman yang variabelnya tidak terikat pada sebuah tipe tertentu. Hal ini berbeda jika dibandingkan dengan bahasa pemrograman desktop seperti Pascal atau C, dimana jika anda membuat sebuah variabel bertipe *integer*, maka variabel itu hanya bisa menampung nilai angka, dan anda tidak akan bisa mengisinya dengan huruf.

Di dalam PHP, setiap variabel bebas diisi dengan nilai apa saja, seperti contoh berikut:

```
1 <?php
2   $a = 17; // nilai variabel a berisi angka (integer)
3   $a = "aku"; // nilai variabel a diubah menjadi kata (string)
4   $a = 17.42; // nilai variabel a diubah menjadi desimal (float)
5 ?>
```

Menampilkan Nilai Variabel

Untuk menampilkan nilai atau isi dari variabel, kita tinggal menampilkannya dengan perintah echo atau print, seperti berikut ini:

```
1 <?php
2   $a='Saya Sedang belajar PHP';
3   $b=5;
4
5   print $a;
6   echo $b;
7 ?>
```

Variabel dapat dikatakan sebagai inti dari sebuah bahasa pemrograman. Karena melalui variabel inilah kita memanipulasi data inputan agar menjadi nilai yang diinginkan. Selain variabel, PHP juga menyediakan sebuah solusi lain untuk menampung nilai data dengan konstanta (constant).

Tipe data dalam PHP

Sebuah variabel atau konstanta merupakan ‘tempat’ dari data. Di dalam bahasa pemrograman (dan juga PHP), data yang diinput kedalam variabel atau konstanta akan memiliki tipe tertentu. Tipe-tipe ini nantinya menentukan bagaimana cara kita memprosesnya. Beberapa tipe data terdengar familiar, seperti tipe data angka, desimal dan text. Namun kita juga akan menemukan tipe data lain seperti boolean dan array.

Tipe Data Integer dalam PHP

Tipe data integer adalah tipe data yang berupa angka bulat seperti: 1, 22, dan -172. Tipe data integer umum digunakan untuk data dengan angka bulat, seperti harga barang, jumlah stock dan jumlah mahasiswa. Jika data yang kita miliki kemungkinan akan mengandung pecahan, maka tipe data yang digunakan adalah float.

```
1 <?php
2   $a=14; $b=16;
3   $c= $a + $b;
4   echo $c; // 30
5 ?>
```

Karena PHP tidak memerlukan pendeklarasian variabel, maka ketika sebuah variabel berisi angka bulat, maka secara otomatis variabel tersebut di sebut sebagai variabel integer.

Tipe Data Float dalam PHP

Tipe data float (disebut juga tipe data floating point, atau real number) adalah tipe data angka yang memiliki bagian desimal di akhir angka, atau memiliki floating point (floating point adalah istilah dalam bahasa Inggris untuk menyebut tanda “titik” yang menandakan bilangan desimal). Contoh angka float adalah seperti: 0,9 atau 3,14.

Tipe data float cocok digunakan untuk variabel yang akan berisi angka pecahan, seperti nilai IPK, hasil pembagian, atau hasil komputasi numerik yang angkanya tidak bisa ditampung oleh data integer.

```
1 <?php  
2   $a=10.66;  
3   $b=12.4;  
4   $c= $a + $b;  
5   echo $c; // 23.06  
6 ?>
```

Sama seperti tipe data integer, variabel dengan tipe data float juga dapat melakukan operasi numerik seperti penambahan, pembagian, perkalian, dan lain-lain. Didalam PHP penulisan nilai desimal ditandai dengan tanda “titik”, bukan “koma” seperti yang biasa kita gunakan sehari-hari. Nilai 0,87 harus ditulis menjadi 0.87.

Tipe Data String dalam PHP

Tipe data string adalah tipe data yang berisi text, kalimat, atau kumpulan karakter. Sebagai contoh, “a”, “saya sedang belajar PHP” atau “tUT0r1al pHp?!” semuanya adalah string. Tipe data string mungkin adalah tipe data yang paling sering digunakan, dan memiliki banyak fitur yang disediakan PHP. Karakter yang didukung saat ini adalah 256 karakter ASCII. List karakter ASCII tersebut dapat dilihat di <http://www.ascii-code.com>.

PHP menyediakan 4 cara penulisan tipe data string, yakni Single Quoted, Double Quoted, Heredoc, dan Nowdoc.

Penulisan Tipe Data String dengan Single Quoted

Penulisan tipe data string menggunakan single quoted atau tanda petik satu (karakter ‘) merupakan cara penulisan string yang paling sederhana. Kita tinggal membuat sebuah kata atau kalimat, dan menambahkan tanda petik satu di awal dan akhir kalimat.

Penulisan Tipe Data String dengan Double Quoted

Cara kedua dalam penulisan tipe data string dalam PHP adalah dengan menggunakan Double Quoted atau tanda petik dua (karakter “). Walaupun seperti tidak ada perbedaan dengan menggunakan single quote, hasil yang di dapat akan sangat berbeda.

Dengan double quoted, PHP akan memproses karakter-karakter khusus seperti carriage return (\n), dan karakter tab (\t) dan juga memproses setiap variabel (yang ditandai dengan tanda \$ di depan kata).

Di karenakan metode double quoted melakukan pemrosesan terlebih dahulu, maka untuk menampilkan karakter khusus seperti tanda petik (karakter ‘), tanda dollar (karakter \$) dan tanda-tanda khusus lainnya, kita harus menggunakan backslash (karakter \).

```

1 <?php
2 $string1="Ini adalah string sederhana";
3
4 $string2="Ini adalah string
5 yang bisa memiliki beberapa
6 baris";
7
8 $string3="Dia berkata: \"I'll be back\"";
9
10 $string4="Anda telah berhasil menghapus C:\xampp\htdocs";
11
12 $string5="Kalimat ini akan akan pindah ke: \n baris baru";
13
14 $string6="Variabel akan otomatis ditampilkan: $string1 dan
    $string3";
15
16 echo $string1; echo "<br \>";
17 echo $string2; echo "<br \>";
18 echo $string3; echo "<br \>";
19 echo $string4; echo "<br \>";
20 echo $string5; echo "<br \>";
21 echo $string6;
22 ?>

```

Ini adalah string sederhana
 Ini adalah string yang bisa memiliki beberapa baris
 Dia berkata: "I'll be back"
 Anda telah berhasil menghapus C:\xampp\htdocs
 Kalimat ini akan akan pindah ke: baris baru
 Variabel akan otomatis ditampilkan: Ini adalah string sederhana dan Dia berkata: "I'll be back"

Pembuatan Fungsi PHP

Dalam merancang kode program, kadang kita sering membuat kode yang melakukan tugas yang sama secara berulang-ulang, seperti membaca tabel dari database, menampilkan penjumlahan, dan lain-lain. Tugas yang sama ini akan lebih efektif jika dipisahkan dari program utama, dan dirancang menjadi sebuah fungsi.

Fungsi (atau Function) dalam bahasa pemrograman adalah kode program yang dirancang untuk menyelesaikan sebuah tugas tertentu, dan merupakan bagian dari program utama. Kita dapat membuat fungsi sendiri, atau menggunakan fungsi yang dibuat oleh programmer lain.

Cara Menggunakan Fungsi PHP

Menggunakan fungsi dalam teori pemrograman sering juga disebut dengan istilah ‘memanggil fungsi’ (calling a function). Fungsi dipanggil dengan menulis nama dari fungsi tersebut, dan diikuti dengan argumen (jika ada). Argumen ditulis di dalam tanda kurung, dan jika jumlah argumen lebih dari satu, maka diantaranya dipisahkan oleh karakter koma.

Sebuah fungsi dalam memproses data, kadang memerlukan beberapa inputan atau nilai masukan. Inputan inilah yang dimaksud dengan **argumen**. **Parameter** adalah sebutan lain untuk **argumen**. Perbedaannya,

parameter merujuk kepada inputan fungsi pada saat pendefenisian fungsi tersebut, dan **argumen** adalah sebutan untuk inputan fungsi pada saat pemanggilan fungsi.

Contoh Pemanggilan Fungsi PHP

Berikut adalah format dasar pemanggilan, dan pengembalian nilai fungsi:

```
1 <?php  
2 $varibel_hasil_fungsi = nama_fungsi(argumen1, argumen2,  
    argumen3)  
3 ?>
```

Keterangan:

- **\$varibel_hasil_fungsi** adalah variabel yang akan menampung hasil pemrosesan fungsi. Tergantung fungsinya, hasil dari sebuah fungsi bisa berupa angka, string, array, bahkan objek.
- **nama_fungsi** adalah nama dari fungsi yang akan dipanggil
- **argumen1, argumen2** adalah nilai inputan fungsi. Banyaknya **argumen** yang dibutuhkan, tergantung kepada fungsi tersebut. Jika sebuah fungsi membutuhkan **argumen** 2 buah angka, maka kita harus menginputnya sesuai dengan aturan tersebut, atau jika tidak, PHP akan mengeluarkan error.

```
1 <?php  
2 $akar_kuadrat = sqrt(49);  
3 echo "Akar kuadrat dari 49 adalah $akar_kuadrat";  
4 // Akar kuadrat dari 49 adalah 7  
5 ?>
```

Sebagai contoh, PHP menyediakan fungsi akar kuadrat, yakni **sqrt()**. Dalam contoh, fungsi **sqrt()** akan menghitung akar kuadrat dari nilai argumen yang diinput.

Nilai hasil dari fungsi **sqrt(49)**,

selanjutnya ditampung dalam variabel **\$akar_kuadrat**, yang kemudian ditampilkan ke dalam web browser.

Cara Penulisan dan Pembuatan Fungsi PHP

Untuk membuat fungsi di dalam PHP, berikut adalah format dasar pembuatan fungsi:

```
1 <?php  
2 function nama_fungsi ($parameter1, $parameter2)  
3 {  
4     // kode program fungsi  
5     return $nilai_akhir  
6 }  
7 ?>
```

Keterangan:

- Kata **function** adalah instruksi kepada PHP bahwa kita akan membuat fungsi
- **nama_fungsi** adalah nama dari fungsi yang akan ditulis
- **\$parameter1, \$parameter2** adalah variabel perantara yang akan menyimpan inputan yang diperlukan dalam pemrosesan fungsi (argumen). Tergantung kebutuhan, anda bebas merancang seberapa banyak parameter yang dibutuhkan.
- **return** adalah perintah khusus untuk fungsi, dimana kata return menginstruksikan kepada PHP bahwa pemrosesan fungsi telah selesai. **return \$nilai_akhir** berarti bahwa fungsi akan ‘mengembalikan’ **\$nilai_akhir** sebagai hasil dari fungsi.

Contoh Pembuatan Fungsi PHP

Nama fungsi ini adalah **perkalian**, dan membutuhkan 2 argumen dengan tipe data angka (integer atau float), berikut adalah cara pembuatan dan penggunaan fungsi perkalian:

```
1 <?php
2 //pembuatan fungsi
3 function perkalian($angka1, $angka2)
4 {
5     $a= $angka1;
6     $b= $angka2;
7     $hasil= $a*$b;
8     return $hasil;
9 }
10
11 //pemanggilan fungsi
12 $hasil=perkalian(4,5);
13 echo "Perkalian 4 x 5 adalah $hasil";
14 echo "<br />";
15 echo "Perkalian 7 x 2 adalah ".perkalian(7,2);
16 ?>
```

Keterangan:

Pada baris ke-3, kita mendefenisikan fungsi **perkalian()** yang memerlukan 2 parameter.

Parameter ditulis sebagai *variabel*, dan kita bebas menentukan nama variabel ini, dalam contoh diatas, parameter untuk fungsi **perkalian()** adalah **\$angka1** dan **\$angka2**.

Di dalam fungsi **perkalian()**, saya membuat variabel **\$a** dan **\$b** yang digunakan untuk menampung nilai **\$angka1** dan **\$angka2**.

Variabel **\$hasil** digunakan untuk menampung nilai akhir dari perkalian **\$a*\$b**, dan hasilnya di-kembalikan

dengan perintah **return \$hasil**. **Return** secara otomatis menutup fungsi, dan jika anda masih memiliki kode program setelah perintah **return**, perintah tersebut tidak akan diproses, oleh karena itu perintah **return** harus diletakkan di akhir penulisan fungsi.

Fungsi Bawaan PHP

PHP memiliki ribuan fungsi-fungsi bawaan. Memahami cara kerja dan cara penggunaan fungsi bawaan PHP ini akan sangat membantu kita untuk membuat kode program yang efektif dan efisien.

Mengubah Huruf Besar ke Kecil atau sebaliknya

Untuk mengubah huruf besar ke kecil dalam sebuah string PHP, kita bisa menggunakan fungsi **strtolower()**. Fungsi ini membutuhkan 1 buah argumen, yakni string yang akan diubah.

```
1 <?php
2 $kalimat = "SAYA Sedang Belajar PHP";
3 $kalimat_new = strtolower($kalimat);
4 echo $kalimat_new;
5 // saya sedang belajar php
6 ?>
```

Untuk mengubah huruf besar ke kecil di dalam PHP, bisa menggunakan fungsi **strtoupper()**. Cara penggunaannya hampir sama dengan fungsi **strtolower()**.

```
1 <?php
2 $kalimat = "SAYA Sedang Belajar PHP";
3 $kalimat_new = strtoupper($kalimat);
4 echo $kalimat_new;
5 // saya sedang belajar php
6 ?>
```

Memotong atau Mengambil Sebagian String

Fungsi **substr()** adalah fungsi PHP untuk memotong string, atau untuk mengambil sebagian nilai dari sebuah string. Fitur ini cukup sering digunakan dalam proses pembuatan program PHP, terutama yang membutuhkan manipulasi string. Berikut contoh beberapa kasus kapan menggunakan **substr()**:

- kita memiliki sebuah string berbentuk tanggal: "14-09-2015". Bagaimana caranya untuk mengambil nilai bulan dari string tersebut, yakni karakter "09" ?
- katakan NIM seorang mahasiswa terdiri dari 8 digit: "12140001". Dua digit pertama adalah tahun masuk mahasiswa, dua digit berikutnya kode jurusan, dan empat digit terakhir adalah no urut mahasiswa. Bagaimana cara memisahkan digit-digit ini?

Fungsi **substr()** membutuhkan 2 buah argumen dan 1 argumen tambahan (opsional). Argumen pertama adalah string asal yang ingin diambil nilainya.

Argumen kedua berupa posisi awal pemotongan, dan argumen ketiga diisi jumlah karakter yang akan diambil. Argumen kedua dan ketiga bertipe integer dan bisa positif maupun negatif.

Mengambil Karakter Dari Awal String

Penggunaan pertama fungsi substr() adalah cara mengambil karakter yang dimulai dari awal string. Berikut contoh penggunaannya:

```
1 <?php
2 $kalimat = "123456789";
3 $sub_kalimat = substr($kalimat,3);
4 echo $sub_kalimat;
5 // 456789
6 ?>
```

Dalam kode, saya mengambil string **\$kalimat** mulai dari index ke-3. Jika fungsi substr() ditulis 2 argumen seperti ini, dan argumen kedua bernilai positif, maka fungsi substr() akan mengembalikan nilai string **\$kalimat** mulai dari angka ke-4, yakni angka "4" hingga akhir string.

Fungsi **substr()** juga memiliki argumen ketiga yang bersifat opsional (boleh diisi atau dikosongkan). Jika kita menambahkan argumen ketiga, nilai ini berfungsi sebagai penentu 'berapa banyak jumlah karakter yang akan diambil'. Berikut contohnya:

```
1 <?php
2 $kalimat = "Belajar PHP di Polibatam";
3 $sub_kalimat = substr($kalimat,8,3);
4 echo $sub_kalimat;
5 // PHP
6 ?>
```

Fungsi **substr(\$kalimat,8,3)** akan mengambil string **\$kalimat** mulai dari index ke-8 (karakter ke-9) dan ambil sebanyak 3 karakter.

Mengambil Karakter Dari Akhir String

Selain dari awal string, kita juga bisa mengambil karakter mulai dari akhir string. Caranya adalah dengan memberikan nilai negatif pada argumen kedua fungsi **substr()**. Berikut contoh penggunaannya:

```
1 <?php
2 $kalimat = "Belajar PHP di Polibatam";
3 $sub_kalimat = substr($kalimat,-10);
4 echo $sub_kalimat;
5 // Polibatam
6 ?>
```

Fungsi **substr(\$kalimat,-10)** berarti ambil 10 karakter terakhir dari string **\$kalimat**.

Konversi String Menjadi Array PHP (Fungsi explode)

Fungsi **explode()** adalah fungsi PHP yang digunakan untuk memecah sebuah string menjadi array, atau dengan kata lain mengkonversi string menjadi array. Dalam memecah sebuah string menjadi array, fungsi **explode()** membutuhkan beberapa argumen. Berikut adalah format dasar dari fungsi **explode** dalam PHP:

```
1 array explode (string $delimiter , string $string [, int $limit ] )
```

Penjelasan:

- Nilai kembalian fungsi **explode** berupa sebuah array.
- Argumen pertama (**\$delimiter**) diisi dengan string karakter yang digunakan sebagai ‘pemecah’ string menjadi array.
- Argumen kedua (**\$string**) diisi dengan string yang akan dikonversikan.
- Argumen ketiga yang bersifat opsional (**\$limit**) diisi dengan batasan jumlah array yang ingin dihasilkan. Argumen ketiga ini bertipe angka (integer) dan bisa berupa nilai positif atau negatif.

Sebagai contoh, dalam kode program berikut ini saya memecah string **\$kalimat** = “satu dua tiga empat lima” menjadi array:

```
1 <?php
2 $kalimat = "satu dua tiga empat lima";
3 $arr_kalimat = explode (" ",$kalimat);
4 var_dump ($arr_kalimat);
5 ?>
```

```
array(5) { [0]=> string(4) "satu" [1]=> string(3)
"dua" [2]=> string(4) "tiga" [3]=> string(5)
"empat" [4]=> string(4) "lima" }
```

Dalam string **\$kalimat** diatas, kita menggunakan ‘spasi’ sebagai pemisah kata sehingga untuk memecah string tersebut, spasi ini akan menjadi pembatas (delimeter) untuk fungsi **explode**.

Argumen ketiga fungsi **explode()** bersifat opsional. Argumen ini bisa diisi dengan angka (integer) positif maupun negatif. Jika diisi dengan angka positif, maka angka disini menunjukkan jumlah batasan element array yang akan dihasilkan.

```
1 <?php
2 $kalimat = "satu, dua, tiga, empat, lima";
3 $arr_kalimat = explode (", ",$kalimat, 3);
4 var_dump ($arr_kalimat);
5 ?>
```

```
array(3) { [0]=> string(4) "satu" [1]=> string(3)
"dua" [2]=> string(17) "tiga, empat, lima" }
```

Memformat Tampilan Angka dalam PHP (Fungsi number_format)

PHP menyediakan beberapa fungsi bawaan untuk memformat tampilan angka, yakni fungsi **number_format()**. Fungsi ini juga bisa digunakan untuk menformat angka agar sesuai dengan penulisan mata uang rupiah.

Fungsi **number_format()** adalah fungsi bawaan PHP yang bisa digunakan untuk memformat tampilan angka, baik itu angka integer maupun float. Dengan memformat tampilan angka, akan membuatnya lebih

'cantik' dan mudah dibaca. Selain itu fungsi ini juga akan membulatkan atau menambahkan angka "0" dibelakang koma jika dibutuhkan.

Fungsi `number_format()` memiliki 2 buah cara penulisan, yakni dengan 2 argumen atau 4 argumen. Untuk fungsi dengan 2 argumen, berikut adalah format dasar penulisannya:

```
1 string number_format ( float $number [, int $decimals = 0 ] )
```

Penjelasan:

- Hasil akhir fungsi ini bertipe string.
- Argumen pertama (**\$number**) membutuhkan input nilai angka yang akan diformat. Argumen ini bertipe float, tapi bisa juga bisa diisi dengan nilai integer.
- Argumen kedua (**\$decimals**) bersifat opsional. Argumen ini menentukan berapa jumlah angka desimal (angka di belakang koma) yang dibutuhkan. Apabila tidak diisi, dianggap sebagai 0.

Berikut contoh penggunaan fungsi `number_format()`:

```
1 <?php
2 $angka = 1999.12345;
3 $angka_format = number_format($angka);
4 echo $angka_format;
5 // 1,999
6 ?>
```

Fungsi `number_format()` akan "membuang" seluruh bagian desimal dari angka diatas. Ini terjadi karena secara default, fungsi ini menggunakan 0 sebagai jumlah digit desimal (angka di belakang koma).

Dengan menambahkan argumen kedua, kita bisa menentukan berapa digit desimal yang diperlukan:

```
1 <?php
2 $angka = 1999.888;
3 $angka_format = number_format($angka,2);
4 echo $angka_format;
5 // 1,999.89
6 ?>
```

fungsi `number_format()` akan membatasi 2 digit desimal. Selain itu bahwa fungsi ini juga membulatkan nilai desimalnya.

Argumen ketiga dan keempat fungsi `number_format()` digunakan untuk menentukan karakter apa sebagai pemisah nilai ribuan dan nilai desimal. Sebagai contoh, karena di Indonesia kita menggunakan karakter titik sebagai pemisah angka ribuan dan karakter koma sebagai pemisah desimal, kita bisa memformat angka tersebut dengan fungsi `number_format()`, seperti contoh berikut:

```
1 <?php
2 $angka = 3050145.756;
3 $angka_format = number_format($angka,2,",",".");
4 echo "Rp. ".$angka_format;
5 // Rp. 3.050.145,76
6 ?>
```

Dalam kode diatas kita menyambungkan awal "Rp. " dengan string hasil fungsi `number_format()`. Tampilan tersebut cocok digunakan untuk hasil akhir angka mata uang untuk membuat laporan yang banyak melibatkan nilai uang.

X ARRAY PADA PHP

Pengertian Array

Array (atau larik dalam bahasa indonesia) bukanlah tipe data dasar seperti integer atau boolean, Array adalah sebuah tipe data bentukan yang terdiri dari kumpulan tipe data lainnya. Menggunakan array akan memudahkan dalam membuat kelompok data, serta menghemat penulisan dan penggunaan variabel. Array adalah salah satu struktur data yang berisi sekumpulan data dan memiliki indeks. Indeks digunakan untuk mengakses nilai array. Indeks array selalu dimulai dari nol (0). Contoh:

"Hardisk 2TB"	"Flashdisk 32GB"	"Modem"
0	1	2

Jadi, apabila kita ingin menampilkan "Hardisk 2TB", maka kita harus mengambil indeks yang ke-0.

Array adalah variabel yang bisa diisi oleh banyak data sekaligus, dan didalam array bisa terdiri dari beberapa kumpulan tipe data. Apabila kita memiliki beberapa data, maka sebelumnya data tersebut harus disimpan pada masing-masing variabel, namun dengan penggunaan array data tersebut dapat kita simpan hanya dalam satu variabel yaitu array.

Sebagai contoh: Semisal dalam sebuah kelas terdapat 5 orang siswa, lalu kita ingin menyimpan 5 nama orang siswa tersebut dalam variabel, maka penulisannya adalah sebagai berikut :

```
1 <?php  
2 $siswa1 = "Anton";  
3 $siswa2 = "Budi";  
4 $siswa3 = "Cika";  
5 $siswa4 = "Danang";  
6 $siswa5 = "Erika";  
7 ?>
```

Dengan cara diatas, kita perlu membuat 5 variabel untuk menyimpan 5 nama siswa, jika kita menyimpan data 5 nama siswa tersebut dengan array, kita hanya perlu menuliskan 1 variabel saja.

```
1 <?php
2 $nama_siswa = array('Anton','Budi','Cika','Danang','Erika');
3 ?>
```

Penulisan Array Pada PHP

Kita bisa menuliskan array pada php menggunakan 2 cara yaitu :

1. menggunakan fungsi array()
2. menggunakan tanda kurung siku []

Perhatikan contoh dibawah ini :



```
1 <?php
2 $buah = array('Pisang','Semangka','Durian','Apel','Jeruk');
3 $sayur = ['Kangkung','Bayam','Buncis','Kubis','Wortel'];
4 ?>
```

Contoh penulisan Array

Keterangan :

- Pada Array \$buah kita gunakan fungsi array() dan menuliskan isi array di dalam fungsi array, dengan pemisah tanda koma.
- Pada Array \$sayur kita gunakan tanda kurung siku [] dan menuliskan isi array diantara kurang siku pembuka ([) dan kurung siku penutup (]).

Menampilkan data array dengan menggunakan foreach

Untuk menampilkan isi array, kita bisa mengaksesnya melalui indeks. Baik dengan memanfaatkan perulangan 'foreach' atau menampilkan satu-persatu.



```
1 <?php
2 // membuat array
3 $barang = ["Buku Tulis", "Penghapus", "Spidol"];
4
5 // menampilkan isi array
6 echo $barang[0]."<br>";
7 echo $barang[1]."<br>";
8 echo $barang[2]."<br>";
9 ?>
```

Buku Tulis
Penghapus
Spidol

Contoh array dan menampilkan isi array satu persatu

Apabila isi array terlalu banyak, dan kita kerepotan dalam menampilkan, maka kita dapat memanfaatkan perulangan untuk menampilkan array tersebut:



```
1 <?php
2 $nama_siswa = ['Anton','Budi','Cika','Danang','Erika'];
3 foreach($nama_siswa as $row)
4 {
5     echo $row;
6     echo "<br/>";
7 }
8 ?>
```

Anton
Budi
Cika
Danang
Erika

Contoh array dan menampilkan isi array dengan foreach

Menghapus Isi Array

Untuk menghapus isi array, kita bisa menggunakan fungsi **unset()**. Fungsi ini juga dapat digunakan untuk menghapus variabel.



```
1 <?php
2 // membuat array
3 $hewan = [
4     "Burung",
5     "Kucing",
6     "Ikan"
7 ];
8
9 // menghapus kucing
10 unset($hewan[1]);
11
12 echo $hewan[0]."<br>";
13 echo $hewan[1]."<br>";
14 echo $hewan[2]."<br>";
15
16 echo "<hr>";
17
18 echo "<pre>";
19 print_r($hewan);
20 echo "</pre>";
21 ?>
```

Burung
Ikan

Array
(
 [0] => Burung
 [2] => Ikan
)

Contoh menghapus array dengan unset()

Array Assosiatif

Array asosiatif adalah array yang indeksnya tidak menggunakan nomer atau angka. Indeks array asosiatif berbentuk kata kunci. Array Asosiatif adalah data array yang menggunakan kata kunci atau index yang kita tentukan sendiri. Dalam pembuatan array asosiatif biasanya menggunakan index atau kata kunci bertipe data string, pada prakteknya kita bebas menggunakan tipe data apapun. Di PHP, jika kita ingin membuat

array asosiatif, kita diharuskan menggunakan operator => sebagai penghubung antara index/ kata kunci dengan value/ nilainya. Terdapat 2 cara dalam membuat data array asosiatif, yaitu

- Menggunakan array () construct
- Menggunakan operator array []

Perhatikan contoh pertama, kita akan membuat data array asosiatif dengan menggunakan array() :



```
1 <?php
2 // membuat array asosiatif
3 $artikel = [
4     "judul" => "Belajar Pemrograman PHP",
5     "penulis" => "supardianto",
6     "view" => 128
7 ];
8
9 // mencetak isi array assosiatif
10 echo "<h2>".$artikel["judul"]."</h2>";
11 echo "<p>oleh: ".$artikel["penulis"]."</p>";
12 echo "<p>View: ".$artikel["view"]."</p>";
13 ?>
```

Contoh baris program array assosiatif

Pada array asosiatif, kita menggunakan tanda => untuk mengasosiasikan sebuah kata kunci dengan isi array. Selain menggunakan tanda =>, kita juga bisa membuat array aosiatif seperti ini:



```
1 <?php
2 $email["subjek"] = "Apa Kabar?";
3 $email["pengirim"] = "dian@polibatam.com";
4 $email["isi"] = "Apa kabar? sudah lama tidak berjumpa";
5
6 echo "<pre>";
7 print_r($email);
8 echo "</pre>";
9 ?>
```

Array

(

```
[subjek] => Apa Kabar?
[pengirim] => dian@polibatam.com
[isi] => Apa kabar?
```

)

Contoh baris program assosiatif dengan array []

Array Multidimensi

Array Multidimensi adalah data array yang berisi satu atau lebih data array. Ini dapat dikatakan array didalam array, singkatnya bahwa kita dapat memiliki banyak level indeks didalam array multi dimensi. Katakanlah kita mempunyai kotak yang didalamnya berisi berbagai macam barang, jika kita bayangkan array multi dimensi seperti kotak didalam kontak, dan didalamnya lagi terdapat kotak lagi, dan seterusnya.

Dimensi array menunjukkan jumlah indeks atau kata kunci yang kita butuhkan. Misal pada array dua dimensi, berarti kita memiliki dua index atau kata kunci. Jika array tiga dimensi, maka kita memiliki tiga index atau kata kunci. Dan seterusnya, tergantung kebutuhan kita.

Tetapi pada prakteknya, yang paling banyak digunakan adalah array dua dimensi, dan pada level tinggi programmer juga sering menggunakan array 3 dimensi atau bahkan lebih. Akan tetapi, array dengan dimensi lebih dari 3 tingkat akan sulit untuk ditangani. Membutuhkan kecermatan dan ketelitian yang tinggi.



```
1 <?php
2 // membuat array 2 dimensi yang berisi array asosiatif
3 $artikel = [
4   [
5     "judul" => "Belajar PHP & MySQL untuk Pemula",
6     "penulis" => "supardianto"
7   ],
8   [
9     "judul" => "Tutorial PHP dari Nol hingga Mahir",
10    "penulis" => "sartono"
11  ],
12  [
13    "judul" => "Membuat Aplikasi Web dengan PHP",
14    "penulis" => "sartini"
15  ]
16 ];
17
18 // menampilkan array
19 foreach($artikel as $post){
20   echo "<h2>".$post["judul"]."</h2>";
21   echo "<p>".$post["penulis"]."</p>";
22   echo "<hr>";
23 }
24 ?>
```

Belajar PHP & MySQL untuk Pemula
supardianto

Tutorial PHP dari Nol hingga Mahir
sartono

Membuat Aplikasi Web dengan PHP
sartini

Contoh baris program array multidimensi

XI FILE HANDLING

PHP juga dapat berinteraksi dengan file yang memungkinkan pengguna melakukan upload file dari browsernya. Ketika melakukan interaksi dengan file yang dikirim dari browser, ada perlakuan khusus dari PHP terhadap kiriman tersebut sebelum akhirnya disimpankan kedalam folder server. Dan ketika dilakukan klik pada form maka server akan menerima kiriman variabel dari form browser.

PHP menyediakan sejumlah fungsi yang dapat digunakan untuk melakukan interaksi dengan file. Ada fungsi untuk melakukan pembuatan, pembukaan, penulisan dan penghapusan file. Urutan pengaksesan file adalah seperti yang ditunjukkan pada gambar:



Urutan akses file pada PHP

Sebuah file dapat dibayangkan seperti buku. Bila ingin membaca isi sebuah buku tersebut , maka perlu membuka buku terlebih dahulu. Serupa dengan hal tersebut maka bisa dilakukan operasi pembacaan atau perekaman data pada berkas. Selanjutnya jika tidak lagi bermaksud menggunakan berkas tersebut, maka perlu menutup berkas.

Memeriksa File

Pemeriksaan apakah file yang akan kita gunakan dalam aplikasi yang kita buat sangat penting. Pemeriksaan file memberikan kepastian kepada program apakah file yang akan diproses sudah ada atau belum, jika belum ada Apakah perlu dibuat dulu filenya atau langsung memberikan pesan peringatan kepada pemakai bahwa file yang akan digunakan atau diproses tidak ada. Skrip secara umum untuk pemeriksaan file adalah: `file_exists("nama file")`.

```
1 <?php
2 if(file_exists("coba.txt")){
3     echo "File tersedia";
4 }else{
5     echo "File yang di cari tidak ada !";
6 }
7 ?>
```

Contoh baris program memeriksa file

Membuka File

Untuk mengakses dan mengubah isi file, maka dibutuhkan file descriptor adalah suatu variable yang digunakan untuk mewakili file tertentu. File descriptor didapat dengan menggunakan fungsi PHP: fopen() untuk membuka file. Bentuk pemanggilan fungsi ini adalah **fopen (nama_berkas , mode)**. Fungsi fopen membutuhkan 2 parameter yaitu nama_berkas dan mode (jenis akses).

Nama berkas bertipe string yang merupakan nama file yang akan dibuka. Harus berada dalam direktori yangsama dengan tempat script php yang memanggilnya. Mode (jenis akses) bertipe string yang digunakan untuk menentukan perlakuan yang diperbolehkan terhadap file yang dibuka. Ada beberapa mode (jenis akses) yaitu:

- r : Membuka file hanya untuk dibaca saja.
- w : Membuka file hanya untuk ditulisi. Isi didalam file tersebut akan dihapus, jika file belum tersedia maka akan menciptakan file baru.
- a : Membuka file hanya untuk ditulisi. Isi didalam file tersebut tidak dihapus, jika file belum tersedia maka akan menciptakan file baru.
- x : Membuat file baru untuk menulis saja.
- r+: Membuka file untuk membaca dan juga ditulisi.
- w+: Membuka file untuk membaca dan juga ditulisi. Menghapus isi file atau membuat file baru jika tidak ada.
- a+: Membuka file untuk dibaca dan juga ditulisi. Membuat file baru jika file tidak ada
- x+: Membuat file baru untuk dibaca / ditulisi.

Berikut contoh dengan menggunakan mode w:

```
1 <?php
2 $fileku = fopen("nama_file.txt", "w");
3 $tulisi = "Bambang Subandono";
4 fwrite($fileku, $tulisi);
5 fclose($fileku);
6 //Hasil didalam fileku.txt berisi tulisan: Bambang Subandono
7 ?>
```

Pada contoh mengolah file dengan PHP diatas, kita asumsikan bahwa dokumen PHP diatas berada dalam satu direktori yang sama dengan file yang akan diolah. Maka dapat diuraikan penjelasan sebagai berikut:

- Kita membuat variabel **\$fileku** dan didalam variabel tersebut kita buka sebuah file dengan nama **nama_file.txt** dan menggunakan model tindakan **w**. Dengan begitu jika tidak terdapat file dengan nama **nama_file.txt** maka PHP akan menciptakan file itu. Jika telah terdapat file tersebut, maka isi dari file akan dihapus dan digantikan dengan isi yang akan dituliskan.
- Variabel **\$tulisi** adalah data atau teks yang akan saya tuliskan kedalam file **nama_file.txt**
- Kemudian penulisan sintak **fwrite(\$fileku, \$tulisi)** adalah proses membuka file dan menuliskan isi kedadamnya.
- Dengan fungsi **fclose(\$fileku)** berarti saya telah menutup file yang telah ditulisi tersebut.

Membaca File

Setelah file berhasil dibuka maka data dapat dilakukan pembacaan file atau memanipulasi file. Ada beberapa fungsi yang digunakan untuk melakukan manipulasi file diantaranya:

Fungsi Menyimpan Data ke File

Proses penyimpanan data ke dalam file membutuhkan fungsi **fwrite()**. Bentuk pemanggilannya: **fwrite(variabel, data)**. Dalam hal ini argumen pegangan menyatakan pegangan file yang diperoleh ketika memanggil **fopen**. Dan argument data (bertipe string) menyatakan data yang akan direkam. Fungsi memberikan nilai balik berupa: TRUE kalau data berhasil direkam FALSE jika data gagal direkam.

```
1 <?php
2 $myfile = fopen("newfile.txt", "w")
3 or die("Unable to open file!");
4 $txt = "John Doe\n";
5 fwrite($myfile, $txt);
6 $txt = "Jane Doe\n";
7 fwrite($myfile, $txt);
8 fclose($myfile);
9 ?>
```

Fungsi Membaca Data ke File

Proses pembacaan data yang terdapat pada file membutuhkan fungsi **fread()**. Fungsi **fread()** merupakan kebalikan dari **fwrite()**. Bentuk pemanggilannya:

```
1 <?php
2 $myfile = fopen("webdictionary.txt", "r")
3 or die("Unable to open file!");
4 echo fread($myfile,filesize("webdictionary.txt"));
5 fclose($myfile);
6 ?>
```

XII MENGIRIM DATA MELALUI FORM

Elemen Penting pada Form

Form adalah sebuah halaman di web yang digunakan untuk mengumpulkan informasi melalui inputan pengguna. Untuk merancang sebuah form, ada 3 elemen penting yaitu :

1. Atribut Method

Method dari sebuah form digunakan untuk menentukan bagaimana data dikirim dan diproses oleh PHP. Method terdiri dari 2 macam yaitu GET dan POST.

2. Atribut Action

Action menentukan dimana data yang diinput oleh form akan diproses. Biasanya action diisi dengan nama file php. Jika action dikosongkan, proses dari form terjadi di halaman yang sama. Jadi halaman form dan halaman proses bisa digabung menjadi satu halaman atau terpisah. Contoh: membuat nilai **action="proses.php"**, yang berarti kita harus menyediakan sebuah file dengan nama: **proses.php** untuk memproses form tersebut.

3. Submit Button

Submit button adalah tombol yang berfungsi sebagai trigger pengiriman data dari form. Jika Submit button ditekan, maka data yang diinput akan diproses.

Contoh sebuah baris kode Form:

```
● ● ●  
1 <form action="" method="post">  
2 Nama: <input type="text" name="nama">  
3 Alamat: <input type="text" name="alamat"><br><br>  
4 <input type="submit" value="submit">  
5 </form>
```

Contoh form dengan action tidak diisi

Contoh di atas menggunakan method post dan action dikosongkan, artinya proses form terjadi di halaman yang sama.

```
● ● ●  
1 <form action="proses.php" method="get">  
2 Nama: <input type="text" name="nama">  
3 Alamat: <input type="text" name="alamat"><br><br>  
4 <input type="submit" value="submit">  
5 </form>
```

Contoh form dengan action pada halaman lain

Contoh di atas menggunakan method get dan action proses.php, artinya proses form terjadi di halaman proses.php.

Jenis-jenis type yang dapat digunakan Form

Terdapat berbagai macam jenis elemen input di dalam form, seperti: *text, password, textarea, select, select multiple, checkbox, radio, file, submit/button, dan hidden*. Dan beberapa tipe input baru pada HTML 5 yang masih belum seluruhnya didukung pada browser seperti; *email, url, number, date, month, week, time, range, dan color*.

Semua elemen input pada form bergantung dengan atribut *name* dalam pengiriman data. Saat data-data pada form dikirim ke server, nama yang tertera di dalam atribut *name* tersebut akan menjadi nama data yang dikirim. Selain atribut *name*, setiap elemen input juga dapat ditambahkan atribut *readonly* dan *disabled*. Atribut *readonly* akan menyebabkan input tidak dapat diedit, sedangkan atribut *disabled* menyebabkan input tidak interaktif dan datanya tidak dikirim, hanya ditampilkan saja.

Input text pada HTML

Digunakan untuk menginput teks. Merupakan tipe default dari elemen input.

```
1 <label for="fname">First name:</label><br>
2 <input type="text" id="fname" name="fname"><br>
```

First name:

Input password pada HTML

Input ini sama dengan input tipe teks, hanya saja teksnya tersembunyi.

```
1 <label>Password:</label><br>
2 <input type="password" name="password" />
```

Password:

Input textarea pada HTML

Textarea dapat digunakan untuk input teks dengan banyak baris. Tidak seperti elemen input, nilai awal pada elemen textarea tidak diletakkan di atribut value, tapi diletakkan di antara tag pembuka dan penutup textarea.

```
1 <label>Alamat:</label><br>
2 <textarea name="alamat">Jl. Suka Suka</textarea>
```

Alamat:

Input select pada HTML

Input select digunakan untuk memilih di antara beberapa pilihan.

```
1 <label>Hobi:</label><br>
2 <select name="hobi">
3   <option value="">--Pilih Hobi--</option>
4   <option value="1">1. Komputer</option>
5   <option value="2">2. Otomotif</option>
6   <option value="3">3. Memasak</option>
7   <option value="4">4. Melukis</option>
8   <option value="5">5. Bernyanyi</option>
9 </select>
```

Hobi: 

- Pilih Hobi--
- 1. Komputer
- 2. Otomotif
- 3. Memasak
- 4. Melukis
- 5. Bernyanyi

Pilihan di dalam elemen select dinyatakan dengan elemen *option*. Teks yang tampil diapit oleh tag *option*, sedangkan nilai yang dikirim ke server dinyatakan dalam atribut value pada elemen *option*.

Input checkbox pada HTML

Checkbox pada HTML menggunakan elemen input juga, hanya typenya yang diganti menjadi checkbox.

```
1 <label>Hobi:</label><br>
2 <input type="checkbox" name="hobi[]" value="1"/>1. Komputer<br />
3 <input type="checkbox" name="hobi[]" value="2"/>2. Otomotif<br />
4 <input type="checkbox" name="hobi[]" value="3"/>3. Memasak<br />
5 <input type="checkbox" name="hobi[]" value="4"/>4. Melukis<br />
6 <input type="checkbox" name="hobi[]" value="5"/>5. Bernyanyi<br />
```

Hobi: 1. Komputer
 2. Otomotif
 3. Memasak
 4. Melukis
 5. Bernyanyi

Jika mengirim data ke server dengan bahasa PHP, atribut name pada checkbox perlu diberi tanda kurung siku seperti di atas agar data hobi menjadi suatu array. Nilai yang terkirim ke server adalah nilai di dalam atribut value. Sedangkan teks yang mendampingi checkbox dapat diketikkan setelah tag input checkbox.

Input radiobutton pada HTML

Tipe input radio memiliki fungsi yang sama dengan input select. Memilih satu di antara banyak. Saya lebih menganjurkan menggunakan select karena tampilannya lebih ringkas.

```
1 <label>Hobi:</label><br>
2 <input type="radio" name="hobi" value="1"/>1. Komputer<br />
3 <input type="radio" name="hobi" value="2"/>2. Otomotif<br />
4 <input type="radio" name="hobi" value="3"/>3. Memasak<br />
5 <input type="radio" name="hobi" value="4"/>4. Melukis<br />
6 <input type="radio" name="hobi" value="5"/>5. Bernyanyi<br />
```

Hobi:

- 1. Komputer
- 2. Otomotif
- 3. Memasak
- 4. Melukis
- 5. Bernyanyi

Karena radio button bersifat pilihan, tidak perlu menggunakan array pada atribut *name*. Dan sama seperti checkbox, atribut checked pada radio button menentukan opsi yang terpilih sejak awal.

Input file pada HTML

Input file pada HTML berguna untuk memilih dan mengirimkan file ke server. Agar dapat mengirimkan file, elemen form yang berisi input file harus ditambahkan atribut *enctype="multipart/form-data"*.

```
1 <form method="post" action="" enctype="multipart/form-data">
2   <input type="file" name="pilih_file" />
3 </form>
```

Output:

Pilih File Upload

No file chosen

Input file memiliki tampilan yang berbeda-beda pada setiap browser.

Input submit pada HTML

Input submit berbentuk seperti tombol dan biasanya diletakkan di akhir form. Ketika diklik, input submit akan mengirimkan isi form ke server, ke alamat yang tertera pada atribut action pada elemen form.

```
1 <button type="submit" value="Daftar">Daftar</button> 
```

Input hidden pada HTML

Seperti namanya, input hidden tidak tampil pada halaman HTML. Biasanya digunakan untuk meneruskan atau menyimpan variabel dari server pada halaman client.

```
1 <input type="hidden" name="username" value="cecep" />
```

Karena digunakan untuk menyimpan nilai, biasanya tipe input ini selalu disertai atribut name dan value. Akan tetapi dengan teknologi ajax dan json, tipe input ini sudah agak jarang dipakai.

Input email pada HTML 5

Tipe input ini pada dasarnya sama seperti input teks, terkecuali bila digabung dengan atribut required, browser langsung memvalidasi apakah input tersebut valid atau tidak valid.

```
1 <form action="" method="post">
2 email: <input type="email" name="email" required /> email: 
3 <br>
4 <br>
5 <input type="submit" value="submit"/>
6 </form>
```

Output apabila format email yang dimasukan tidak valid:

The screenshot shows a form with an email input field containing "supar". A validation message box appears below the input field, stating: "Please include an '@' in the email address. 'supar' is missing an '@'." The entire screenshot is enclosed in a light gray border.

Input number pada HTML 5

Tipe number juga sama dengan input text, kecuali input number hanya dapat diisi dengan angka.

```
1 <input type="number" name="qty" /> 
```

Input tanggal pada HTML 5

Terdapat beberapa tipe input tanggal, antara lain: *date*, *month*, *week* dan *time*. Akan tetapi tipe input *month*, *week*, dan *time* ini baru didukung oleh chrome saja.

```
1 Tanggal: <input type="date" name="tanggal" /> Tanggal: 
```

Variabel Global \$_GET

`$_GET` digunakan untuk mengirimkan informasi melalui URL untuk ditampilkan pada halaman web, informasi bisa berupa variabel maupun nilai variabel tersebut. Biasanya informasi tersebut berasal dari form pengisian data yang dilakukan oleh pengguna.

Informasi yang dikirim melalui metode GET ini bisa dilihat oleh semua orang, untuk itu disarankan tidak mengirimkan informasi rahasia seperti password melalui metode GET. Metode GET mempunyai batasan data / informasi yang bisa dikirim, yaitu sekitar 2000 karakter. GET tidak dapat digunakan untuk mengirim data biner seperti gambar atau dokumen.

Tujuan Penggunaan

Tujuan menggunakan metode GET adalah agar halaman website dapat dikelola secara dinamis.

Penggunaan GET

Misalkan, buat file baru dengan nama **get.php**, lalu tulis kode berikut:

```
1 <html>
2 <body>
3 <form action="proses.php" method="get">
4 Nama: <input type="text" name="nama">
5 Alamat: <input type="text" name="alamat"><br><br>
6 <input type="submit" value="submit">
7 </form>
8 </body>
9 </html>
```

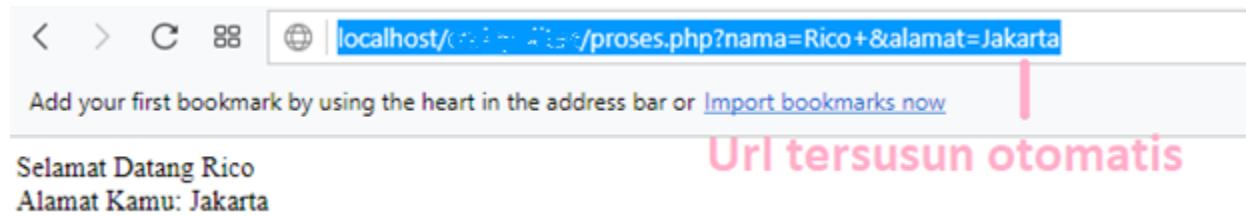
Nama:

Alamat:

Lalu buat kembali file **proses.php**, lalu tulis kode berikut:

```
1 <html>
2 <body>
3 Selamat Datang <?php echo $_GET["nama"]; ?><br>
4 Alamat Kamu: <?php echo $_GET["alamat"]; ?>
5 </body>
6 </html>
```

Lalu output yang dihasilkan apabila kita mengisi form, dan meng-klik submit adalah:



Jika kita memiliki tag dengan kode HTML `<input type="text" name="nama" />`, maka untuk mengakses nilainya adalah dengan `$_GET['nama']`, dan untuk tag `<input type="text" name="alamat" />` diakses dengan nilai `$_GET['alamat']`.

Variabel Global `$_POST`

Selain menggunakan GET, untuk mengumpulkan informasi dari form, Kita juga bisa menggunakan POST. `$_POST` digunakan untuk mengirimkan informasi melalui metode HTTP POST. Informasi yang dikirim dari form dengan metode POST tidak dapat dilihat oleh orang lain dan tidak memiliki batasan pada jumlah informasi / data yang akan dikirim. POST dapat digunakan untuk mengirimkan data biner seperti gambar dan dokumen.

Tujuan Penggunaan

Tujuan menggunakan metode POST adalah agar halaman website dapat dikelola secara dinamis.

Penggunaan GET

Misalkan, buat file baru dengan nama **post.php**, lalu tulis kode berikut:

```

1 <html>
2 <body>
3 <form action="proses2.php" method="post">
4 Nama: <input type="text" name="nama">
5 Alamat: <input type="text" name="alamat"><br><br>
6 <input type="submit" value="submit">
7 </form>
8 </body>
9 </html>

```

Nama:

Alamat:

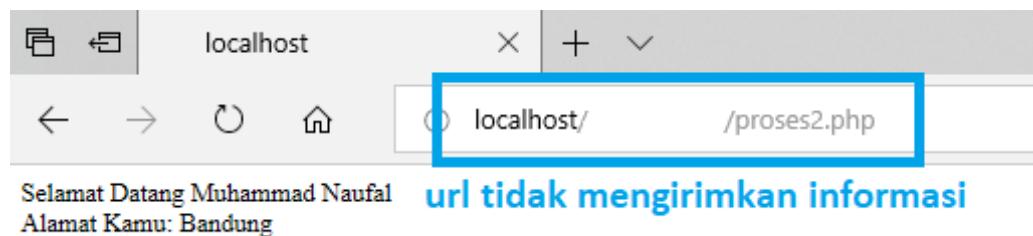
Lalu buat kembali file **proses2.php**, lalu tulis kode berikut:

```

1 <html>
2 <body>
3 Selamat Datang <?php echo $_POST["nama"]; ?><br>
4 Alamat Kamu: <?php echo $_POST["alamat"]; ?>
5 </body>
6 </html>

```

Lalu output yang dihasilkan apabila kita mengisi form, dan meng-klik submit adalah:



Perbedaan Metode Pengiriman Form GET dan POST dalam PHP

Salah satu pertimbangan dasar dalam membuat dan memproses form PHP adalah apakah form tersebut dikirim menggunakan **method=get** atau **method=post**.

Keunggulan dan Kelemahan Method Form ‘GET’

Kelemahan yang paling jelas jika kita menggunakan **method=get** adalah nilai dari form dapat dilihat langsung di dalam URL yang dikirimkan. Jika kita membuat form untuk data-data yang sensitif seperti password, maka form dengan **method=get** bukanlah pilihan yang tepat.

Form dengan **method=get** disarankan untuk form yang berfungsi menampilkan data, yaitu dimana hasil isian form hanya digunakan untuk menampilkan data, sesuai dengan arti kata get yang bisa berarti: ‘ambil’. Sehingga **method=get** sebaiknya digunakan untuk form yang ‘mengambil’ data dari database.

Salah satu penggunaan **method=get** yang umum digunakan adalah pada form pencarian (form search). Dengan membuat hasil inputan form terlihat jelas dalam URL, pengunjung web bisa dengan mudah menebak ‘alur kerja’ dari situs kita.

Keunggulan dan Kelemahan Method Form ‘POST’

Keuntungan menggunakan **method=post** dalam pembuatan form PHP adalah bahwa isi dari form tidak ditampilkan di URL, sehingga method ini sesuai untuk data-data yang bersifat sensitif seperti *username* dan *password*.

Sesuai dengan namanya, **method=post** disarankan digunakan untuk form yang digunakan untuk menambah data (**posting data**) atau sebuah proses yang akan mengubah isi database, seperti query *Insert*, *Delete*. Contoh penggunaan **method=post** ini sering digunakan untuk *form register* dan *form login* yang dianggap bersifat rahasia.

Validasi Form PHP (isset dan empty)

Setelah berhasil mengambil dan menampilkan nilai dari form, hal berikutnya yang harus kita lakukan terhadap data tersebut adalah melakukan proses validasi. Validasi dapat dilakukan dari sisi client atau juga server.

Penggunaan attribut seperti *required* pada tiap input, juga merupakan bagian dari validasi dari sisi client, yaitu kita wajibkan bahwa inputan tersebut harus diisi, tidak boleh kosong. Sedangkan dari sisi server kita bisa menggunakan fungsi **isset()** atau **empty()**.

Pentingnya Melakukan Validasi Nilai Form

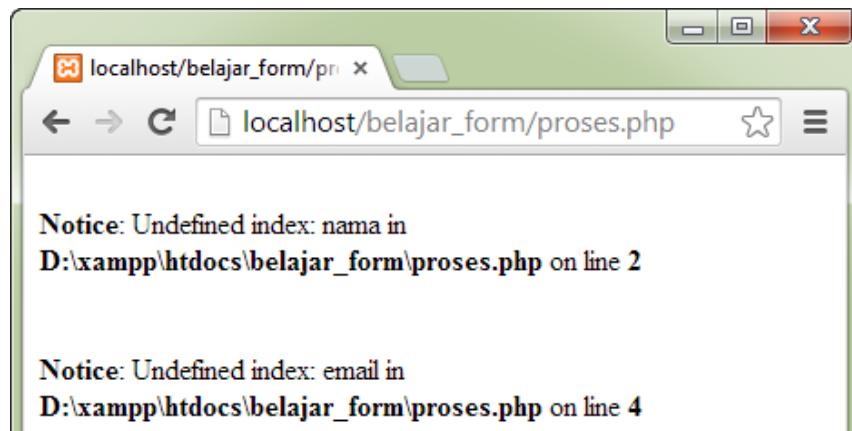
Nilai yang telah diinput oleh user atau pengunjung web, tidak bisa begitu saja di simpan langsung ke dalam database. Karena kita tidak tahu apakah nilai tersebut telah sesuai dengan nilai yang kita kehendaki. Misalkan apakah nilai tersebut harus berupa angka, atau hanya bisa berupa huruf, atau apakah hanya bisa diinput dalam range tertentu saja.

Dalam kasus yang ekstrim, seorang user bisa saja memasukkan kode script atau tag HTML yang bisa merusak situs kita, hal ini dikenal dengan **Cross-site Scripting**.

Memeriksa Ketersediaan Variabel Form dengan Fungsi **isset()**

Validasi pertama yang paling sederhana dan ‘hampir’ selalu ada dalam tiap proses validasi form dalam PHP adalah memeriksa apakah objek form tersebut sudah tersedia atau tidak. Sebagai contoh sederhananya: apakah variabel **\$_GET['nama']** tersedia untuk diproses atau tidak.

Proses memeriksa ‘ketersediaan’ variabel ini menjadi penting karena PHP akan mengeluarkan pesan peringatan jika kita mengakses nilai sebuah variabel yang belum didefinisikan terlebih dahulu.



Contoh pesan error apabila variabel belum tersedia

Notice: Undefined index adalah pesan error yang terjadi karena kita langsung menampilkan variabel `$_GET['nama']` dan `$_GET['email']` yang memang belum diset sebelumnya.

Untuk memeriksa apakah sebuah objek form telah didefinisikan atau telah di-set sebelumnya, kita bisa menggunakan fungsi bawaan PHP: `isset()`. Fungsi `isset()` akan menghasilkan nilai true jika sebuah variabel telah didefinisikan, dan false jika variabel tersebut belum dibuat.



```
1 <?php  
2 if (isset($_GET['nama']) AND isset($_GET['email']))  
3 {  
4     echo $_GET['nama'];  
5     echo $_GET['email'];  
6 }  
7 else  
8 {  
9     echo "Maaf, anda harus mengakses halaman ini dari  
    form.html";  
10 }  
11 ?>
```

Contoh penerapan `isset()` pada proses

Pada kode PHP diatas mengharuskan nilai `$_GET['nama']` dan `$_GET['email']` tersedia, baru nilai ditampilkan, namun jika tidak ada, akan ditampilkan pesan bahwa halaman ini hanya bisa diakses dari `form.html`.

Memeriksa Apakah Variabel Form Telah Diisi

Fungsi `isset()` yang kita bahas sebelumnya hanya memeriksa apakah sebuah objek form ada atau tidak. Fungsi `isset()` tetap bernilai true meskipun user tidak mengisi form sama sekali (variabel form bernilai kosong, namun variabel tersebut dianggap telah di-set).

Untuk memeriksa apakah sebuah objek form telah diisi atau tidak, kita bisa menggunakan fungsi: `empty()`.

Fungsi `empty()` akan menghasilkan nilai false jika sebuah variabel telah diisi, dan bernilai true jika variabel tersebut belum diisi. Dengan menggunakan struktur IF dan fungsi `empty()`, kita bisa membuat logika validasi

objek form mana saja yang dianggap perlu (harus diisi) dan mana yang boleh dikosongkan. Dan kemudian menampilkan pesan error yang sesuai.



```
1 <?php
2 if (isset($_GET['nama']) AND isset($_GET['email']))
3 {
4     $nama=$_GET['nama'];
5     $email=$_GET['email'];
6 }
7 else
8 {
9     die("Maaf, anda harus mengakses halaman ini dari form.html");
10}
11
12 if (!empty($nama))
13 {
14     echo "Nama: $nama <br /> Email: $email";
15 }
16 else
17 {
18     die("Maaf, anda harus mengisi nama");
19 }
20 ?>
```

Contoh penerapan isset dan empty pada proses

Keterangan:

1. Pada logika IF pertama, saya melakukan pengecekan apakah variabel `$_GET['nama']` dan `$_GET['email']` tersedia atau tidak. Jika tersedia maka pindahkan nilainya ke variabel `$nama` dan `$email` agar lebih mudah untuk diproses. Namun jika tidak, fungsi `die()` akan menghentikan proses dan menampilkan pesan kesalahan.
2. Pada logika IF kedua, saya memeriksa apakah variabel `$nama` kosong atau tidak dengan fungsi `!empty()`. Fungsi `!empty($nama)` akan menghasilkan nilai true hanya jika variabel `$nama` tidak kosong (perhatikan tanda `!` sebagai pembalik logika `empty()`). Namun jika `$nama` ternyata kosong (tidak diisi), maka tampilkan pesan kesalahan.

Mencegah XSS dan HTML Injection

Cross-site Scripting atau sering disingkat dengan **XSS** adalah jenis serangan ke sebuah situs dengan cara 'menyisipkan' kode script (biasanya **JavaScript**) ke dalam sebuah situs. Hal ini hanya akan berhasil jika situs tersebut memiliki fitur untuk menampilkan kembali isian form ke web browser, seperti form komentar. Sedangkan **HTML injection** adalah istilah yang lebih spesifik kepada cara 'menyisipkan' kode HTML kedalam sebuah situs.

Sebagai programmer web, penanganan untuk **Cross-site Scripting** maupun **HTML injection** merupakan hal yang sangat penting, terutama dalam pembuatan kode form dengan PHP. Karena form pada dasarnya dapat diinput oleh siapa saja, maka kita perlu memproteksi situs dari kode-kode berbahaya yang bisa diinput oleh user melalui form.

Form yang memiliki kelemahan terhadap XSS ketika input form nya di isikan baris kode berikut,

```
1 <script>alert('Percobaan XSS')</script>
```

Maka ketika hasil nya akan ditampilkan, maka akan meng-eksekusi perintah javaScript tersebut. Hal ini sangat berbahaya karena dengan kode JavaScript seseorang bisa melakukan 'hampir segalanya' dengan situs kita. Tidak hanya sekedar menampilkan ucapan selamat seperti kode diatas, tetapi seseorang juga bisa merubah background, mengubah tampilan seluruh web, bahkan mengarahkan pengunjung ke situs lain (redirect).

Browser Chrome memiliki filter terhadap url yang dikunjungi oleh pengguna, jika pengguna mengunjungi url yang mengandung kode XSS maka Chrome akan memberikan peringatan bahwa url yang dikunjungi mengandung kode XSS.

Cara Mencegah Cross-site Scripting dan HTML injection

Salah satu cara sederhana untuk menghindari Cross-site Scripting dan HTML injection adalah dengan membuat karakter-karakter yang memiliki 'makna' di dalam HTML dan JavaScript untuk diubah menjadi named entity, yaitu menkonversi karakter khusus seperti < menjadi <, dan karakter > menjadi >. Atau cara lainnya adalah dengan menghilangkan sama sekali seluruh tag HTML atau script dari inputan user.

Untuk kedua keperluan ini, PHP memiliki fungsi **htmlspecialchars()** dan **fungsi strip_tags()**.

Fungsi **htmlspecialchars()** akan mengonversi 4 karakter 'khusus' HTML menjadi named entity sehingga tidak akan di 'proses' oleh web browser. Keempat karakter tersebut adalah: <, >, & dan ". Keempat karakter khusus inilah yang membuat web browser akan menerjemahkan sebuah string menjadi kode HTML/JavaScript.

Sedangkan fungsi **strip_tags()** akan menghapus seluruh tag HTML dari inputan user.

```
1 $nama=$_GET['nama'];
2 $email=$_GET['email'];
3 $nama=htmlspecialchars($nama);
4 $email=strip_tags($email);
```

Mencegah Cross Site Request Forgery (CSRF)

Dalam serangan **Cross Site Request Forgery (CSRF)**, penyerang mengelabuhi korban untuk memuat informasi sensitif atau melakukan transaksi tanpa sepengetahuan mereka. Hal ini terutama terjadi pada aplikasi web yang buruk kode programnya untuk memicu logika bisnis menggunakan permintaan GET.

Idealnya, permintaan GET idempotent secara alamiahnya. **Idempotency** berarti halaman yang sama dapat diakses beberapa kali tanpa menimbulkan efek samping. Oleh karena itu, permintaan GET harus digunakan hanya untuk mengakses informasi dan tidak untuk melakukan transaksi.

Solusinya adalah untuk memproses fungsi yang mengubah keadaan database dalam permintaan POST, dan hindari menggunakan **\$_REQUEST**. Gunakan **\$_GET** untuk mengambil parameter GET, dan menggunakan **\$_POST** untuk mengambil parameter POST.

Selain itu, harus ada token acak disebut token CSRF terkait dengan setiap permintaan POST. Ketika user login ke akunnya, aplikasi harus menghasilkan token acak dan menyimpannya dalam sesi. Setiap kali form apapun akan ditampilkan kepada pengguna, token harus hadir di halaman sebagai field input tersembunyi. Logika aplikasi harus memeriksa token dan memastikan bahwa itu cocok dengan yang ada dalam session saat ini.

Langkah pertama, membuat random token csrf (biasa dibuat bersamaan ketika user login), melalui kode berikut:

```
1 <?php
2 session_start();
3 $length = 32;
4 $_SESSION['token'] = substr(base_convert(sh1(uniqid(mt_rand()))), 16, 36), 0, $length);
5 ?>
```

Langkah kedua, Setiap kali form apapun akan ditampilkan kepada pengguna, token harus hadir di halaman sebagai field input tersembunyi. Lihat baris program berikut:

```
1 <?php
2 session_start();
3 $token = $_SESSION['token'];
4 ?>
5 <form action="proses.php" method="post">
6 <input type="hidden" id="token" name="token" value="<?php echo $token; ?>">
7 <input type="text" name="nama"/>
8 <button type="submit">Kirim</button>
9 ?>
```

Langkah ketiga, pada file proses.php, lakukan pengecekan token tersebut:

```
1 <?php
2 session_start();
3 if ($_SESSION['token']==$_POST['token']) {
4     // VALID TOKEN PROVIDED - PROCEED WITH PROCESS
5 } else {
6     // ERROR - INVALID TOKEN
7 }
8 ?>
```

XIII C.R.U.D DENGAN PHP DAN MYSQL

Koneksi PHP dan MySQL

PHP memiliki 3 cara pengaksesan MySQL, yakni melalui PDO (PHP Data Objects), mysqli extension dan mysql extension. PDO menggunakan pemrograman objek, mysqli extension tersedia dalam bentuk objek dan prosedural (diakses melalui fungsi-fungsi) sedangkan mysql extension sepenuhnya menggunakan pemograman prosedural.

1. Koneksi MySQL dengan mysql extension

Saat pertama kali mempelajari PHP MySQL sekitar tahun 2008 (atau jika anda pernah mempelajari PHP-MySQL beberapa tahun yang lalu), untuk mengakses MySQL dari PHP, kita menggunakan fungsi-fungsi seperti *mysql_connect()*, *mysql_query()*, dan *mysql_fetch_array()*. Fungsi-fungsi ini tergabung ke dalam mysql extension (saat itu PDO dan mysqli extension masih jarang digunakan)

Namun sekarang (tepatnya mulai PHP versi 5.5.0) PHP memutuskan untuk membuat mysql extension berstatus *deprecated*. Yang artinya pengaksesan database MySQL menggunakan fungsi mysql extension sudah tidak disarankan lagi. Programmer PHP diharapkan pindah ke *mysqli extension* atau PDO yang berbasis objek.

Alasannya, MySQL versi terbaru memiliki fitur-fitur yang semakin lengkap dan kompleks, sehingga PHP memutuskan untuk membuat fungsi extension baru agar programmer PHP bisa menggunakan fitur-fitur ini.

2. Koneksi MySQL dengan mysqli extension

Sebagai pengganti mysql extension, PHP menyediakan mysqli extension (mysqli merupakan singkatan dari MySQL Improved). Mysqli extension ini pada dasarnya adalah perbaikan dari mysql extension dan dikembangkan untuk mendukung fitur-fitur terbaru untuk MySQL 4.1 keatas.

Hampir semua fungsi yang ada pada mysql extension juga tersedia pada mysqli. Syntax (aturan penulisan) mysqli sangat mirip dengan mysql extension. Sehingga jika anda telah lama menggunakan mysql extension, akan sangat mudah untuk beralih menggunakan mysqli extension.

3. Koneksi MySQL dengan PDO (PHP Data Objects)

PDO (PHP Data Objects), adalah extension atau penambahan fitur dalam PHP yang dirancang sebagai interface universal untuk pengaksesan berbagai jenis database (tidak hanya MySQL). Contohnya, jika kita menggunakan PDO dalam menulis kode pemrograman, lalu suatu saat website kita bertukar database dari MySQL ke Oracle, maka kita tidak perlu mengubah semua kode program, cukup mengubah cara pemanggilan PDO diawal program saja.

Dari ketiga cara koneksi PHP dengan MySQL ini, metode yang disarankan adalah menggunakan mysqli atau PDO. Berikut perbandingan ketiga cara tersebut:

Comparison of MySQL API options for PHP			
	PHP's mysqli Extension	PDO (Using PDO MySQL Driver and MySQL Native Driver)	PHP's MySQL Extension
PHP version introduced	5.0	5.0	Prior to 3.0
Included with PHP 5.x	yes	yes	Yes
MySQL development status	Active development	Active development as of PHP 5.3	Maintenance only
Recommended by MySQL for new projects	Yes - preferred option	Yes	No
API supports Charsets	Yes	Yes	No
API supports server-side Prepared Statements	Yes	Yes	No
API supports client-side Prepared Statements	No	Yes	No
API supports Stored Procedures	Yes	Yes	No
API supports Multiple Statements	Yes	Most	No
Supports all MySQL 4.1+ functionality	Yes	Most	No

Perbandingan cara koneksi MySQL dengan menggunakan PHP

Membuat Koneksi dengan PHP dan MySQL

MySQLi adalah sebuah fungsi baru dari MySQL. MySQLi di informasikan lebih aman digunakan. dan lebih cepat dari pada MySQL. MySQLi di ciptakan untuk di jalankan pada versi PHP 5 ke atas (PHP 7).

Berikut baris kode yang digunakan untuk membuat koneksi, dan simpan dalam nama: **config.php**

```

1 $koneksi = mysqli_connect("localhost", "username_db", "password_db", "nama_db");
2 // Check connection
3 if (mysqli_connect_errno()){
4     echo "Koneksi database gagal : " . mysqli_connect_error();
5 }
6 ?>

```

Ket:

Username_db : merupakan username pada basisdata, pada xampp username default adalah *root*
 Password_db : merupakan password basisdata, pada xampp pass default nya adalah *blank (tidak ada password)* jadi kita bisa mengosongkan menjadi "".

```

1 $koneksi = mysqli_connect("localhost","root","","akademik");
2 // Check connection
3 if (mysqli_connect_errno()){
4     echo "Koneksi database gagal : " . mysqli_connect_error();
5 }
6 ?>

```

Keterangan:

Pada parameter pertama, kita mengetikkan nama host kita. yaitu **localhost**. dan pada parameter kedua kita mengetikkan username dari mysql kita, defaultnya adalah **root**. parameter ketiga kita ketikkan password mysql kita (jika ada), karena default nya **kosong**. Dan terakhir pada parameter ke empat kita mengetik nama database yang ingin kita gunakan. atau database yang ingin kita hubungkan ke php. isikan sesuai dengan nama database yang ingin digunakan.Fungsi **mysqli_connect_error()** berguna untuk memeriksa koneksi apakah berhasil atau tidak, jika gagal maka akan kita jalankan fungsi berikutnya.

Create Data atau Input

Untuk mendapatkan inputan dari pengguna dan memasukan data nya ke basis data, maka dibutuhkan sebuah form yang menjadi interface bagi pengguna. Berikut baris kode untuk sebuah form, dan simpan dengan nama **input.php**:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <title>Bootstrap Example</title>
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7     <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
8 </head>
9 <body>
10
11    <div class="container">
12        <h1>Input Data Siswa</h1>
13        <p>Isi Form dibawah ini untuk menginput data siswa</p>
14        <hr>
15        <form action="proses_input.php" method="POST">
16            <div class="form-group">
17                <label for="email">Nama :</label>
18                <input type="text" class="form-control" name="nama">
19                <label for="alamat">Alamat :</label>
20                <input type="text" class="form-control" name="alamat">
21                <label for="no_hp">No Handphone:</label>
22                <input type="text" class="form-control" name="no_hp">
23            </div>
24            <button type="submit" class="btn btn-primary">Submit</button>
25        </form>
26    </div>
27
28 </body>
29 </html>

```

Hasil output nya sebagai berikut:

Input Data Siswa

Isi Form dibawah ini untuk menginput data siswa

Nama :

Alamat :

No Handphone:

Submit

action mengarahkan ke file **proses_input.php**, maka kita akan membuat **proses_input.php** dengan kode sebagai berikut:

```
1 <?php
2 include 'config.php';
3
4 $nama = $_POST['nama'];
5 $alamat = $_POST['alamat'];
6 $no_hp = $_POST['no_hp'];
7
8 $sql = "INSERT INTO `tb_data_siswa` (`id_siswa`, `nama_siswa`, `alamat_siswa`, `no_telp`) VALUES
9      (NULL, '$nama', '$alamat', '$no_hp'); ";
10
11 $result = $koneksi->query($sql) or die("Cannot write");
12
13 if($result){
14 echo '<script language="javascript">alert("Input berhasil ditambahkan.");
15     document.location="list_siswa.php";</script>';
16 }
17 else {
18
19 echo '<script language="javascript">alert("Input gagal ditambahkan.");
20     document.location="list_siswa.php";</script>';
21 }
22 ?>
```

Untuk dapat menyimpan data ke basisdata, maka kita perlu membuat sebuah file yang dapat melakukan hal tersebut. Pada kode form diatas, atribut

Keterangan:

- include "koneksi.php"; → berfungsi untuk memanggil atau menyertakan file koneksi.
- \$sql = berfungsi untuk memasukan perintah query mysql agar nantinya dapat dieksekusi.
- \$result = berfungsi untuk melakukan pengecekan apakah query tersebut dapat di eksekusi atau tidak. Jika berhasil di eksekusi maka akan memberikan nilai TRUE.
- If(\$result) = berfungsi untuk memberikan notifikasi kepada pengguna, apabila query berhasil di eksekusi maka tampil pesan dan mendirect ke halaman list siswa.

Read Data (Menampilkan data dari basisdata)

Untuk dapat menampilkan data yang telah berhasil di inputkan, maka buatlah sebuah file php dengan nama “list_siswa.php” dan isi lah dengan baris berikut:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
8 </head>
9 <body>
10  <div class="container">
11    <h1>Data Siswa</h1>
12    <p>List Data Siswa</p>
13    <p><a href="input.php"> + Tambah Data Siswa </a> </p>
14    <table class="table">
15      <tr>
16        <th>No</th>
17        <th>Nama</th>
18        <th>Alamat</th>
19        <th>No Hp</th>
20        <th>Opsi</th>
21      </tr>
22      <?php
23      include "config.php";
24      $koneksi->query("SELECT * FROM tb_data_siswa");
25      while($data = mysqli_fetch_assoc($sql)){
26          echo '<tr>';
27          echo '<td>'.$data['id_siswa'].'</td>';
28          echo '<td>'.$data['nama_siswa'].'</td>';
29          echo '<td>'.$data['alamat_siswa'].'</td>';
30          echo '<td>'.$data['no_telp'].'</td>';
31          echo '<td>';
32          echo '<a class="edit" href="edit.php?id='.$data['id_siswa'].'">Edit</a>
33          <a class="hapus" href="hapus.php?id='.$data['id_siswa'].'">Hapus</a>
34          ';
35          echo '</td>';
36          echo '</tr>';
37      }
38  ?>
39  </table>
40 </div>
41 </body>
42 </html>
```

Penjelasan:

- Proses while berfungsi untuk memecah data menjadi array dan memasukannya kedalam variabel data agar data bisa kita tampilkan dalam bentuk perulangan (\$data = mysqli_fetch_assoc(\$sql))
- Pada bagian echo '<td>'.\$data['id_siswa'].'</td>'; Variabel data sudah menjadi array dan masukan nama field pada table di di tb_data_siswa yang ingin ditampilkan seperti contoh diatas

Update Data

Pada file **list_siswa.php** terdapat baris di table untuk melakukan edit yaitu :

```
<a class="edit" href="edit.php?id='.$data['id_siswa'].'">Edit</a> .
```

Bagian kunci pada update adalah mengirimkan primary key (id_siswa) agar dapat dilakukan update data pada data tersebut. Id Siswa tersebut dikirimkan pada form update yang akan dibuat. Buatlah sebuah file untuk edit data dengan nama file php "**edit.php**" dan isi lah dengan baris berikut:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
8 </head>
9 <body>
10 <?php
11 include 'koneksi.php';
12 $id_siswa = $_GET['id'];
13 $sql = $koneksi→query("SELECT * FROM tb_data_siswa WHERE id_siswa = '$id_siswa' ");
14 $data = mysqli_fetch_assoc($sql);
15 $nama_siswa = $data['nama_siswa'];
16 $alamat_siswa = $data['alamat_siswa'];
17 $no_telp = $data['no_telp'];
18 ?>
19   <div class="container">
20     <h1>Update Data Siswa</h1>
21     <p>Lakukan perubahan data</p>
22     <hr>
23     <form action="proses_update.php" method="POST">
24       <input type="hidden" id="id_siswa" name="id_siswa" value=<?php echo $id_siswa; ?>">
25       <div class="form-group">
26         <label for="email">Nama :</label>
27         <input type="text" class="form-control" name="nama" value=<?php echo $nama_siswa; ?>">
28         <label for="alamat">Alamat :</label>
29         <input type="text" class="form-control" name="alamat" value=<?php echo $alamat_siswa; ?>">
30         <label for="no_hp">No Handphone:</label>
31         <input type="text" class="form-control" name="no_hp" value=<?php echo $no_telp; ?>">
32       </div>
33       <button type="submit" class="btn btn-primary">Submit</button>
34     </form>
35   </div>
36 </body>
37 </html>
```

Dan akan menghasilkan output sebagai berikut:

Update Data Siswa

Lakukan perubahan data

Nama :

Supardianto

Alamat :

Batam

No Handphone:

085762315262

Submit

Penjelasan:

- Karena link edit pada halaman list siswa menghasil link seperti ini `edit.php?id=1` , maka dibutuhkan fungsi GET untuk mengambil nilai id yaitu pada perintah berikut: `$id_siswa = $_GET['id'];`
- \$sql dan \$data berfungsi untuk melakuka query terhadap data pada basis data untuk mendapat hasil data yang sesuai dengan id siswa yang akan di update.
- Data yang sudah berhasil di query ditampung dalam variabel nama_siswa melalui perintah berikut:
`$nama_siswa = $data['nama_siswa'];`
- Variabel yang sudah memiliki value data tersebut kemudian akan di isi pada input textbox dengan memasukan variabel tersebut pada value di input seperti `value="<?php echo $nama_siswa; ?>"` , pengguna tag php pada value dikarenakan php di sisipkan pada tag html, karena kita membutuhkan php untuk menampil varibel \$nama_siswa maka kita membutuhkan tag PHP.
- Agar pada proses update nanti bisa dilakukan, maka kita perlu mengirimkan juga nilai id_siswa melalui perintah berikut: `<input type="hidden" id="id_siswa" name="id_siswa" value="<?php echo $id_siswa; ?>">`
Type hidden mengartikan bahwa input tersebut tidak ditampilkan, namun tetap disertakan untuk dikirim pada form update.

Untuk memproses form update, maka diperlukan sebuah file untuk melakukan proses update, berikut baris kode program, buat dengan **proses_update.php**

```
1 <?php
2 include 'config.php';
3
4 $id_siswa = $_POST['id_siswa'];
5 $nama = $_POST['nama'];
6 $alamat = $_POST['alamat'];
7 $no_hp = $_POST['no_hp'];
8
9 $sql = "UPDATE `tb_data_siswa` SET
10 `nama_siswa` = '$nama',
11 `alamat_siswa` = '$alamat',
12 `no_telp` = '$no_hp'
13 WHERE `tb_data_siswa`.`id_siswa` = '$id_siswa'";
14
15 $result = $koneksi→query($sql) or die("Cannot write");
16
17 if($result){
18 echo '<script language="javascript">alert("Update berhasil dilakukan.");
19 document.location="list_siswa.php";</script>';
20 }
21 else {
22 echo '<script language="javascript">alert("Update gagal dilakukan.");
23 document.location="list_siswa.php";</script>';
24 }
25
26 ?>
```

Menghapus Data (Delete)

Pada file **list_siswa.php** terdapat baris di table untuk melakukan delete yaitu

```
"<a class="edit" href="hapus.php?id='".$data['id_siswa']."'>Delete</a>"
```

Bagian kunci pada hapus adalah mengirimkan primary key (`id_siswa`) agar dapat dilakukan hapus data pada data tersebut. Id Siswa tersebut dikirimkan pada file php yang memproses untuk menghapus data. Buatlah file PHP dengan nama file "**hapus.php**", dan isi dengan baris berikut:

```
1 <?php
2 include 'config.php';
3
4 $id = $_GET['id'];
5 $sql = "DELETE FROM tb_data_siswa WHERE id_siswa ='$id' ";
6 $result = $koneksi→query($sql) or die("Cannot write");
7
8 if($result){
9 echo '<script language="javascript">alert("Hapus data berhasil dilakukan.");
10 document.location="list_siswa.php";</script>';
11 }
12 else {
13 echo '<script language="javascript">alert("Hapus data gagal dilakukan.");
14 document.location="list_siswa.php";</script>';
15 }
16 ?>
```

Penjelasan:

- Untuk menentukan id siswa yang akan dihapus maka perlu untuk mengambil nilai yang telah dikirim sebelumnya yaitu dengan menggunakan method GET melalui perintah: `$id = $_GET['id'];`
- `$sql` dan `$result` berfungsi untuk mengeksekusi query untuk menghapus.

Mencari data siswa (SEARCH)

Buatlah sebuah file PHP dengan nama “cari.php” dan isi lah dengan baris berikut:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Bootstrap Example</title>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
8 </head>
9 <body>
10  <div class="container">
11    <h1>Cari Data Siswa</h1>
12    <p>Isi nama siswa yang ingin dicari</p>
13    <hr>
14    <form action="" method="POST">
15      <div class="form-group">
16        <label for="email">Nama :</label>
17        <input type="text" class="form-control" name="nama">
18      </div>
19      <button type="submit" class="btn btn-primary">Search</button>
20    </form>
21  </div>
22  <br>
23  <?php
24  include 'config.php';
25  if(isset($_POST['nama'])){
26    $nama = $_POST['nama'];
27    $sql = $koneksi→query("SELECT * FROM tb_data_siswa WHERE nama_siswa = '$nama' ");
28    $data = mysqli_fetch_assoc($sql);
29
30    $nama_siswa = $data['nama_siswa'];
31    $alamat_siswa = $data['alamat_siswa'];
32    $no_telp = $data['no_telp'];
33
```

```

34 echo '<div class="container">';
35 echo '<p>Hasil Pencarian Nama Siswa: '.$nama_siswa.'</p>';
36
37 if(isset($data)){
38 echo '
39 <table class="table">
40     <tr>
41         <th>Nama</th>
42         <th>Alamat</th>
43         <th>No Hp</th>
44     </tr>
45     <tr>
46         <td>'.$nama_siswa.'</td>
47         <td>'.$alamat_siswa.'</td>
48         <td>'.$no_telp.'</td>
49     </tr>
50 </table>
51 </div>
52 ';
53 }
54 else
55 {
56     echo "Nama siswa tidak ditemukan";
57 }
58 }
59 ?>
60 </body>
61 </html>
62

```

Penjelasan:

- Pada tag **<form>** action tidak melakukan pengiriman data ke file lain, tetapi ke diri sendiri.
- Pada tag **<?php ?>** merupakan proses yang dilakukan untuk melakukan pencarian data. **\$sql** dan **\$data** melakukan query untuk mencari data berdasar **\$nama_siswa** yang telah di dapat melalui POST.
- **if(isset(\$data)){** berguna untuk melakukan pengecekan apabila hasil pada query **\$data** terdapat nilai maka tampilkan data dalam bentuk tabel, Jika tidak maka tampil pesa bahwa data yang dicari tidak ada.

XIV ENKRIPSI, AUTHENTIFIKASI, SESSION DAN COOKIES

Enkripsi adalah suatu metode yang digunakan untuk mengkodekan data sedemikian rupa sehingga keamanan informasinya terjaga dan tidak dapat dibaca tanpa di dekripsi (kebalikan dari proses enkripsi) dahulu. Encryption berasal dari bahasa Yunani krytos yang artinya tersembunyi atau rahasia.

6 Fungsi Enkripsi di PHP untuk Mengamankan Data

Enkripsi adalah teknik untuk mengamankan data-data tersebut agar isinya tidak diketahui orang lain. Enkripsi biasanya dilakukan terhadap data-data sensitif seperti password. Enkripsi akan menjamin data-data tetap aman meskipun berada di tangan orang lain, karena tidak tahu isi aslinya. Beberapa fungsi enkripsi yang sudah disediakan oleh PHP, diantaranya: *password_hash()*, *crypt()*, *md5()*, *hash()*, *sha1()*, dan *base64_encode()*.

1. Fungsi *password_hash()*

Fungsi ini akan menghasilkan sebuah kode hash baru dengan metode one-way hashing. one-way hashing artinya, hasil enkripsinya tidak bisa dikembalikan seperti semula (decrypt/decode). **Password_hash** adalah salah satu fungsi yang dimiliki PHP untuk melakukan hashing menggunakan algoritma satu arah (one-way hashing). Fungsi ini terdapat pada PHP versi 5.5 ke atas.

Sedangkan **Hash** sendiri adalah suatu mekanisme untuk memetakan suatu data yang diinputkan berupa teks polos (plain text) menjadi output berupa checksum atau fingerprint (kode acak dengan panjang karakter yang tetap). Karena **password_hash** menggunakan algoritma hashing satu arah, maka data yang sudah terhash tidak akan bisa dikembalikan lagi menjadi data aslinya. Berikut contoh penggunaannya:

```
1 <?php echo password_hash("Polibatam", PASSWORD_DEFAULT); ?>
1 $2y$10$7PZGQcOAK6ZJXCA03UFRC.hpv8hjKsQhy5Y7xB0hLNoFNTuxF3lrz2
```

Fungsi ini sangat disarankan untuk mengenkripsi password, karena sulit didekripsi atau di-crack. Sebelum mengenal fungsi ini, biasanya menggunakan fungsi md5(). Fungsi **password_hash()** ini tidak bisa bekerja sendirian, dia memiliki teman bernama **password_verify()**. Fungsi **password_verify()**, biasanya digunakan saat membuat login untuk membandingkan password yang dikirim user dengan password yang tersimpan pada database. Contoh:

```
1 <?php
2 // ambil data dari form login
3 $username = $_POST['username'];
4
5 // ambil data dari database
6 $query = mysqli_query("SELECT * FROM user WHERE username='$username'");
7 $user = mysqli_fetch_assoc($query);
8
9 // bandingkan password yang dikirim dari form login dengan password
10 // yang ada di database
11 if( password_verify($_POST['password'], $user['password']) ) {
12     // login berhasil
13 } else {
14     // login gagal
15 }
16 ?>
```

2. Fungsi *crypt()*

Crypt merupakan sebuah fungsi yang terdapat pada PHP yang bisa kita gunakan untuk enkripsi atau keamanan, pada crypt menghasilkan kode atau kata sandi yang lebih pendek dari fungsi **password_hash()**. Pada crypt menghasilkan hash yang berbasis pada algoritma DES (Data Encryption

Standard), string yang di enkripsi dengan menggunakan fungsi crypt akan menghasilkan chipertext yang dibuat secara acak. Berikut contoh penggunaan nya:

```
1 <?php
2 $password = 'qwerty';
3 $salt = 'helloworld';
4 echo crypt($password, $salt);
5 ?>
```

Pada script diatas kita perhatikan script diatas maka akan menghasilkan sebuah kode acak seperti dibawah ini. Namun kode yang dihasilkan lebih pendek dibandingkan dengan dari fungsi **password_hash()**.

```
1 hay0ScmL8GDLA
```

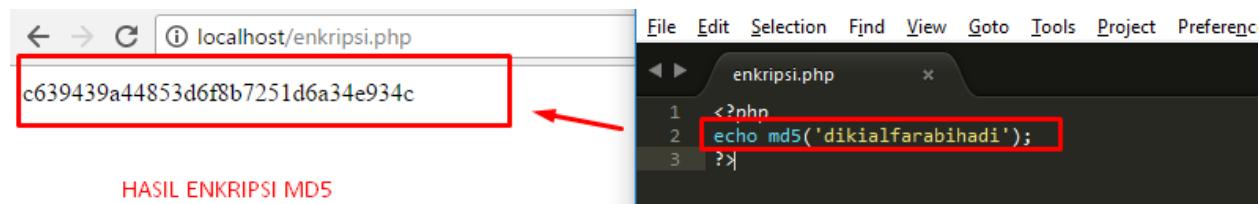
Pada contoh diatas, kita memberikan dua parameter kepada fungsinya. **Parameter pertama** adalah teks yang akan dienkripsi. Sementara, **parameter kedua** adalah salt (garam). Garam adalah sebuah data acak yang dimasukkan ke dalam fungsi enkripsi. Fungsi ini juga dapat digunakan untuk mengamankan password, karena sulit di-decode.

3. Fungsi md5()

MD5 adalah singkatan dari “Message-Direst Algorithm 5”. dan merupakan sebuah algoritma fungsi hash dari kriptografi yang di gunakan dengan hash value 128-bit. Fungsi ini akan menghasilkan kode hash sepanjang 32 karakter. Contoh:

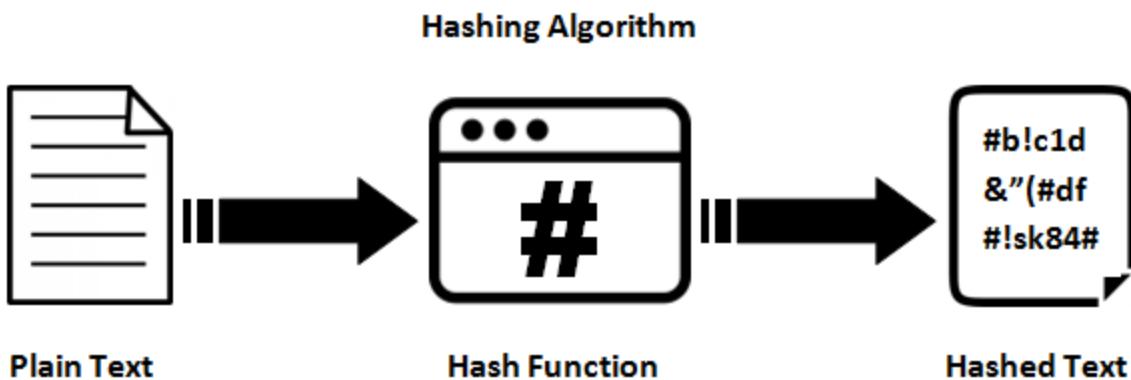
```
1 <?php
2 echo md5('dikialfarabihadi');
3 ?>
```

Hasilnya:



4. Fungsi hash()

Fungsi ini akan menciptakan sebuah kode hash dengan algoritma tertentu.



Contoh:

```
1 <?php echo hash("md5", "polibatam"); ?>
```

Pada contoh di atas, parameter "md5" adalah nama algortima yang akan digunakan untuk mengenkripsi. Parameter "polibatam" adalah teks yang akan dienkripsi.

5. Fungsi sha1()

Fungsi ini akan menghasilkan kode hash sepanjang 40 karakter. Mirip seperti fungsi md5(). **Secure Hash Algorithm (SHA)** dikembangkan oleh NIST (National Institute of Standards and Technology) dan digunakan bersama DSS (Digital Signature Standard). SHA-1 adalah revisi terhadap SHA yang dipublikasikan pada tahun 1995. SHA-1 disebut aman karena dirancang sedemikian rupa sehingga secara komputasi tidak mungkin menemukan pesan yang berkoresponden dengan message digest yang diberikan. SHA1 memiliki panjang 20 bytes atau 40 karakter, contohnya : 356a192b7913b04c54574d18c28d46e6395428ab.

Contoh penggunaannya:

```
1 <?php echo sha1("polibatam"); ?>
```

6. Fungsi base64_encode()

Fungsi ini akan menghasilkan kode hash dari teks yang diinputkan dan bisa dikembalikan ke bentuk semula dengan fungsi **base64_decode()**. Metode ini disebut two-way hashing.

Contoh:

```
1 <?php echo base64_encode("Polibatam"); ?>
```

Sementara itu, untuk mengembalikan (decrypt) atau decode dapat menggunakan fungsi **base64_decode()**. Contoh:

```
1 <?php echo base64_decode("UGV0YW5pIGtvZGU="); ?>
```

Session dan Cookies

Di dalam suatu website biasanya user hanya memasukkan password atau login sekali saja, akan tetapi mampu masuk ke halaman lain tanpa login kembali. Informasi atau data login dari user tersebutlah mampu disimpan dan juga digunakan kembali sehingga mampu untuk memaksimalkan kinerja dan efektifitas suatu website. Kemampuan untuk mengingat suatu informasi tersebut dapat Anda sesuaikan waktunya sesuai dengan keinginan programmer/ pemilik website tersebut. Fasilitas dari session dan cookie ini juga dapat untuk meningkatkan keamanan dari website itu sendiri, karena waktu pengaksesan user atau client dapat

dibatasi, sehingga jika waktu session habis, user atau client bisa memperpanjang waktu session ataupun mangakhiri waktu session tersebut.

Cookies

Cookies atau bisa isebut HTTP cookies, web cookies, atau browser cookies adalah data atau informasi yang diciptakan oleh suatu website untuk disimpan di web browser, ketika user sedang menjelajahi website tersebut. Cookies yang telah disimpan pada browser bisa iambil untuk memberitahu website tentang aktifitas user pada waktu sebelumnya. Aktivitas tersebut misalnya mengklik suatu tombol, login, atau halaman mana yang telah dibuka user pada bulan lalu bahkan tahun lalu. Berdasarkan sifat yang telah diuraikan diatas.

Fungsi cookies:

1. Menyimpan username dan password login agar pengguna tidak selalu harus mengisikannya pada saat membuka page (halaman web). Biasanya akan muncul chechbox "Remember Me" atau "Ingat Saya Selalu" yang terdapat pada bagian bawah box login.
2. Merekam daftar barang yang ingin dibeli pengunjung
3. Untuk mencatat konfigurasi yang dilakukan oleh user. Seperti: warna tema, jenis huruf, pilihan bahasa, dll.
4. Membuat suatu sesi yang akan memungkinkan seseorang dapat langsung masuk ke halaman-halaman lain tanpa harus melakukan login kembali.
5. Untuk mengetahui apakah pengunjung pernah datang atau belum ke halaman yang sedang dibuka (seperti yang digunakan pada hit counter /penghitung jumlah pengunjung pada blog).

Cookies mempunyai umur, artinya adalah setiap data yang Anda simpan dalam komputer user suatu saat bisa hilang atau musnah. Ini disebabkan karena bahwa cookie itu mempunyai umur atau masa disimpan dalam sebuah komputer/ perangkat dari user.

Jenis-Jenis Cookies

1. Non persistent (session) cookies

Suatu cookie yang mana akan hilang jika sewaktu user menutup browser mereka dan biasanya juga digunakan pada suatu 'shopping carts' di toko online shop/ toko belanja online untuk menelusuri item-item yang akan dibeli,

2. Persistent cookies

Diatur oleh suatu situs-situs portal, banner/ media iklan situs dan lain sebagainya yang ingin tahu ketika user kembali mengunjungi site mereka. (contohnya saja dengan cara memberikan opsi "Remember Me" saat melakukan login). File-file ini akan tersimpan di hardisk/ media penyimpanan user.

Dari kedua tipe cookies ini menyimpan informasi tentang suatu URL atau domain name dari sebuah situs yang dikunjungi oleh user dan juga beberapa kode yang mengindikasikan halaman apa saja yang sudah dikunjungi oleh user.

Cookies juga dapat berisi suatu informasi pribadi dari data user, contohnya seperti nama dan alamat email, namun dapat juga user memberikan suatu informasi ke website tersebut melalui proses registrasi/ pendaftaran. Dengan kata lain, cookies ini tidak akan dapat 'mencuri' nama dan alamat email kecuali diberikan oleh user itu sendiri. Akan tetapi, ada suatu kode tertentu (malicious code) yang dibuat misalnya saja dengan ActiveX control, yang dapat mengambil informasi dari PC/ perangkat tanpa sepengetahuan dari user.

Cookies itu umumnya kurang dari 100 bytes, sehingga tidak akan mudah untuk mempengaruhi kecepatan browsing. Akan tetapi tetapi karena umumnya browser itu diatur dengan cara default untuk menerima suatu

cookies maka user/ pengguna tidak akan tahu bahwa cookies sudah ada di komputer yang sudah terinstal otomatis.

Cookies juga tentunya dapat berguna terutama untuk suatu situs yang memerlukan registrasi/ pendaftaran di dalamnya, sehingga setiap kali mengunjungi situs tersebut, cookies akan me-login-kan user tanpa harus memasukkan user name dan password lagi. Suatu server hanya bisa menyimpan kurang lebih 20 cookies saja pada browser user/ pemakai, masing-masing cookies berukuran terbatas hanya sampai sebesar 4 KB saja.

Cara Mendeklarasikan Cookies Dalam PHP

Suatu cookies haruslah dideklarasikan sebelum halaman ditampilkan di browser, yang artinya dituliskan sebelum tag <+html>. Secara mendasar untuk dapat membuat suatu cookies ditulis dengan susunan, sebagai berikut ini:

```
1 <?php
2 $nilai = 'Martabak Mesir';
3 // Cara Pertama
4 setcookie("Enak", $nilai);
5 // Cara Kedua
6 setcookie("Enak", $nilai, time() + 3600); /* Berlaku Satu Jam */
7 // Cara Ketiga
8 setcookie("Enak", $nilai, time() + 3600, "/Nama Folder/", "martabak.com");
9 ?>
```

Dan supaya untuk bisa mengambil nilai dari cookies tersebut dapat dilakukan dengan cara, sebagai berikut ini :

```
1 <?php
2 $enak = $_COOKIE["Enak"];
3 echo $enak;
4 ?>
```

Session

Session adalah salah satu fasilitas yang ada pada sebuah pemrograman PHP, yang digunakan untuk menyimpan/ mensave data sementara ke dalam suatu variabel (variabel session) sehingga data tadi bisa diakses oleh client selama variabel session tadi tidak dikosongkan ataupun dihilangkan. Nilai variabel dari sebuah session itu disimpan di sisi server yang bernama web server.

Berbeda halnya dengan cookies yang nilai variabelnya itu hanya disimpan di sisi client browser saja. Jadi penggunaan session relatif lebih aman digunakan untuk menyimpan suatu variabel nilai yang sifatnya rahasia seperti contohnya username dan password pada saat Anda login. Manfaat session diantaranya supaya menyimpan informasi login yang berlaku hanya dalam satu sesi saja dan menyimpan catatan order barang dalam suatu sistem e-commerce/ transaksi online.

Beberapa hal tentang session di PHP:

1. Mengawali Suatu Session

Dipergunakan untuk membuat sebuah session atau melanjutkan session sebelumnya berdasarkan pada pengidentifikasi session melalui GET atau POST atau cookie. Membuktikan bahwa suatu halaman

menggunakan session, maka pada awal halaman harus juga ada awal session yaitu dengan menggunakan session_start() atau session_start;

2. Penggunaan pada Variabel Session

Sesudah session dimulai maka variabel session sudah bisa mulai untuk digunakan. Penggunaannya menggunakan format `$_SESSION['nama_variabel']`. Setiap halaman yang akan menggunakan variable session harus mendelarasikan kode seperti di atas, supaya dapat menggunakan/ mengoprasikannya. Kemudian baru setelah itu, Anda bisa membuat nama session baru, sebagai berikut ini :

```
1 <?php  
2 $_SESSION['Try'] = "OKESIP";  
3 ?>  
1 <?php  
2 echo $_SESSION['Try'];  
3 ?>
```

Session dengan menggunakan nama 'Try' akan dapat Anda gunakan dalam sebuah halaman/ page

3. Menghapus Session

Setelah variabel session digunakan, variabel tersebut dapat dihapus. Banyak cara untuk menghapus session di antaranya sebagai berikut:

1. `$_SESSION['nama_variabel'] = "...."`

digunakan untuk memberikan atau mengganti sebuah nilai dari variabel session menjadi null atau kosong.

2. `unset($_SESSION['nama_variabel'])`

digunakan untuk menghapus sebuah variabel session.

3. `Session_destroy()`

digunakan untuk menghapus pada semua variabel session yang mungkin ada banyak variabel session yang sudah dibuat. Fungsi session destroy tidak memerlukan argumen dalam penggunaanya.

Autentifikasi User di PHP

Autentifikasi user adalah suatu mekanisme untuk memastikan apakah suatu user itu berhak masuk ke dalam sistem atau bukan. Implementasinya adalah berupa login. Dalam hal ini user yang berhak mengakses akan diberikan nama user tertentu beserta password. Mekanisme autentifikasi user dimulai dari berikut:

Pendaftaran User

1. User mengisi form pendaftaran sebagai legal user (user mengisi username dan password)
2. Sistem akan mengecek apakah username yang didaftarkan sudah ada yang memiliki atau belum
3. Jika sudah ada, user diminta mengisi kembali username yang lain beserta passwordnya. Sedangkan jika belum ada, data user ini akan disimpan dalam database, dengan password terenkripsi.

Proses Autentifikasi

1. User yang akan mengakses sistem diminta memasukkan username dan password (asli)
2. Sistem akan mencari password terenkripsi yang tersimpan dalam database berdasarkan username yang terdaftar
3. Sistem akan mencocokkan antara password asli terenkripsi yang diisikan user melalui form login, dengan password terenkripsi yang tersimpan dalam database

4. Jika password asli terenkripsi yang dikirim via form login ini sama dengan password terenkripsi yang ada dalam database, maka user tadi bisa masuk ke dalam sistem. Jika tidak sama, maka user tadi tidak berhak masuk ke sistem.

Membuat Register dan Login pada PHP

Register dan login akan digunakan sebagai percobaan dengan memanfaatkan enkripsi dan proses autentifikasi serta memanfaatkan session sebagai validasi halaman.

Percobaan tersedia pada modul praktikum.

XV TABEL DINAMIS DENGAN DATATABLES

Tabel (Table) adalah sebuah data tabular dalam bentuk grid yang terdiri dari kolom (column), baris (row) dan cell yang merupakan pertemuan antara kolom dan baris. HTML <table> element merepresentasikan data dengan lebih dari satu dimensi dalam bentuk sebuah tabel (table).

Table di dalam HTML dibentuk dengan sebuah baris (row), tepatnya menggunakan element <tr> yang merupakan kepanjangan dari table row. Adapun element pendukung lain yang membentuk sebuah table adalah <th> (table header) dan <td> (table data). <td> menunjukkan sebuah cell, sedangkan <th> menunjukkan cell induk dan ditandai dengan tulisan cetak tebal.

Contoh sebuah tabel sederhana:

```
1 <table>
2   <tr>
3     <td>No.</td>
4     <td>Nama</td>
5     <td>Tempat Lahir</td>
6   </tr>
7   <tr>
8     <td>1.</td>
9     <td>Rodiah</td>
10    <td>Serang</td>
11   </tr>
12 </table>
```

No.	Nama	Tempat Lahir
1.	Rodiah	Serang

Saat ini untuk menampilkan data pada sebuah tabel tidak cukup hanya dengan menampilkan saja, namun keperluan terhadap data pada sebuah tabel melengkapi:

1. Kemudahan mencari data langsung pada tabel
2. Fungsi filtering data
3. Sorting data dengan cepat terhadap semua field kolom
4. Dan export data pada tabel dalam bentuk xls, pdf atau yang lain.

Plugin datatables adalah plugin yang di bangun dari jquery untuk menampilkan data dalam bentuk table dan sudah terintegrasi search form/form pencarian dan pagination untuk penomoran data pada table. sehingga kita tidak perlu lagi membuat form pencarian pada data. dan juga kita tidak perlu lagi membuat pagination secara manual. karena kelebihan dari datatables ini adalah dia telah mengintegrasikan pagination dan form pencarian pada table data kita.

Tentu saja akan sangat menghemat waktu kita jika kita menggunakan plugin jquery datatables. kita tidak perlu lagi repot-repot membuat search form dan pagination pada table. kita tinggal menghubungkan file dengan plugin datatables dan kemudian memanggilnya dengan fungs datatables.

Instalasi Datatables

Tahap awal yang harus kita lakukan untuk mulai menggunakan plugin datatables adalah download dulu plugin datatables di situs resminya (<https://www.datatables.net/>). Karena plugin jquery datatables dibangun menggunakan jquery maka kita juga memerlukan library jquery. Untuk bisa menggunakan plugin datatables kita harus menghubungkan dulu dengan file library jquery dan plugin datatables nya. Ada 2 file yang nanti akan letakan pada direktori project kita, yaitu:

1. jquery.dataTables.min.css → letakan pada direktori khusus css
2. jquery.dataTables.min.js → letakan pada direktori khusus js

Disk (C:) > xampp > htdocs > BS4_crud > bootstrap > css			
Name	Date modified	Type	Size
bootstrap.css	28/11/2019 19:38	Cascading Style S...	193 KB
bootstrap.css.map	28/11/2019 19:38	MAP File	493 KB
bootstrap.min.css	28/11/2019 19:38	Cascading Style S...	156 KB
bootstrap.min.css.map	28/11/2019 19:38	MAP File	627 KB
bootstrap-grid.css	28/11/2019 19:38	Cascading Style S...	67 KB
bootstrap-grid.css.map	28/11/2019 19:38	MAP File	155 KB
bootstrap-grid.min.css	28/11/2019 19:38	Cascading Style S...	50 KB
bootstrap-grid.min.css.map	28/11/2019 19:38	MAP File	113 KB
bootstrap-reboot.css	28/11/2019 19:38	Cascading Style S...	5 KB
bootstrap-reboot.css.map	28/11/2019 19:38	MAP File	76 KB
bootstrap-reboot.min.css	28/11/2019 19:38	Cascading Style S...	4 KB
bootstrap-reboot.min.css.map	28/11/2019 19:38	MAP File	32 KB
jquery.dataTables.min.css	07/04/2020 14:23	Cascading Style S...	14 KB

Disk (C:) > xampp > htdocs > BS4_crud > bootstrap > js			
Name	Date modified	Type	Size
bootstrap.bundle.js	28/11/2019 19:38	JavaScript File	223 KB
bootstrap.bundle.js.map	28/11/2019 19:38	MAP File	401 KB
bootstrap.bundle.min.js	28/11/2019 19:38	JavaScript File	79 KB
bootstrap.bundle.min.js.map	28/11/2019 19:38	MAP File	311 KB
bootstrap.js	28/11/2019 19:38	JavaScript File	132 KB
bootstrap.js.map	28/11/2019 19:38	MAP File	251 KB
bootstrap.min.js	28/11/2019 19:38	JavaScript File	59 KB
bootstrap.min.js.map	28/11/2019 19:38	MAP File	190 KB
jquery.dataTables.min.js	07/04/2020 14:23	JavaScript File	83 KB
jquery.js	11/02/2020 14:08	JavaScript File	87 KB

Setelah kedua file tadi berada pada direktori project, maka langkah selanjut nya adalah menambahkan script jquery untuk mengakses datatables. Contoh script nya:

```
1 $(document).ready( function () {  
2     $('#myTable').DataTable();  
3 } );
```

Script tersebut akan menjalankan fungsi dari datatables dan akan dikenakan terhadap sebuah tabel yang memiliki atribut id berupa '**#myTable**'. Contoh sebuah table yang sudah menggunakan datatables:

Name		Position	Office	Age	Start date
(+)	Airi Satou	Accountant	Tokyo	33	2008/11/28
(+)	Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09
(+)	Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12
(+)	Bradley Greer	Software Engineer	London	41	2012/10/13
(+)	Brenden Wagner	Software Engineer	San Francisco	28	2011/06/07
(+)	Brielle Williamson	Integration Specialist	New York	61	2012/12/02
(+)	Bruno Nash	Software Engineer	London	38	2011/05/03
(+)	Caesar Vance	Pre-Sales Support	New York	21	2011/12/12
(+)	Cara Stevens	Sales Assistant	New York	46	2011/12/06
(+)	Cedric Kelly	Senior Javascript Developer	Edinburgh	22	2012/03/29

Name	Position	Office	Age	Start date				
Showing 1 to 10 of 57 entries	Previous	1	2	3	4	5	6	Next

Menggunakan Datatables

Persiapan

Pada materi CRUD pada PHP dan MySQL, kita telah membuat sebuah file untuk menampilkan data dalam bentuk tabel, namun belum menggunakan datatables. Berikut baris kode dengan nama file **list_siswa.php**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Bootstrap Example</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
</head>
<body>
    <div class="container">
        <h1>Data Siswa</h1>
        <p>List Data Siswa</p>
        <p><a href="input.php"> + Tambah Data Siswa </a> </p>
        <table class="table">
            <tr>
                <th>No</th>
                <th>Nama</th>
                <th>Alamat</th>
                <th>No Hp</th>
                <th>Opsi</th>
            </tr>
            <?php
                include "koneksi.php";
                $sql = $conn->query("SELECT * FROM tb_data_siswa");
                while($data = mysqli_fetch_assoc($sql)){
                    echo '<tr>';
                    echo '<td>'.$data['id_siswa'].'</td>';
                    echo '<td>'.$data['nama_siswa'].'</td>';
                    echo '<td>'.$data['alamat_siswa'].'</td>';
                    echo '<td>'.$data['no_telp'].'</td>';
                    echo '<td>';
                    echo '<a class="edit" href="edit.php?id='.$data['id_siswa'].'">Edit</a>
                    <a class="hapus" href="hapus.php?id='.$data['id_siswa'].'">Hapus</a>
                    ';
                    echo '</td>';
                    echo '</tr>';
                }
            ?>
        </table>
    </div>
</body>
</html>
```

Agar tables tersebut dapat menggunakan datatables, maka kita harus menambahkan 2 file yang sudah disebutkan sebelumnya, sehingga menjadi:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <title>Bootstrap Example</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
    <script type="text/javascript" src="bootstrap/js/jquery.js"></script>

    <script type="text/javascript" src="bootstrap/js/jquery.dataTables.min.js"></script>
    <link rel="stylesheet" type="text/css" href="bootstrap/css/jquery.dataTables.min.css">

    <script type="text/javascript">
        $(document).ready( function () {
            $('#myTable').DataTable();
        } );
    </script>

</head>
<body>
    <div class="container">
        <h1>Data Siswa</h1>
        <p>List Data Siswa</p>
        <p><a href="input.php"> + Tambah Data Siswa </a> </p>

        <table id="myTable" class="table">
            <thead>
                <tr>
                    <th>No</th>
                    <th>Nama</th>
                    <th>Alamat</th>
                    <th>No Hp</th>
                    <th>Opsi</th>
                </tr>
            </thead>
            <tbody>
                <?php
                include "config.php";
                $sql = $koneksi->query("SELECT * FROM tb_data_siswa");
                while($data = mysqli_fetch_assoc($sql)){
                    echo '<tr>';
                    echo '<td>'.$data['id_siswa'].'</td>';
                    echo '<td>'.$data['nama_siswa'].'</td>';
                    echo '<td>'.$data['alamat_siswa'].'</td>';
                    echo '<td>'.$data['no_telp'].'</td>';
                    echo '<td>';
                        <a class="edit" href="edit.php?id='.$data['id_siswa'].'">Edit</a>
                        <a class="hapus" href="hapus.php?id='.$data['id_siswa'].'">Hapus</a>
                    '>;
                    echo '</td>';
                    echo '</tr>';
                }
                ?>
            </tbody>
        </table>
    </div>
</body>
</html>

```

Sehingga output nya akan menjadi seperti berikut:

Data Siswa

List Data Siswa

+ Tambah Data Siswa

Show 10 entries

Search:

No	Nama	Alamat	No Hp	Opsi
1	Toni	Batam Center	082123456	Edit Hapus

Showing 1 to 1 of 1 entries

Previous 1 Next

Keterangan:

1. Yang diberi warna kuning, adalah penambahan terhadap baris program sebelumnya.
2. Menambahkan 2 buah file datatables.
3. Menambahkan script jquery untuk mengaktifkan datatables
4. Menambahkan id="myTable"
5. Menambahkan atribut <thead> dan <tbody> pada tabel agar datatable dapat tampil.

Styling Boostrap 4 untuk Datatables

Apabila kita ingin menggunakan style Boostrap 4 maka file yang digunakan bertambah, yaitu:

1. dataTables.bootstrap4.min.css (<https://cdn.datatables.net/1.10.20/css/dataTables.bootstrap4.min.css>)
2. dataTables.bootstrap4.min.js (<https://cdn.datatables.net/1.10.20/js/dataTables.bootstrap4.min.js>)

```
<script type="text/javascript" src="bootstrap/js/dataTables.bootstrap4.min.js">
</script>

<link rel="stylesheet" type="text/css"
href="bootstrap/css/dataTables.bootstrap4.min.css">
```

Output hasil:

Data Siswa

List Data Siswa

+ Tambah Data Siswa

Show 10 entries

Search:

No	Nama	Alamat	No Hp	Opsi
1	Toni	Batam Center	082123456	Edit Hapus

Showing 1 to 1 of 1 entries

Previous 1 Next

XVI REPORT DAN UNGGAH FILE PADA PHP

Selain fungsi dari sebuah aplikasi berbasis web yaitu Create Update Delete dan Read, maka beberapa fitur yang harus hadir sebagai pelengkap aplikasi adalah aplikasi bisa menghasilkan report atau output dalam bentuk format file lainnya seperti excel, atau PDF.

Fitur lainnya yang harus ada adalah, mampu mengunggah sebuah file baik itu foto atau dokumen. Contoh sederhana adalah menampilkan foto profile. Agar pengguna dapat mengganti foto profile, maka kita harus menyediakan fitur unggah foto, atau kelengkapan data, seperti ktp atau NPWP maka kita harus memiliki fitur unggah file dokumen. Fitur-fitur seperti inilah yang dapat melengkapi aplikasi berbasis web menjadi lebih baik.

Report dengan PHP

Laporan adalah salah satu fitur yang selalu ada dalam sebuah aplikasi berbasis web, jadi skill untuk bisa menyajikan laporan adalah hal wajib yang dikuasai oleh developer, biasanya format laporan yang sering digunakan adalah dalam format PDF, Excel.

Membuat Laporan PDF dengan FPDF

Untuk membuat laporan dalam bentuk PDF maka memerlukan sebuah library. Salah satu library yang paling banyak digunakan oleh web developer adalah FPDF. FPDF merupakan suatu kumpulan class (library) PHP yang memungkinkan penggunanya dapat membuat halaman PDF dengan mudah dan tanpa perlu menginstall modul tambahan. F dari nama FPDF merupakan singkatan dari Free, jadi FPDF dapat didownload, digunakan dan dikembangkan secara bebas. Library ini punya banyak fitur seperti bisa membuat posisi text dan gambar berdasarkan koordinat, mengganti font yang akan digunakan, menambahkan garis dan masih banyak lagi.

Download dan Instalasi FPDF

Persiapan pertama yang harus dilakukan adalah mendownload library FPDF (<http://www.fpdf.org/>). Kita dapat meletakan Folder FPDF tersebut pada project kita atau pada folder root seperti di htdocs.



The screenshot shows the FPDF website's download page. On the left, there's a sidebar with links for Home, Downloads, Tutorials, and Manual, each accompanied by small icons. Below these are icons for the United Kingdom, France, and Italy flags. The main content area has a large blue elephant logo on the left and the word "Downloads" in a large purple font. Below it, a purple link says "You can view the changelog [here](#)". To the right, there are four bullet points, each with a colored square icon (blue, red, green, and orange) followed by text about manual translations in ZIP and TGZ formats, with the translator's name in blue text.

- An Arabic version of the 1.53 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Abdullah E.AL-Mehmadi](#))
- A Catalan version of the 1.81 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [David Gimeno i Ayuso](#))
- A Hungarian version of the 1.81 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Csaba Varga](#))
- An Indonesian version of the 1.82 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Kang Oedin](#))

Hello World! Sebuah Contoh Sederhana

Contoh sederhana untuk membuat file PDF melalui PHP dengan library FPDF dapat dilihat pada program berikut ini.

```
1 <?php
2 require('fpdf16/fpdf.php');
3
4 $pdf = new FPDF();
5 $pdf→AddPage();
6 $pdf→SetFont('Arial','B',16);
7 $pdf→Cell(40,10,'Hello World!');
8 $pdf→Output();
9 ?>
```

Penjelasan:

- **Baris 2.** Menyertakan file library FPDF. Jika library disimpan di dalam folder tertentu, sertakan nama foldernya. Fungsi require() di atas dapat diganti dengan include(), include_once() maupun require_once() (Lihat manual PHP untuk keterangan lebih lanjut mengenai fungsi tersebut).
- **Baris 4.** Membuat object baru bernama \$pdf dari class FPDF(). Dengan terbentuknya object tersebut, maka konstruktor dari class FPDF akan otomatis dijalankan. Konstruktor antara lain mengatur orientasi kertas, ukuran kertas (paper size) dan satuan ukuran kertas. Sebagai contoh lain, jika kita ingin membuat halaman FPDF dengan ukuran kertas Letter, orientasi Lanscape, dan satuan ukuran kertas dalam centimetres (cm) maka pembentukan object menjadi \$pdf = new FPDF('L', 'cm', 'Letter').
- **Baris 5.** Fungsi AddPage() digunakan untuk membuat halaman PDF baru (insert new page). Jika diinginkan suatu halaman dengan ukuran maupun orientasi berbeda, maka dapat ditambahkan parameter di fungsi ini. Contohnya AddPage('P', 'A4')
- **Baris 6.** Fungsi SetFont() digunakan untuk mengatur font tulisan yang akan ditampilkan. Dalam program diatas diatur font tulisan sebagai Arial, ukuran 16 dan huruf tebal (bold).
- **Baris 7.** Fungsi Cell() digunakan untuk menampilkan suatu text di halaman. Text akan ditampilkan dengan ukuran sesuai yang sudah diatur dengan SetFont(). Parameter yang dapat disertakan dalam fungsi Cell() secara berurutan adalah lebar cell, tinggi cell, text yang akan ditampilkan, border cell, pilihan pindah baris setelah cell, perataan text, isi dari cell dan terakhir pilihan untuk menambahkan link pada text.
- **Baris 8.** Fungsi Output() digunakan untuk menampilkan atau meng-generate halaman PDF. Kita juga dapat menentukan nama file dan target output dari PDF yang dihasilkan. Target dapat ke file, browser atau download.

Membuat Laporan PDF dengan HTML2PDF

Apa itu HTML2PDF?

HTML2PDF adalah sebuah library yang berguna untuk membuat report atau laporan dengan format PDF. License pada library HTML2PDF ini Open Source, jadi bisa Anda gunakan secara GRATIS. Cara Kerja dari library HTML2PDF ini yaitu dengan mengkonversi sebuah halaman HTML yang sudah Anda buat ke dalam bentuk format PDF. Library ini sangatlah mudah digunakan, untuk bisa menggunakan library ini Anda harus mengerti dan menguasai script HTML.

Instalasi HTML2PDF

Persiapan pertama yang harus dilakukan adalah mendownload library HTML2PDF (<https://html2pdf.fr/download>). Kita dapat meletakan Folder HTML2PDF tersebut pada project kita atau pada folder root seperti di htdocs.



Menggunakan HTML2PDF

Berikut langkah-langkah penggunaan HTML2PDF:

1. Download dan extract HTML2PDF , kemudian copy kan folder nya pada project yang kita miliki. Anda dapat meng-copy nya pada folder project BS4_CRUD seperti pada minggu sebelumnya.
2. Dari hasil tersebut, maka pada folder Project akan terdapat folder HTML2PDF yang akan kita gunakan sebagai library untuk menghasilkan report PDF.

Hello World! Sebuah Contoh Sederhana

Contoh sederhana untuk membuat file PDF melalui PHP dengan library HTML2PDF dapat dilihat pada program berikut ini, anda dapat menyimpan dengan nama file “**report_html2pdf.php**”



```
1 <?php
2     $content = "
3         <html>
4             <body>
5                 <h1>Hello World !</h1>
6                 Hasil Export dengan HTML2PDF
7             </body>
8         </html>
9     ";
10
11    require __DIR__ . '/html2pdf-master/vendor/autoload.php';
12    use Spipu\Html2Pdf\Html2Pdf;
13    $html2pdf = new Html2Pdf('P', 'A4', 'fr', true, 'UTF-8', array(15, 15, 15, 15), false);
14    $html2pdf->writeHTML($content);
15    $html2pdf->output();
16 ?>
```

Penjelasan:

- **\$content** merupakan variabel yang menampung data baris kode html yang akan di convert menjadi output PDF.
- Pada baris 11, require merupakan baris untuk menentukan root folder dari HTML2PDF agar dapat memanggil keseluruhan fungsi yang digunakan untuk menghasilkan report.
- Pada baris 13, \$html2pdf merupakan inisiasi awal untuk memulai menghasilkan report PDF. Keterangan lebih lanjut bisa baca di <https://github.com/spipu/html2pdf/blob/master/doc/basic.md>
- Baris terakhir berupa \$html2pdf->output(); sebagai tanda selesai. Sehingga apabila halaman tersebut di klik maka akan men-generate report berupa PDF.
- Untuk menentukan report nya tersimpan pada direktori tertentu, gunakan baris perintah seperti berikut:

```
1 $html2pdf->output(__DIR__ . "/contoh_laporan.pdf", "F");
```

Upload File Dengan PHP Dan MySQL

Upload file adalah salah satu yang terpenting di dalam membuat aplikasi. khususnya aplikasi berbasis web. karena terkadang kita di haruskan membuat upload file, bisa itu fitur upload file laporan, atau mungkin upload file gambar seperti pada facebook contoh nya. di facebook terdapat fitur untuk mengupload gambar untuk di jadikan foto profil. dan contoh aplikasi lain yang harus ada fitur upload file nya adalah aplikasi manajemen arsip. contoh nya manajemen arsip kantor. di mana data-data pdf atau document lainnya bisa di upload untuk di arsipkan.

Ada beberapa perintah pada PHP yang digunakan pada saat upload file, seperti berikut:

1. `$_FILES['file']['name']` = berguna untuk mendapatkan nama file yang di upload.
2. `$_FILES['file']['size']` = mendapatkan ukuran file yang di upload.
3. `$_FILES['file']['tmp_name']` = untuk mendapatkan temporary file yang di upload (tmp).
4. Fungsi atau function `move_uploaded_file()` berfungsi untuk mengupload. atau dengan kata lain dengan fungsi ini lah yang bertugas untuk memindahkan file yang di upload tadi dari temporary file menuju ke folder file. dan memberikan nama file nya sesuai yang kita tentukan.
5. `$ekstensi_diperbolehkan = array('png', 'jpg');` Digunakan apabila kita menginginkan hanya menerima file dengan extension tertentu saja.

Contoh baris program sederhana untuk mengupload gambar:

```
1 <?php
2 include 'koneksi.php';
3 if($_POST['upload']){
4     $ekstensi_diperbolehkan = array('png', 'jpg');
5     $nama = $_FILES['file']['name'];
6     $x = explode('.', $nama);
7     $ekstensi = strtolower(end($x));
8     $ukuran = $_FILES['file']['size'];
9     $file_tmp = $_FILES['file']['tmp_name'];
10    if(in_array($ekstensi, $ekstensi_diperbolehkan) === true){
11        if($ukuran < 1044070){
12            move_uploaded_file($file_tmp, 'file/'.$nama);
13            $query = mysql_query("INSERT INTO upload VALUES(NULL, '$nama')");
14            if($query){
15                echo 'FILE BERHASIL DI UPLOAD';
16            }else{
17                echo 'GAGAL MENGUPLOAD GAMBAR';
18            }
19        }else{
20            echo 'UKURAN FILE TERLALU BESAR';
21        }
22    }else{
23        echo 'EKSTENSI FILE YANG DI UPLOAD TIDAK DI PERBOLEHKAN';
24    }
25}
26 ?>
```

Beberapa perintah tersebut biasa digunakan juga sebagai validasi terhadap sebuah file sebelum kemudian diproses lebih dulu. Ada beberapa kasus, seperti file yang diijinkan hanya berukuran kurang dari 10mb, atau extension file yang digunakan hanya boleh .pdf.

Validasi upload file dengan JavaScript

Seperti yang kita ketahui bahwa proses validasi sebaiknya sudah selesai pada sisi client sebelum diproses, hal tersebut agar tidak membebankan server untuk memeriksa dan memproses. Agar memudahkan kita dapat menggunakan bantuan javascript.

Memeriksa Ekstensi File dengan JavaScript

Berikut contoh baris program sederhana:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5 </head>
6 <body>
7 <script type="text/javascript">
8   function validasiFile(){
9     var inputFile = document.getElementById('upload_file');
10    var pathFile = inputFile.value;
11    var ekstensiOk = /(\.pdf|\.PDF)$/i;
12    if(!ekstensiOk.exec(pathFile)){
13      alert('Silakan upload file yang memiliki ekstensi .pdf');
14      inputFile.value = '';
15      return false;
16    }
17  }
18 </script>
19
20 <input type="file" class="form-control" id="upload_file" name="upload_file" onchange="return
21   validasiFile();"/>
22 </body>
23 </html>
```

Penjelasan:

- Sebuah fungsi pada javascript untuk memeriksa ekstensi pada file dengan nama fungsi **validasiFile()**.
- Memeriksa element input file yang memiliki id **upload_file**.
- Mengambil value dari input file dan menyimpan pada variabel **pathFile**.
- Menentukan ekstensi yang di terima pada variabel **ekstensiOk**.
- Membandingkan **pathFile** (berisikan ekstensi) dengan **ekstensiOk**.
- Menghasilkan pesan alert apabila tidak sesuai dan me-reset value upload_file menjadi kosong kembali.

Memeriksa Ukuran File dengan JavaScript

Berikut contoh baris program sederhana:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5 </head>
6 <body>
7 <script type="text/javascript">
8   function validasiFile(){
9     var inputFile = document.getElementById('upload_file');
10    var pathFile = inputFile.value;
11    if (inputFile.files.length > 0) {
12      for (const i = 0; i <= inputFile.files.length - 1; i++) {
13        const fsize = inputFile.files.item(i).size;
14        const file = Math.round((fsize / 1024));
15        // The size of the file.
16        if (file >= 10485) {
17          alert("Ukuran File Terlalu Besar, Ukuran file yang diizinkan < 10MB");
18          inputFile.value = '';
19          return false;
20        }
21      }
22    }
23  }
24 </script>
25
26 <input type="file" class="form-control" id="upload_file" name="upload_file" onchange="return
27   validasiFile();"/>
28 </body>
29 </html>
```

Penjelasan:

- Sebuah fungsi pada javascript untuk memeriksa ekstensi pada file dengan nama fungsi **validasiFile()**.
- Memeriksa element input file yang memiliki id **upload_file**.
- Memeriksa apakah file tersebut memiliki ukuran size lebih dari 0, kemudian di proses.
- Menghitung ukuran size dari file yang di pilih dalam satuan byte
- Membandingkan, apakah ukuran size sesuai dengan yang sudah ditentukan.

XVII CURL PADA PHP

Agar dapat berkomunikasi dengan aplikasi yang lain, aplikasi kita harus mampu melakukan HTTP Request. Salah satu library yang kita bisa gunakan untuk melakukan HTTP Request di PHP adalah CURL.

Apa itu Curl?

Curl adalah sebuah program dan library untuk mengirim dan mengambil data melalui URL. Curl adalah sebuah program, artinya curl adalah sebuah program atau tools yang digunakan pada command line (CMD). Curl adalah sebuah library, artinya sekumpulan fungsi-fungsi curl yang dibungkus dalam paket libcurl dan bisa digunakan dalam berbagai macam bahasa pemrograman. Curl bisa juga diterapkan untuk:

- Mengambil api json lalu menampilkannya;
- Mengambil halaman web lalu mengubahnya jadi PDF;
- Upload dan Download File;
- Melakukan Login;
- Scrape;
- dan sebagainya.

Sejarah Singkat Curl

Curl awalnya bernama HttpGet dari versi pertama sampai ke-3. HttpGet awalnya hanya mendukung protokol Http saja lalu semakin berkembang dan banyak protokol yang ditambahkan. Akhirnya pada rilis versi yang ke-4, nama HttpGet tidak lagi digunakan. Sekarang namanya adalah Curl (mulai ditetapkan pada 20 Maret 1998). Diberikan nama Curl, karena digunakan untuk download/upload data melalui URL. Huruf "c" artinya "See" (melihat), jadi kalau cURL di-eja menjadi "See URL". Selain itu, huruf "c" juga mengacu kepada client.

Instalasi dan Konfigurasi Curl

Ketika kita menggunakan XAMPP maka curl secara otomatis sudah diaktifkan kita dapat mengeceknya dengan melihat informasi pada phpinfo (<http://localhost/dashboard/phpinfo.php>). Namun apabila web server yang kita gunakan adalah linux, maka kita dapat melakukan installasi seperti berikut:

```
apt install curl
```

atau

```
apt install php-curl
```

Untuk yang menggunakan xampp tampilan informasi akan seperti ini:

cURL support	enabled
cURL Information	7.64.0
Age	4
Features	
AsynchDNS	Yes
CharConv	No
Debug	No
GSS-Negotiate	No
IDN	Yes
IPv6	Yes
krb4	No
Largefile	Yes
libz	Yes
NTLM	Yes
NTLMWB	No
SPNEGO	Yes
SSL	Yes
SSPI	Yes
TLS-SRP	No
HTTP2	Yes
GSSAPI	No
KERBEROS5	Yes
UNIX_SOCKETS	No
PSL	No
Protocols	dict, file, ftp, ftps, gopher, http, https, imap, imaps, ldap, ldaps, pop3, pop3s, rtsp, scp, sftp, smb, smbs, smtp, smtps, telnet, tftp
Host	x86_64-pc-win32
SSL Version	OpenSSL/1.0.2s
ZLib Version	1.2.8
libSSH Version	libssh2/1.8.2

Cara Menggunakan Curl pada PHP

Ada 4 langkah penggunaan Curl di PHP:

1. Inisialisasi;
2. Set Option;
3. Eksekusi Curl;
4. Tutup Curl;

Setiap kita ingin menggunakan fungsi Curl, kita harus melakukan inisialisasi terlebih dahulu dengan cara seperti ini:

```
1 <?php
2 // create curl resource
3 $ch = curl_init();
```

Fungsi yang digunakan untuk melakukan inisialisasi adalah curl_init().

Setelah itu, kita harus memberikan nilai options seperti alamat URL yang akan dituju, format hasilnya, header, dll. Untuk memberikan options, kita menggunakan fungsi curl_setopt() seperti ini:

```
1 // set url
2 curl_setopt($ch, CURLOPT_URL, "example.com");
3
4 //return the transfer as a string
5 curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
```

Berikutnya melakukan eksekusi:

```
1 // $output contains the output string
2 $output = curl_exec($ch);
```

Pada tahapan eksekusi, Curl akan melakukan HTTP Request sesuai dengan options yang diberikan. Fungsi yang digunakan untuk mengeksekusi Curl adalah curl_exec(). Karena kita sudah memberikan options hasil Curl akan berupa string, maka variabel \$output akan berisi sebuah string.

Kita bisa melihat isinya dengan echo.

```
1 echo $output;
```

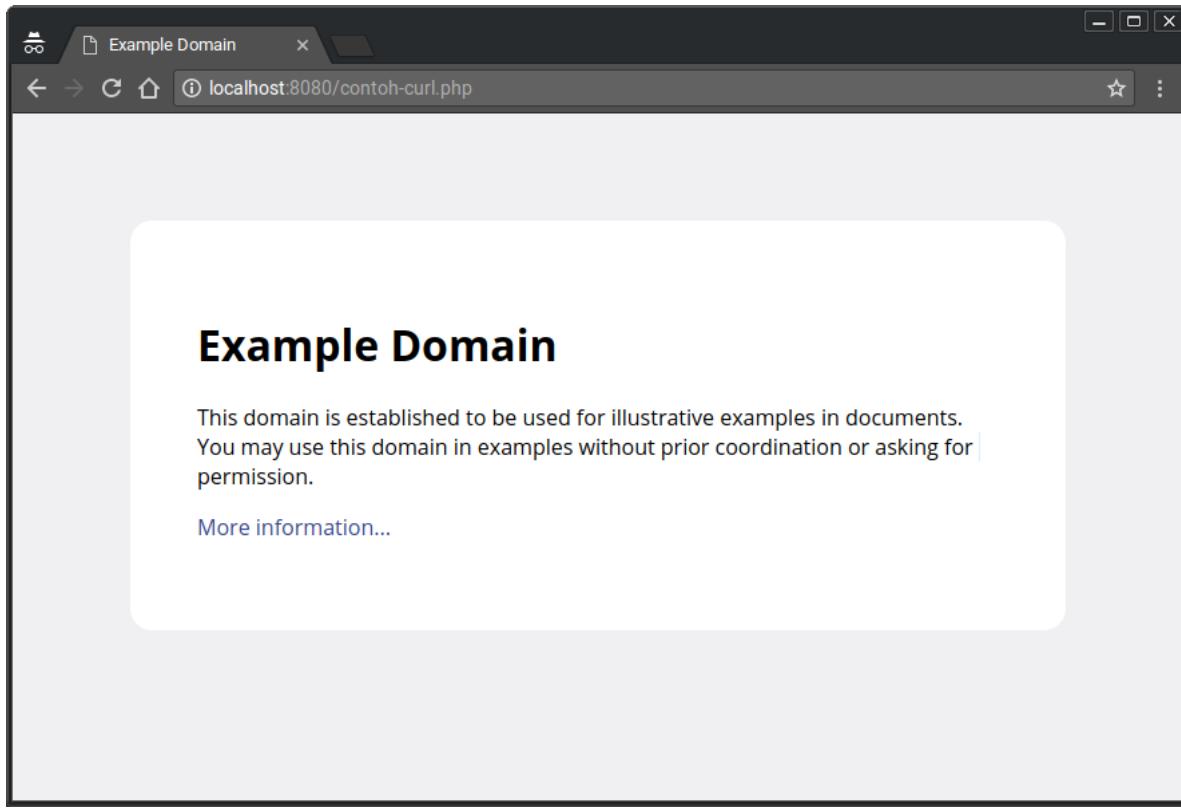
Terakhir menutup Curl dengan fungsi curl_close(), karena sudah tidak digunakan lagi.

```
1 // close curl resource to free up system resources
2 curl_close($ch);
```

Kode lengkap nya akan menjadi seperti ini:

```
1 <?php
2     // persiapkan curl
3     $ch = curl_init();
4
5     // set url
6     curl_setopt($ch, CURLOPT_URL, "example.com");
7
8     // return the transfer as a string
9     curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
10
11    // $output contains the output string
12    $output = curl_exec($ch);
13
14    // tutup curl
15    curl_close($ch);
16
17    // menampilkan hasil curl
18    echo $output;
19 ?>|
```

Saat kita eksekusi skrip PHP di atas, maka akan menghasilkan seperti ini:



Halaman di atas didapat dari example.com.

Membungkus Curl dalam Fungsi

Baru kode di atas bisa kita buat dalam bentuk fungsi agar tidak diketik ulang terus menerus. Fungsinya akan menjadi seperti ini:

```
1 <?php
2 function http_request($url){
3     // persiapkan curl
4     $ch = curl_init();
5     // set url
6     curl_setopt($ch, CURLOPT_URL, $url);
7     // return the transfer as a string
8     curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
9     // $output contains the output string
10    $output = curl_exec($ch);
11    // tutup curl
12    curl_close($ch);
13    // mengembalikan hasil curl
14    return $output;
15 }
16
17 $data = http_request("https://www.example.com/");
18 ?>
```

Mengambil Data JSON dengan Curl

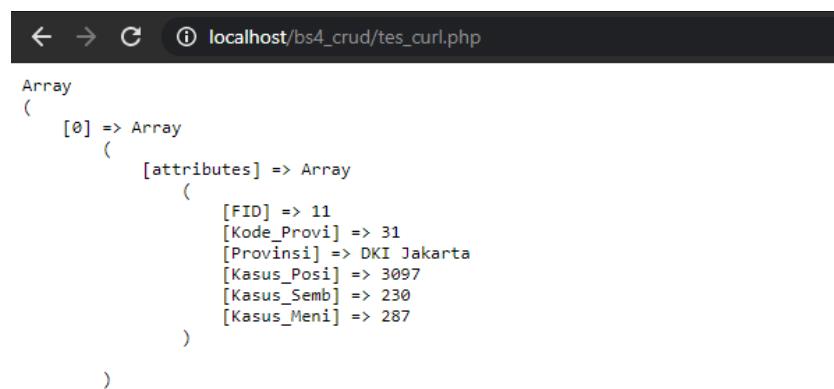
Biasanya webservice menyediakan data berupa JSON. Data JSON ini bisa kita ambil dengan CURL, lalu melakukan parse dengan fungsi json_decode(). Contoh:

Kita akan mengambil data JSON dari Github: <https://api.kawalcorona.com/indonesia/provinsi/>

```
1 <?php
2 function http_request($url){
3     // persiapkan curl
4     $ch = curl_init();
5     // set url
6     curl_setopt($ch, CURLOPT_URL, $url);
7     // set user agent
8     curl_setopt($ch,CURLOPT_USERAGENT,'Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.13) Gecko/20080311 Firefox/2.0.0.13');
9     // return the transfer as a string
10    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
11    // $output contains the output string
12    $output = curl_exec($ch);
13    // tutup curl
14    curl_close($ch);
15    // mengembalikan hasil curl
16    return $output;
17 }
18
19 $profile = http_request("https://api.kawalcorona.com/indonesia/provinsi/");
20
21 // ubah string JSON menjadi array
22 $profile = json_decode($profile, TRUE);
23
24 echo "<pre>";
25 print_r($profile);
26 echo "</pre>";
```

Perhatikan options yang kita berikan pada contoh di atas, Kita memberikan option user agent, karena untuk mengakses API, kita harus menyertakan user agent.

Hasilnya akan seperti ini:



```
← → C ⓘ localhost/bs4_crud/tes_curl.php

Array
(
    [0] => Array
        (
            [attributes] => Array
                (
                    [FID] => 11
                    [Kode_Provi] => 31
                    [Provinsi] => DKI Jakarta
                    [Kasus_Positif] => 3097
                    [Kasus_Sembuh] => 230
                    [Kasus_Meninggal] => 287
                )
        )
)
```

Mengubah Data JSON ke HTML

Kita dapat melakukan apapun terhadap data yang kita dapatkan. Bisa disimpan ke database atau ditampilkan ke dalam HTML.

Seperti ini:

```
1 <?php
2 $ch = curl_init();
3 curl_setopt($ch, CURLOPT_URL, 'https://api.kawalcorona.com/indonesia/provinsi/');
4 curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
5 $content = curl_exec($ch);
6 curl_close($ch);
7 //mengubah data json menjadi data array assosiatif
8 $result=json_decode($content,true);
9 ?>
10 <div class="container">
11     <h1>Data Sebaran Covid-19 Di Indonesia</h1>
12     <p>List Data Per Provinsi</p>
13     <table class="table">
14         <thead class="thead-dark">
15             <tr>
16                 <th scope="col">Provinsi</th>
17                 <th scope="col">Positif</th>
18                 <th scope="col">Sembuh</th>
19                 <th scope="col">Meninggal</th>
20             </tr>
21         </thead>
22         <tbody>
23             <?php
24                 $no = 1;
25                 foreach ($result as $value) {
26                     echo '
27                         <tr>
28                             <td>' . $value['attributes']['Provinsi'] . '</td>
29                             <td>' . $value['attributes']['Kasus_Posi'] . '</td>
30                             <td>' . $value['attributes']['Kasus_Semb'] . '</td>
31                             <td>' . $value['attributes']['Kasus_Meni'] . '</td>
32                         </tr>';
33                     $no++;
34                 }
35             ?>
36         </tbody>
37     </table>
38 </div>
```

Perhatikan pada baris program yang di beri garis kotak. Pada foreach \$result akan juga dikenali sebagai \$value. Karena \$result menghasilkan data json dalam bentuk array assosiatif.

```

Array
(
    [0] => Array
        (
            [attributes] => Array
                (
                    [FID] => 11
                    [Kode_Provi] => 31
                    [Provinsi] => DKI Jakarta
                    [Kasus_Positif] => 3097
                    [Kasus_Sembuh] => 230
                    [Kasus_Meninggal] => 287
                )
        )
)

```

Maka ketika kita ingin mengambil nilai maka digunakan:

```
$value['attributes']['Provinsi']
```

Output nya akan seperti berikut:

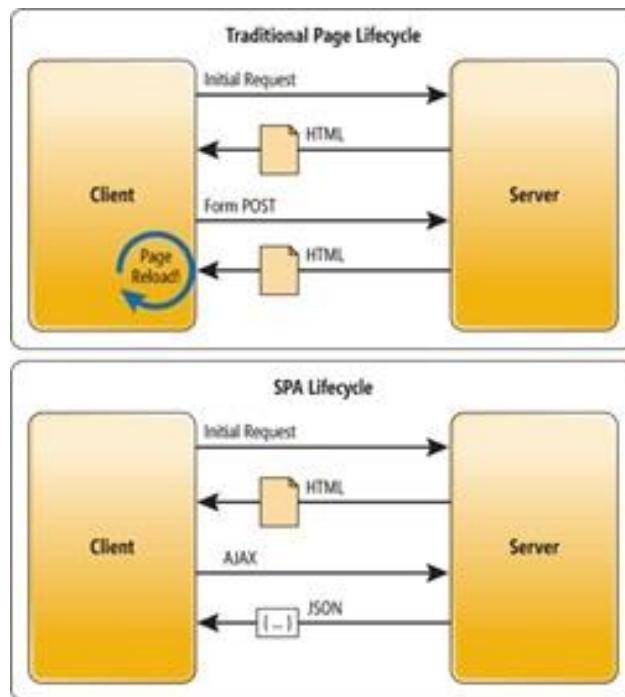
Data Sebaran Covid-19 Di Indonesia

List Data Per Provinsi

Provinsi	Positif	Sembuh	Meninggal
DKI Jakarta	3097	230	287
Jawa Barat	747	56	62
Jawa Timur	590	98	56
Sulawesi Selatan	370	63	25
Jawa Tengah	351	51	44
Banten	341	17	35
Bali	140	42	3
Papua	107	28	6
Kalimantan Selatan	96	9	6
Sumatera Selatan	89	5	3
Sumatera Utara	83	13	9
Kepulauan Riau	79	6	7
Sumatera Barat	74	13	7
Kalimantan Utara	74	2	1

XVIII SINGLE PAGE APPLICATION (SPA)

Secara teori, single page application merupakan aplikasi yang bekerja di dalam browser yang tidak membutuhkan reload page saat digunakan. Dengan kata lain, pengguna atau user tidak akan berpindah halaman dengan melakukan request kepada server setiap kali terjadi interaksi pada aplikasi. Yang membedakan SPA dengan non-SPA adalah single page application hanya akan melakukan load terhadap satu halaman dari server kemudian mekanisme routing yang biasanya di-handle oleh server kini dibebankan pada client. Akibatnya, website yang menggunakan SPA memiliki performa yang lebih cepat tanpa harus load halaman secara terus menerus.



Adapun 2 keuntungan utama yang didapat ketika menerapkan single page application yaitu mengurangi penggunaan bandwidth jaringan dan proses navigasi yang lebih cepat. Bandwidth jaringan berkurang dikarenakan SPA ini banyak didukung dengan library JavaScript. Sedangkan proses navigasi yang lebih cepat dikarenakan data yang diterima dari server berupa JSON (JavaScript Object Notation) sehingga bisa di-render secara asynchronous dengan menggunakan JavaScript.

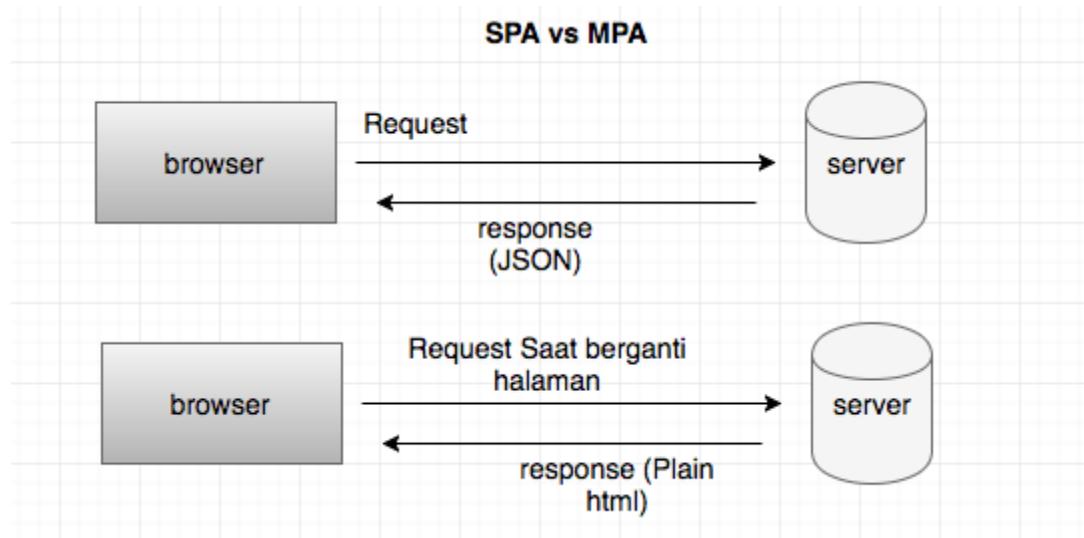
Single Page Application (SPA) adalah salah satu jenis aplikasi website dimana hanya ada 1 halaman yang meng-handle semua aktivitas yang terjadi dalam aplikasi tersebut. User tidak akan berpindah halaman browser ketika melakukan request seperti mengisi formulir, klik link, atau aksi lain yang membutuhkan data dari server.

Semua aksi-aksi tersebut akan di-handle secara Asynchronous oleh Javascript (seperti AJAX). Perpindahan 1 halaman ke halaman lain di handle dengan sistem routing, tanpa loading browser sehingga prosesnya lebih cepat dan lebih baik UX-nya. SPA sangat bergantung pada utilitas browser karena menggunakan javascript sepenuhnya. Sementara untuk berkomunikasi terhadap backend tetap menggunakan http request terhadap suatu API/Web services.

Teknologi SPA berkembang seiring dengan kebutuhan industri pada aplikasi website yang dapat bekerja lebih cepat dan hemat resource sehingga akan menghemat pengeluaran untuk server. Saat ini, website SPA umumnya digunakan pada aplikasi dashboard atau aplikasi sejenis yang mana pada satu halaman membutuhkan banyak informasi dan data dari proses bisnis, transaksi, perhitungan, dan sebagainya.

Single Page Application VS Multi Page Application

Yang membedakan SPA dari Multi Page Application, adalah bagaimana SPA melakukan request dan resource. SPA lebih hemat dibanding dengan Multi Page Application. Contohnya, pada saat berpindah ke halaman baru, SPA hanya me-request data-data yang diperlukannya saja berupa JSON (Javascript Object Notation), tanpa me-request ulang static files seperti file css, html, dan js nya karena itu sudah di load saat pertama kali.



Berbeda dengan Multipage Application yang setiap perpindahan halaman akan merequest ulang seluruh elemen baik itu static files dan data baru untuk halaman tersebut. Contoh website yang menggunakan SPA antara lain gmail, facebook, github, techinasia dan lain sebagainya. Untuk multipage web application contohnya adalah wikipedia, kaskus, bukalapak dan lain-lain.

Mekanisme yang berjalan pada SPA layaknya kita menggunakan browser secara native dimana secara penuh memanfaatkan perpindahan halaman, data binding, form, dan lain sebagainya dengan menggunakan javascript. Tidak ada waktu menunggu saat terjadi perpindahan halaman. Hal seperti ini memberikan UX(User experience) yang luar biasa kepada user.

Keuntungan Single Page Application

Lebih cepat dibandingkan dengan multi page application.

Kebanyakan resource seperti HTML, CSS dan JS diload pada saat pertama kali aplikasi dibuka dan akan berfungsi terus sampai aplikasi ditutup. Perpindahan halaman tidak perlu melakukan request keseluruhan elemen halaman, tetapi hanya data tertentu saja yang belum di request saat inisiasi pertama kali. Ada konsep “virtual DOM” dimana perubahan pada DOM hanya terjadi pada elemen yang berubah. Sedangkan pada Multi Page Application, DOM akan diperbarui secara menyeluruh karena perpindahan halaman harus melakukan request kepada server dan server memberikan respon berupa html.

Tidak perlu mensetup “server”.

SPA hanya memiliki satu buah index.html, beberapa css dan beberapa javascript. Sehingga kita cukup drag index.html tersebut untuk menjalankan aplikasi SPA. Namun tentunya ini sangat basic. Kita tetap membutuhkan server jika ingin mengoptimasi performa dari SPA itu sendiri.

Proses debug sangat mudah cukup dengan menggunakan browser (console).

Jika terjadi error cukup refresh pada browser maka akan tampil errornya di console tidak perlu melalui proses compile. Pada SPA compile akan dilakukan diakhir saat seluruh aplikasi telah selesai dikerjakan(build) dengan menggunakan webpack dan babel(untuk transpile) untuk menghasilkan index.html , file-file css serta file-file javascript yang telah digabungkan menjadi satu.

Satu kode untuk berbagai macam platform.

Dengan javascript kita dapat membuat website, desktop, android, iOS, Windows phone. Terdapat beberapa framework untuk membangun mobile apps dengan javascript antara lain react native dan ionic (berbasiskan angular)

Proses Caching lebih efektif karena memanfaatkan local storage pada browser.

Sebuah aplikasi dapat mengirim hanya 1 request, lalu responnya di simpan dalam local storage sehingga dapat digunakan kembali bahkan dalam keadaan offline sekalipun.

Kekurangan Single Page Application

Tidak bagus dalam hal SEO.

Optimasi SEO pada SPA sangat sulit dilakukan karena sejatinya hanya ada 1 halaman yang bisa diindex oleh mesin pencari, serumit dan sebanyak apapun data dari aplikasi tersebut. Route pada SPA tidak bisa dijadikan sebagai URL yang bisa diindex pada mesin pencari. Saat ini, mulai dikembangkan tools SPA untuk keperluan SEO. Metodenya adalah mem-build SPA secara server-side sehingga setiap route yang dihasilkan akan menjadi URL. Contoh tools ini adalah Angular Universal dan Server-side Rendering pada VueJS

Berat saat di-load/buka pertama kali.

SPA merequest kebanyakan resourcenyaa saat pertama kali aplikasi dibuka. Hal ini pastinya menyebabkan loading yang sangat lama.

Sangat bergantung pada Javascript.

Semua aksi dalam SPA menggunakan javascript, sehingga ketika javascript di browser dimatikan oleh user aplikasi tidak akan berjalan sama sekali.

Kurang aman dibanding dengan Website biasa.

SPA rawan terhadap serangan Cross-site Scripting (XSS), yang memungkinkan hacker me-inject script client-side (Javascript) pada aplikasi.

Masalah kompatibilitas browser.

Karena SPA mengandalkan javascript dan browser, ketidakcocokan antara browser satu dan lainnya lebih beresiko terjadi. Ini bisa disebabkan karena user menggunakan browser lama, perbedaan versi javascript pada browser yang berbeda, dan sebagainya.

Bagaimana Cara Membuat Single Page Application ?

Pada chapter 8 atau Bab XIII anda sudah belajar dan mengimplementasi Create read update dan delete dengan konsep multi page pada PHP. Kali ini kita akan memodifikasi dan sesuaikan dengan konsep Single page application. Yang dipersiapkan adalah:

1. Halaman utama (index), yang merupakan halaman yang akan menampilkan halaman lain.
2. Halaman lain seperti form input, edit, dan tampil data.
3. File service.php yang akan berfungsi untuk memproses insert, update, dan juga delete data.
4. File config.php sebagai koneksi.
5. Boostrap sebagai tampilan dan Library jQuery.

Langkah demi langkah terdapat pada modul praktikum.