

---

PANDUAN PRAKTIS

**PEMROGRAMAN WEB**

---

**Wirasto S. Karim**

September 2012

*Hak cipta buku ini tetap pada penulis. Tetapi buku ini bebas untuk diperbanyak, dikutip baik sebagian atau seluruhnya bahkan disebarluaskan selama bukan untuk tujuan komersial, dan dengan syarat tidak menghapus atau merubah atribut penulis.*

## **PENGANTAR**

Bagian “pengantar” akan saya selesaikan pada rilis berikutnya. Untuk sementara silahkan dinikmati saja dulu panduan ini apa adanya.

Jika Anda mendapatkan panduan ini dalam bentuk tercetak (atau *fotocopy*-an), maka Anda tetap bisa mendapatkan berkas aslinya secara gratis di:

[http://repository.ung.ac.id/categori/materi\\_kuliah/](http://repository.ung.ac.id/categori/materi_kuliah/)

Gorontalo, September 2012

Wirasto S. Karim

wirastokarim@gmail.com

# **BAB I**

## **WORLD WIDE WEB**

### **WWW**

World Wide Web (www) atau yang lebih dikenal dengan sebutan web, merupakan salah satu layanan yang didapat oleh para pemakai komputer yang terhubung ke internet. Web adalah ruang informasi dengan menggunakan teknologi hyperlink untuk menghubungkan tiap dokumen. Dengan teknologi ini pemakai dituntun untuk menemukan informasi dengan mengikuti tautan (link) yang disediakan dalam halaman web yang ditampilkan lewat web browser.

Internet identik dengan web, karena kepopuleran web sebagai interface pada layanan-layanan yang ada di Internet. Dimana yang pada awalnya web hanya sebagai penyedia informasi kini digunakan juga untuk berkomunikasi surat elektronik (email), chatting sampai dengan video conference.

### **Bagaimana WWW Bekerja ?**

- Informasi web disimpan dalam dokumen yang disebut dengan halaman-halaman web (web pages)
- Web page adalah berkas-berkas yang disimpan dalam komputer yang disebut dengan server-server web (web server)
- Komputer-komputer yang mengakses web page disebut sebagai web client
- Web client menampilkan web page dengan menggunakan program yang disebut dengan web browser

### **HTTP**

HyperText Transfer Protocol (HTTP) merupakan protokol yang dipergunakan untuk mentransfer dokumen dalam www. HTTP mendefinisikan bagaimana suatu pesan bisa diformat dan dikirimkan dari server ke client. HTTP juga mengatur aksi-aksi apa saja yang harus dilakukan oleh web server dan juga web browser sebagai respon atas permintaan-permintaan yang lewat dalam protokol http.

Sebagai contoh, ketika kita mengetikkan suatu alamat atau URL pada web browser, maka sebenarnya web browser akan mengirimkan permintaan http ke web server. Web server kemudian akan menerima permintaan ini dan melakukan proses sesuai dengan yang diminta oleh web browser (misalnya akses ke database, file, e-mail dan lain sebagainya). Hasil proses tadi akan dikirimkan kembali ke web browser untuk ditampilkan kepada pengguna.

HTTP bekerja diatas TCP (Transmission Control Protocol) yang menjamin sampainya data di tujuan dalam urutan yang benar. Bila suatu kesalahan terjadi selama proses pengiriman, pihak pengirim akan mendapat pemberitahuan bahwa telah terjadi ketidakberesan dan perlu mengirimkan permintaan ulang.

## **URL**

Uniform Resource Locator (URL) merupakan alamat dari sebuah resource yang dapat diakses di internet. Resource yang dimaksud disini bisa berupa berkas halaman web, video, mp3, dll.

URL terdiri dari 3 bagian:

- Protokol (http:// atau https://)
- Alamat Server (lokasi data tersimpan)
- File Path (folder dan atau nama file)

Misalnya bila ada URL : <http://www.galau.com/foto/wirasto.jpg>

Berarti :

<http://> adalah protokol

[www.galau.com](http://www.galau.com) adalah alamat server

[/foto/wirasto.jpg](http://www.galau.com/foto/wirasto.jpg) adalah folder dan nama file

## **DNS**

Domain Name System (DNS) merupakan versi teks dari ip address. Dalam dunia jaringan komputer, komputer berkomunikasi satu sama lain dengan mengenali ip address-nya.

Namun bagi manusia tidak mungkin menghafalkan ip address tersebut, manusia lebih mudah menghafalkan kata-kata seperti www.yahoo.com, www.google.com, atau www.facebook.com. DNS berfungsi untuk mengkonversi nama yang mudah diingat menjadi ip address dari komputer yang ingin dihubungi.

## **JENIS WEB**

- Web Statis : dibentuk menggunakan client side scripting. Update dilakukan secara manual dengan melakukan perubahan terhadap script halaman tersebut.
- Web Dinamis : dibentuk menggunakan aplikasi tambahan (server side scripting dan database) sehingga data dapat diupdate tanpa harus mengubah script, namun cukup dengan mengubah data yang ada dalam database.

## **WEB PROGRAMING**

1. Client Side Scripting
  - HTML
  - Javascript
  - CSS
2. Server Side Scripting
  - PHP
  - ASP
  - Perl

# BAB II

## HTML

### APA ITU HTML?

Hyper Text Markup Language (html), merupakan bahasa markup yang digunakan untuk membuat halaman web dan menampilkan berbagai informasi didalam sebuah browser. Dokumen html (atau halaman web) terdiri dari elemen-elemen yang menginstruksikan browser untuk menghasilkan tampilan sesuai dengan yang diinginkan.

Ada dua cara untuk membuat halaman web dengan html yaitu menggunakan web designer atau dengan teks editor biasa, dan setiap berkas html disimpan dalam ekstensi .html. Dan perlu diingat bahwa nama berkas pada beberapa sistem operasi akan bersifat case-sensitive, sehingga berkas dengan nama dokumen.html berbeda dengan berkas DOKUMEN.html dan berbeda juga dengan berkas Dokumen.html.

Setiap website yang dibuat dengan menggunakan html memiliki dokumen utama yang diberi nama index.html. Sehingga bila ingin membuat website, wajib memiliki berkas ini.

### STRUKTUR DASAR DOKUMEN HTML

Seperti halnya bahasa-bahasa markup komputer yang lain, html juga memiliki struktur dasar. Adapun struktur dasar dari html adalah sebagai berikut:

```
<html>
<head>
    ..... Masukkan informasi dokumen disini .....
</head>
<body>
    ..... Masukkan informasi yang ingin ditampilkan disini .....
</body>
</html>
```

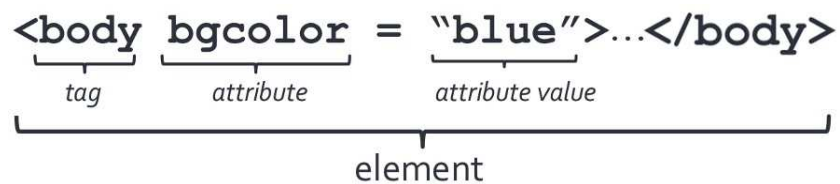
## Contoh dokumen html

```
index.html
<html>
<head>
  <title>Contoh Dokumen HTML</title>
</head>
<body bgcolor="blue">

Ini adalah contoh dari dokumen HTML. Dokumen HTML ini biasa juga disebut sebagai
halaman web ketika sudah ditampilkan di web browser.

</body>
</html>
```

Sudah disebutkan sebelumnya bahwa dokumen html terdiri dari elemen-elemen, maka berikut ini adalah struktur dari elemen:



- Elemen dalam dokumen html ditandai dengan penulisan tag yang berpasangan (contoh: `<body></body>`), walau ada pula yang tidak harus berpasangan (contoh: `<img>`)
- Tag diapit dengan dua karakter kurung bersudut `<` dan `>`
- Tag pertama dalam suatu elemen adalah tag pembuka dan yang kedua merupakan tag penutup
- Tag case-insensitive, sehingga `<body>` sama dengan `<BODY>`
- Tag dapat mempunyai atribut untuk mengubah default pemformatan dokumen dengan tag tersebut.



## MEMFORMAT DOKUMEN HTML

Teks dalam halaman web dapat diformat secara khusus untuk menunjukkan perbedaan dan penekanan terhadap isi dan maksud dari teks tersebut.

### Heading (Judul)

Heading terdiri dari 6 level atau tingkatan. Semakin besar tingkatannya maka semakin kecil tulisannya.

```
heading.html
<html>
<head>
  <title>Belajar Heading</title>
</head>
<body>

<h1>Judul Level 1</h1>
<h2>Judul Level 2</h2>
<h3>Judul Level 3</h3>
<h4>Judul Level 4</h4>
<h5>Judul Level 5</h5>
<h6>Judul Level 6</h6>

</body>
</html>
```

### Break (Pindah Baris)

Dalam dokumen html kita tidak bisa menulis teks pada baris yang baru dengan menekan enter saja. Namun harus menggunakan tag br

```
break.html
<html>
<head>
  <title>Belajar Break</title>
</head>
<body>
Saya Baris 1
Saya Baris 2, saya ingin di baris baru namun tidak bisa
<br>Saya Baris 3, saya ada dibaris baru karena ada tag br
</body>
</html>
```

## Paragraf

paragraf.html

```
<html>
<head>
  <title>Belajar Paragraf</title>
</head>
<body>

<p>Setiap website yang dibuat dengan menggunakan html memiliki dokumen utama
yang diberi nama index.html. Sehingga bila ingin membuat website, wajib memiliki
berkas ini.</p>

<p align="right">Setiap website yang dibuat dengan menggunakan html memiliki
dokumen utama yang diberi nama index.html. Sehingga bila ingin membuat website,
wajib memiliki berkas ini.</p>

<p align="center">Setiap website yang dibuat dengan menggunakan html memiliki
dokumen utama yang diberi nama index.html. Sehingga bila ingin membuat website,
wajib memiliki berkas ini.</p>

<p align="justify">Setiap website yang dibuat dengan menggunakan html memiliki
dokumen utama yang diberi nama index.html. Sehingga bila ingin membuat website,
wajib memiliki berkas ini.</p><p align="right"></p>

</body>
</html>
```

---

## Tebal, Miring, Garis Bawah

tmgb.html

```
<html>
<head>
  <title>Belajar T-M-GB</title>
</head>
<body>

<b>Ini ditulis tebal</b><br>
<i>Ini ditulis miring</i><br>
<u>Ini digaris bawah</u>

</body>
</html>
```

---

## Huruf

```
huruf.html
<html>
<head>
    <title>Belajar Huruf</title>
</head>
<body>

<font size=10>Ukuran huruf 10</font>
<br>
<font color="red">Warna huruf merah</font>
<br>
<font size=4 color="blue">Ukuran huruf 4 dan warna huruf biru</font>

</body>
</html>
```

---

## GAMBAR

Untuk menampilkan gambar dalam dokumen web, bisa menggunakan tag `img`. Tag `img` merupakan salah satu tag html yang tidak membutuhkan pasangan.

```
gambar.html
<html>
<head>
    <title>Belajar Gambar</title>
</head>
<body>



</body>
</html>
```

---

Tag `img` memiliki beberapa atribut, antara lain:

- `src` = untuk lokasi gambar yang akan ditampilkan
- `width` = untuk lebar gambar
- `height` = untuk tinggi gambar
- `title` = untuk tulisan yang akan muncul jika gambar disorot dengan mouse

## LIST

List merupakan bentuk umum yang biasa digunakan untuk menguraikan daftar. List dalam html antara lain adalah list yang berurutan/bernomor dan tidak bernomor.

### Ordered List

Ordered list merupakan daftar yang berurutan atau bernomor. Ordered list dibuat dengan menggunakan tag ol dan tiap itemnya dibuat dengan tag li

```
ol.html
<html>
<head>
  <title>Belajar OL</title>
</head>
<body>

<ol>
  <li>Slackware</li>
  <li>Ubuntu</li>
  <li>Fedora</li>
</ol>

<ol type="1">
  <li>Slackware</li>
  <li>Ubuntu</li>
  <li>Fedora</li>
</ol>

<ol type="i">
  <li>Slackware</li>
  <li>Ubuntu</li>
  <li>Fedora</li>
</ol>

<ol type="I">
  <li>Slackware</li>
  <li>Ubuntu</li>
  <li>Fedora</li>
</ol>

<ol type="a">
  <li>Slackware</li>
```

```
        <li>Ubuntu</li>
        <li>Fedora</li>
</ol>

<ol type="A">
    <li>Slackware</li>
    <li>Ubuntu</li>
    <li>Fedora</li>
</ol>

</body>
</html>
```

---

## Unordered List

Unordered list merupakan daftar yang tidak berurutan atau tidak bernomor. Unordered list dibuat dengan menggunakan tag ul dan tiap itemnya dibuat dengan tag li

### ul.html

```
<html>
<head>
    <title>Belajar UL</title>
</head>
<body>

<ul>
    <li>Slackware</li>
    <li>Ubuntu</li>
    <li>Fedora</li>
</ul>

<ul type="disc">
    <li>Slackware</li>
    <li>Ubuntu</li>
    <li>Fedora</li>
</ul>

<ul type="circle">
    <li>Slackware</li>
    <li>Ubuntu</li>
    <li>Fedora</li>
</ul>
```

```
<ul type="square">
  <li>Slackware</li>
  <li>Ubuntu</li>
  <li>Fedora</li>
</ul>
</body>
</html>
```

---

## Nested List

Nested list atau daftar yang bersarang merupakan daftar berupa gabungan dari ordered list dan unodererd list, atau bisa juga hanya salah satu diantaranya namun ada list dalam list.

### nl.html

```
<html>
<head>
  <title>Belajar NL</title>
</head>
<body>

<ol>
  <li>Slackware</li>
    <ol type="i">
      <li>Slackware</li>
      <li>Ubuntu</li>
      <li>Fedora</li>
    </ol>
  <li>Ubuntu</li>
    <ol type="a">
      <li>Slackware</li>
      <li>Ubuntu</li>
      <li>Fedora</li>
    </ol>
  <li>Fedora</li>
    <ul type="circle">
      <li>Slackware</li>
      <li>Ubuntu</li>
      <li>Fedora</li>
    </ul>
</ol>

</body>
</html>
```

---

## ANCHOR

Dokumen html menggunakan hyperlink (anchor) untuk menghubungkan satu dokumen ke dokumen lain. Anchor dibuat dengan menggunakan tag a.

Sebagai contoh kita memiliki 3 buah dokumen html, masing-masing adalah index.html sebagai dokumen utama, dan halaman\_1.html serta halaman\_2.html sebagai dokumen pendamping.

### index.html

```
<html>
<head>
  <title>Belajar Anchor</title>
</head>
<body>

<h1>Halaman Utama</h1>

<a href="halaman_1.html">Halaman 1</a>
<a href="halaman_2.html">Halaman 2</a>

</body>
</html>
```

---

### halaman\_1.html

```
<html>
<head>
  <title>Belajar Anchor</title>
</head>
<body bgcolor=red>

<h1>Halaman 1</h1>

<a href="index.html">Halaman Utama</a>
<a href="halaman_2.html">Halaman 2</a>

</body>
</html>
```

---

### halaman\_2.html

```
<html>
<head>
```

```

        <title>Belajar Anchor</title>
</head>
<body bgcolor=lime>

<h1>Halaman 2</h1>

<a href="index.html">Halaman Utama</a>
<a href="halaman_1.html">Halaman 1</a>

</body>
</html>

```

---

## TABEL

Tabel digunakan untuk menyajikan data dalam bentuk baris dan kolom. Umumnya setiap kolom menunjukkan data yang sejenis, dan setiap baris yang terdiri atas kolom-kolom menunjukkan kelompok data dalam satu kesatuan.

Tabel dibuat dengan menggunakan tag table, baris dibuat dengan tag tr, sedangkan kolom untuk judul tabel dibuat dengan menggunakan th dan kolom untuk data menggunakan tag td.

```

tabel1.html
<html>
<head>
    <title>Pengenalan Tabel</title>
</head>
<body>

<table border=1>
    <tr>
        <th width=30>NO</th>
        <th width=200>NAMA</th>
        <th width=120>NIM</th>
        <th width=50>JENKEL</th>
    </tr>
    <tr>
        <td>1</td>
        <td>Wirasto S. Karim</td>
        <td>01010101</td>
        <td>Pria</td>

```



```

</tr>
<tr>
    <td>2</td>
    <td>Abdul Gani</td>
    <td>02020202</td>
    <td>Pria</td>
</tr>
<tr>
    <td>3</td>
    <td>Ramona</td>
    <td>03030303</td>
    <td>Wanita</td>
</tr>
</table>

</body>
</html>

```

### tabel2.html

```

<html>
<head>
    <title>Pengenalan Tabel</title>
</head>
<body>

<table border=1>
    <tr bgcolor=red>
        <th rowspan=2 width=30>NO</th>
        <th colspan=3>MAHASISWA</th>
        <th colspan=2>NILAI</th>
    </tr>
    <tr bgcolor=red>
        <th width=200>NAMA</th>
        <th width=80>NIM</th>
        <th width=40>JENKEL</th>
        <th>TUGAS</th>
        <th>QUIZ</th>
    </tr>
    <tr>
        <td>1</td>
        <td>Wirasto S. Karim</td>
        <td>01010101</td>

```

```

        <td>Pria</td>
        <td>B</td>
        <td>B</td>
    </tr>
    <tr>
        <td>2</td>
        <td>Abdul Gani</td>
        <td>02020202</td>
        <td>Pria</td>
        <td>C</td>
        <td>A</td>
    </tr>
    <tr>
        <td>3</td>
        <td>Ramona</td>
        <td>03030303</td>
        <td>Wanita</td>
        <td>A</td>
        <td>A</td>
    </tr>
</table>

</body>
</html>

```

### tabel3.html

```

<html>
<head>
    <title>Pengenalan Tabel</title>
</head>
<body>

<table border=1>
    <tr bgcolor=yellow>
        <th width=30>NO</th>
        <th width=200>NAMA</th>
        <th>DATA</th>
    </tr>
    <tr>
        <td>1</td>
        <td valign=top>Wirasto S. Karim</td>
        <td align=center>

```

```

        <table border=1 cellpadding=20>
            <tr>
                <td>A</td>
                <td>B</td>
            </tr>
            <tr>
                <td>C</td>
                <td>D</td>
            </tr>
        </table>
    </td>
</tr>
<tr valign=top>
    <td>2</td>
    <td>Abdul Gani</td>
    <td align=center>
        
    </td>
</tr>
<tr>
    <td>3</td>
    <td>Ramona</td>
    <td>
        <ol>
            <li>HTML</li>
            <li>CSS</li>
            <li>JavaScript</li>
        </ol>
    </td>
</tr>
</table>

</body>
</html>

```

Selain untuk menyajikan data dalam bentuk baris dan kolom, tabel juga bisa digunakan untuk mengatur tata letak dari halaman web.

#### tabelweb.html

```

<html>
<head>
    <title>MUKABUKU</title>

```

```
</head>
<body>

<table border=1 width=100%>
  <tr>
    <td colspan=2 bgcolor=blue><h1>MUKABUKU.COM</h1></td>
  </tr>
  <tr valign=top>
    <td width=150>
      <b>
        <a href=''>Beranda</a><br>
        <a href=''>Profil</a><br>
        <a href=''>Teman</a><br>
        <a href=''>Permainan</a>
      </b>
    </td>

    <td>
      <p>Ini adalah halaman yang menyediakan informasi</p>
    </td>
  </tr>
</table>

</body>
</html>
```

---

## BAB III

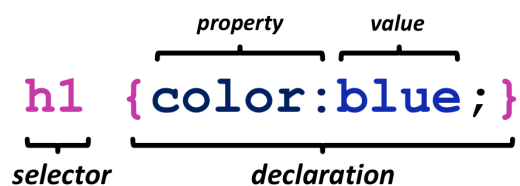
### CSS

#### APA ITU CSS?

Cascading Style Sheet (css), merupakan style yang digunakan untuk mengatur tampilan dari halaman web. Dengan menggunakan css, maka untuk menghasilkan tampilan web yang indah dan menarik jauh lebih mudah dan ringkas dibandingkan menggunakan atribut di tiap tag html.

Selain bisa membuat indah, css juga bisa membantu menciptakan web dengan tampilan yang dinamis dan fleksibel. Baik itu untuk tampilan web di browser ataupun ketika halaman web dicetak lewat printer.

#### STRUKTUR CSS



- Perintah css terdiri atas 2 komponen, yakni Selector dan Declaration
- Selector berfungsi untuk memberi tahu web browser bahwa pada elemen mana aturan css akan diterapkan
- Selector dapat berupa nama tag, selector class atau selector id
- Declaration merupakan aturan css yang diterapkan yang mana terdiri atas property dan value

## **PENULISAN CSS**

Css juga memiliki aturan-aturan penulisan yaitu:

### **Internal Style**

Internal style dengan menggunakan tag style dan diletakkan diantara tag head.

```
<html>
<head>
  <title>CSS</title>
  <style>
    h1 { color : blue; }
  </style>
</head>
<body>
```

Internal style tidak harus selalu diletakkan diantara tag head. Kita selalu bisa meletakkan diantara tag lain yang ada didalam tag body. Namun untuk menciptakan kode yang lebih bersih dan terstruktur dengan baik, css biasa diletakkan diantara tag head.

### **Inline Style**

Inline style dengan menambahkan langsung kode css kedalam tag html sebagai atribut.

```
<h1 style="color:blue;">Judul Satu</h1>
```

### **External Style**

External style dengan menggunakan tag link yang diletakkan diantara tag head dan merujuk keberkas css.

```
<html>
<head>
  <title>CSS</title>
  <link href="style.css" rel="stylesheet">
</head>
<body>
```

## SELECTOR

Seperti sudah disebutkan sebelumnya, selector dapat berupa nama tag, selector class atau selector id. Berikut ini adalah contoh penggunaan selector dalam css.

```
selector.html
<html>
<head>
  <title>Belajar Selector</title>
  <style>
    h1 {
      color: blue;
    }

    .judul {
      color: red;
    }

    #satu {
      color: green;
    }
  </style>
</head>
<body>

<h1>Selector Tag</h1>

<h1 class=judul>Selector Class</h1>

<h1 id=satu>Selector Id</h1>

</body>
</html>
```

## MEMFORMAT DOKUMEN DENGAN CSS

Memformat dokumen dengan css jauh lebih fleksibel dibandingkan dengan menggunakan tag html saja.

```
format.html
<html>
<head>
```

```

<title>Format CSS</title>
<style type=text/css>
    body {
        background-color:yellow;
    }

    h1 {
        color:blue;
        text-align:center;
    }

    b {
        font-weight:normal;
    }

    i {
        font-style:normal;
    }

    u {
        text-decoration:none;
    }

    b, i {
        color:green;
    }
</style>
</head>
<body bgcolor=green>

<h1>DOKUMEN HTML</h1>

<b>Pemrograman Web</b>

<br>

<i>Pemrograman Web</i>

<br>

<u>Pemrograman Web</u>

</body>
</html>

```



## ANCHOR CSS

Kita bisa menjadikan anchor menjadi lebih menarik dengan css.

### link1.html

```
<html>
<head>
  <title>Anchor CSS</title>
  <style type=text/css>
    a {
      color:red;
      text-decoration:none;
    }

    a:hover {
      color:blue;
    }
  </style>
</head>
<body>

<h1>Halaman Utama</h1>

<a href="">Ke Halaman 1</a>
<a href="">Ke Halaman 2</a>

</body>
</html>
```

---

### link2.html

```
<html>
<head>
  <title>Anchor CSS</title>
  <style type=text/css>
    a {
      color:red;
      text-decoration:none;
      border:1px solid red;
      background:yellow;
      padding:2px;
    }

    a:hover {
```

```

        color:blue;
    }

    .facebook {
        background:#3b5998;
        color:white;
        border:1px solid #d8dfea;
        font-size: 11px;
        font-family: "lucida grande", tahoma, verdana, arial, sans-
serif;
    }

    .facebook:hover {
        color:white;
    }
</style>
</head>
<body>

<h1>Halaman Utama</h1>

<a href="">Ke Halaman 1</a>
<a href="" class=facebook>Ke Halaman 2</a>

</body>
</html>

```

---

## TATA LETAK

Sudah pernah dipelajari di bab sebelumnya bagaimana mengatur tata letak dengan menggunakan tabel. Sekarang dengan bantuan css, kita tidak hanya bisa menjadikan tata letak dengan tabel lebih menarik, namun juga bisa menggunakan divisi sebagai pengganti tabel. Divisi jauh lebih fleksibel dibandingkan tabel karena tidak terikat oleh susunan baris dan kolom.

## Tabel

Contoh untuk tabel kita menggunakan external style bernama tabel.css

```
tabel.css

body {
    margin:0px;
}

table {
    border:0px;
    width:100%;
    height:100%;
}

.judul {
    font-size:25px;
    background-color:#4C8CE7;
    font-weight:bold;
    color:white;
    padding:5px;
    border-bottom:1px solid red;
    height:30px;
}

.menu {
    width:150px;
}

a {
    text-decoration:none;
    color:#4C8CE7;
}

.isi {
    font-style:italic;
    border:1px solid red;
}

#kanan {
    text-align:right;
    color:green;
}
```

## tabel.html

```
<html>
<head>
  <title>MUKABUKU</title>
  <link href="tabel.css" rel="stylesheet">
</head>
<body>

<table>
  <tr>
    <td colspan=2 class=judul>MUKABUKU.COM</td>
  </tr>
  <tr valign=top>
    <td class=menu>
      <a href=''>Beranda</a><br>
      <a href=''>Profil</a><br>
      <a href=''>Teman</a><br>
      <a href=''>Permainan</a>
    </td>

    <td class=isi>
      <p>Ini adalah halaman yang menyediakan informasi.</p>
      <p id=kanan>Ini adalah halaman yang menyediakan
informasi.</p>
    </td>
  </tr>
</table>

</body>
</html>
```

## Divisi

Divisi dibuat dengan menggunakan tag div

## div1.html

```
<html>
<head>
  <title>Divisi</title>
  <style type="text/css">
    .div1, .div2, .div3 {
      width:150px;
      height:150px;
```

```

        position:absolute;
        border:1px solid black;
    }

    .div1 {
        background-color:red;
        top:50px;
        left:50px;
    }

    .div2 {
        background-color:blue;
        top:100px;
        left:100px;
    }

    .div3 {
        background-color:green;
        top:150px;
        left:150px;
    }

    .div3:hover {
        background-color:yellow;
    }
</style>
</head>
<body>

<div class=div1></div>

<div class=div2></div>

<div class=div3></div>

</body>
</html>

```

---

#### div2.html

```

<html>
<head>
    <title>Divisi</title>

```

```
<style type="text/css">
    body {
        margin:0;
    }

    .div1 {
        width:100%;
        height:50px;
        background-color:red;
    }

    .div2 {
        float:left;
        width:120px;
    }

    .div3 {
        float:left;
    }
</style>
</head>
<body>

<div class=div1>
MUKABUKU.COM
</div>

<div class=div2>
    <a href=''>Beranda</a><br>
    <a href=''>Profil</a><br>
    <a href=''>Teman</a><br>
    <a href=''>Permainan</a>
</div>
<div class=div3>
    Ini adalah halaman yang menyediakan informasi.
</div>
</body>
</html>
```

---

# BAB IV

## JAVASCRIPT

### APA ITU JAVASCRIPT?

Javascript merupakan bahasa scripting yang didisain untuk membuat halaman web menjadi lebih interaktif. Javascript bekerja secara client-side (disisi client). Sehingga javascript bergantung pada browser yang digunakan untuk menampilkan halaman web yang mengandung javascript.

Perlu diperhatikan bahwa Javascript berbeda dengan Java. Keduanya sangat berbeda baik itu dari sisi bahasa maupun dari konsep dan disainnya.

```
index.html
```

```
<html>
<head>
  <title>Belajar JavaScript</title>
  <script language=javascript>
    document.write("<h1>Belajar JavaScript</h1>");
  </script>
</head>
<body>

</body>
</html>
```

### PENULISAN JAVASCRIPT

Aturan penulisan javascript yaitu:

#### Internal Script

Internal script dengan menggunakan tag script dan diletakkan diantara tag head

```
<html>
<head>
  <title>JavaScript</title>
  <script language="javascript">
```

```
        document.write("<h1>Belajar JavaScript</h1>");
    </script>
</head>
<body>
```

## External Script

External script tetap dengan menggunakan tag script namun merujuk keberkas javascript

```
<html>
<head>
    <title>JavaScript</title>
    <script language="javascript" src="script.js"></script>
</head>
<body>
```

## VARIABEL

Dalam javascript, variabel dideklarasikan dengan statement var dan diakhiri dengan tanda ; (titik koma). Variabel juga bersifat case-sensitive yang artinya membedakan huruf kecil dan huruf besar. Sehingga variabel nama berbeda dengan variabel NAMA.

### var.html

```
<html>
<head>
    <title>Belajar JavaScript</title>
    <script language=javascript>
        var a = 4;
        var b = 10;
        var c = a + b;
        var d = "<br>Bo gambangi da'a ei";

        document.write(c);
        document.write(d);
    </script>
</head>
<body>

</body>
</html>
```



## KONDISI

Kondisi digunakan untuk melakukan eksekusi suatu pernyataan secara bersyarat.

### kondisi1.html

```
<html>
<head>
  <title>Belajar JavaScript</title>
  <script language=javascript>
    var nama = "wirasto";

    if (nama == "wirasto") {
      document.write("Benar");
    }
  </script>
</head>
<body>

</body>
</html>
```

### kondisi2.html

```
<html>
<head>
  <title>Belajar JavaScript</title>
  <script language=javascript>
    var awal = 1;
    var akhir = 5;

    if (awal < akhir) {
      document.write("Nilai awal lebih kecil dari nilai akhir");
    } else {
      document.write("Nilai awal TIDAK lebih kecil dari nilai
akhir");
    }
  </script>
</head>
<body>

</body>
</html>
```

## kondisi3.html

```
<html>
<head>
  <title>Belajar JavaScript</title>
  <script language=javascript>
    var awal = 1;
    var akhir = 5;

    if (awal < akhir) {
      document.write("Nilai awal lebih kecil dari nilai akhir");
    } else {
      document.write("Nilai awal TIDAK lebih kecil dari nilai
akhir");
    }
  </script>
</head>
<body>

</body>
</html>
```

---

## PERULANGAN

Perulangan digunakan untuk mengeksekusi satu atau beberapa baris pernyataan secara berulang-ulang selama syaratnya terpenuhi.

### 1. Do...While

#### dowhile.html

```
<html>
<head>
  <title>Belajar JavaScript</title>
  <script language=javascript>
    var awal = 1;
    var akhir = 5;
    var i = awal;

    do {
      document.write("Cetakan ke " + i + "<br>");
      i++;
    } while (i <= akhir);
```

```
        </script>
</head>
<body>

</body>
</html>
```

---

## 2. While

while.html

```
<html>
<head>
  <title>Belajar JavaScript</title>
  <script language=javascript>
    var awal = 1;
    var akhir = 5;
    var i = awal;

    while (i <= akhir) {
      document.write("Cetakan ke " + i + "<br>");
      i++;
    }
  </script>
</head>
<body>

</body>
</html>
```

---

## 3. For

for.html

```
<html>
<head>
  <title>Belajar JavaScript</title>
  <script language=javascript>
    var awal = 1;
    var akhir = 5;

    for (i = awal; i <= akhir; i++) {
      document.write("Cetakan ke " + i + "<br>");
    }
  </script>
</head>
<body>

</body>
</html>
```

```
    </script>
</head>
<body>

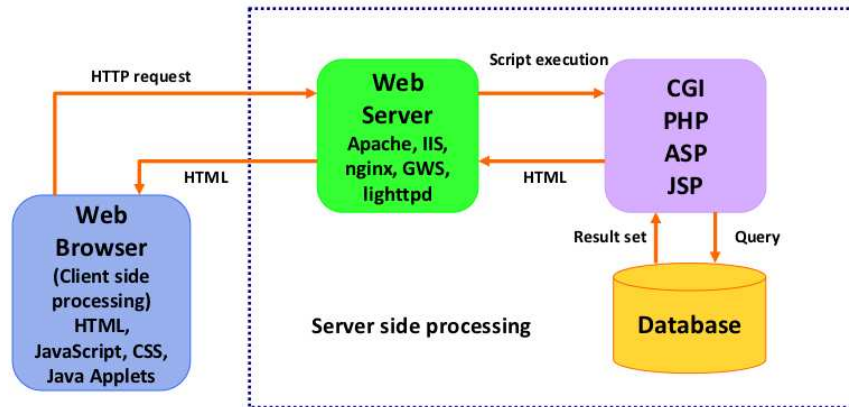
</body>
</html>
```

---

# BAB V

## SERVER SIDE SCRIPTING

### SKEMA SERVER SIDE SCRIPTING



Sumber: Slide Perkuliahan Teknik Informatika UNPAS<sup>[4]</sup>

### WEB SERVER

Untuk aplikasi web yang bekerja disisi server (server side), dibutuhkan adanya web server untuk mengirimkan hasil proses berupa halaman web ke client. Dimana halaman web ini dikirim menggunakan protokol http. Web server pulalah yang menerima permintaan dari client.

Web server juga berfungsi untuk memetakan lokasi-lokasi berkas aplikasi web dalam komputer server agar bisa diakses oleh client. Termasuk mengatur apa web bisa diakses dari jaringan global atau hanya jaringan tertentu saja yang diijinkan untuk mengaksesnya.

Beberapa aplikasi web server yang populer:

- Apache
- Ngix
- IIS
- lighttpd

## **WEB PROGRAMMING LANGUAGE**

Bahasa pemrograman web berfungsi untuk menghasilkan halaman web yang dinamis. Dinamis tidak hanya dari sisi tampilan, namun juga informasi yang diambil dari database.

Bahasa pemrograman untuk web sering juga disebut sebagai bahasa scripting. Karena berupa kumpulan script yang dijalankan oleh interpreter. Makanya aplikasi-aplikasi web yang ada tidak melewati proses kompilasi terlebih dahulu sebelum digunakan.

Beberapa bahasa pemrograman web yang populer:

- PHP
- ASP
- Perl
- Python
- Java

Bahasa pemrograman yang telah digunakan untuk membuat web yang akan dipublikasikan, harus diaktifkan dulu dukungannya di sisi web server. Kalau tidak, web server tidak akan bisa mengenali dan menampilkan web yang sudah dibuat.

## **DBMS**

Database Manajemen System (DBMS) merupakan paket perangkat lunak yang dirancang khusus untuk membuat, memelihara, mengontrol, menyimpan serta memberikan akses secara terkontrol terhadap data.

DBMS biasanya juga menyediakan bahasa tersendiri yang bisa digunakan untuk berinteraksi dengannya. Namun hampir semua dbms populer menyediakan dukungan pula untuk bahasa SQL.

Beberapa aplikasi dbms yang populer:

- MySQL
- PostgreSQL
- Microsoft SQL Server

# BAB VI

## DASAR PHP

### PHP

PHP merupakan salah satu bahasa pemrograman untuk web yang bekerja disisi server. Hampir semua website dinamis yang bisa diakses lewat internet dibangun dengan menggunakan bahasa ini. Sampai saat ini php merupakan satu-satunya bahasa yang menawarkan kemudahan dalam mengembangkan aplikasi internet, terutama dengan dukungan luar biasa ke berbagai database server.

Dalam hal sintaks, sintaks php sangat mirip dengan sintaks bahasa C. Karena memang bahasa ini beserta ekstensionnya juga dibuat dengan menggunakan bahasa C. Sehingga bagi para pengguna bahasa tersebut, akan lebih mudah untuk pindah atau mempelajari php.

### Keunggulan PHP

- Cepat, karena disisipkan diantara tag html maka waktu proses dan generate halaman web bisa dilakukan dengan singkat
- Gratis, untuk mendapatkan interpreter php cukup dengan mendownloadnya di internet
- Mudah digunakan, sintaks sederhana dan mirip dengan banyak bahasa pemrograman populer
- Serba guna, dapat dijalankan hampir di semua sistem operasi
- Dukungan teknis yang luas
- Aman, selama website didesain dengan benar, pengguna tidak dapat melihat kode php yang dibuat
- Bisa dimodifikasi, dengan lisensi Open Source, programmer dapat memodifikasi kode sumber php untuk disesuaikan dengan kebutuhan

### Contoh Kode

```
index.php
<?php
echo "Kita ini kode PHP";
?>
```

## Ciri-Ciri PHP

- Ekstensi berkas berupa .php
- Case-sensitive
- Tiap akhir perintah diakhiri dengan tanda ;
- Diawali dengan tanda <?php dan diakhiri dengan tanda ?>

## Menjalankan Kode PHP

Untuk bisa menjalankan kode php untuk aplikasi web, dibutuhkan adanya interpreter php yang sudah terinstall dan juga web server. Setiap berkas php yang dibuat harus disimpan dalam document root dari web server yang digunakan. Untuk web server apache di beberapa distribusi linux, menempatkan document rootnya di /var/www.

Untuk mencoba apakah web server sudah bekerja dengan baik dikomputer yang digunakan adalah dengan menghubungi URL <http://localhost> atau <http://127.0.0.1> lewat web browser.

## VARIABEL

Dalam php, variabel diawali dengan tanda \$ (dolar) dan tidak perlu ditentukan tipe datanya terlebih dahulu. Misalnya variabel nama ditulis menjadi \$nama. Interpreter php akan mendeteksinya secara otomatis tipe data dari variabel yang dibuat dan akan melakukan konversi tipe data secara otomatis jika dibutuhkan.

```
var.php
<?php
$a = 4;
$b = 10;
$c = $a + $b;
$d = "Bo gampang ei";
$e = true;

echo "$c <br>";
echo "$d <br>";
echo "$e <br>";
?>
```



## OPERATOR

- **Operator Assigment**

Misalkan  $x=6$  dan  $y=9$

Operator	Contoh	Sama Dengan	Hasil
=	$\$nama = \text{“Wirasto S. Karim”}$		Wirasto S. Karim
+=	$\$x += 4$	$\$x = \$x + 4$	10
-=	$\$y -= \$x$	$\$y = \$y - \$x$	3
*=	$\$x *= \$y$	$\$x = \$x * \$y$	54
.=	$\$y .= \$x$	$\$y = \$y . \$x$	96

- **Operator Numerik**

Misalkan  $x=10$

Operator	Deskripsi	Contoh	Hasil
+	Penjumlahan	$\$x = 1 + 2$	3
-	Pengurangan	$\$x = \$x - 1$	9
*	Perkalian	$\$x = 4 * \$x$	40
/	Pembagian	$\$x = \$x / 2$	5
%	Modulus	$\$x = \$x \% 2$	0
++	Increment (Naik 1)	$\$x++$	11
--	Decrement (Turun 1)	$\$x--$	9

- **Operator String**

Operator	Deskripsi	Contoh	Hasil
.	Menggabungkan String	$\$x = \text{“Wirasto”} . \text{“Karim”}$	WirastoKarim

- **Operator Pembanding**

Misalkan  $x=69$

Operator	Deskripsi	Contoh	Hasil
==	Sama dengan	$\$x == 69$	True
!=	Tidak sama dengan	$\$x != 69$	False
>	Lebih besar	$\$x > 69$	False
<	Lebih kecil	$\$x < 100$	True
>=	Lebih besar atau sama dengan	$\$x >= 69$	True

<=	Lebih kecil atau sama dengan	\$x <= 69	True
----	------------------------------	-----------	------

- **Operator Logika**

Misalkan \$x=6 dan \$y=9

Operator	Deskripsi	Contoh	Hasil
&&	Dan (and)	(\$x < 10 && \$y > 1)	True
	Atau (or)	(\$x == 1    \$y == 9)	True
!	Bukan (not)	!(\$x == \$y)	True

## KONDISI

Kondisi digunakan untuk melakukan eksekusi suatu pernyataan secara bersyarat. Jika syaratnya terpenuhi barulah pernyataan akan dieksekusi, jika syarat tidak terpenuhi, maka pernyataan tidak akan dieksekusi (dijalankan).

### kondisi1.php

```
<?php
$name = "wirasto";

if ($name == "wirasto") {
    echo "Benar";
}
?>
```

### kondisi2.php

```
<?php
$a = 1;
$b = 5;

if ($a < $b) {
    echo "Nilai $a lebih kecil dari nilai $b";
} else {
    echo "Nilai $a TIDAK lebih kecil dari nilai $b";
}
?>
```

### kondisi3.php

```
<?php
$awal = 1;
$akhir = 5;

if ($awal == 2) {
    echo "Nilai $awal adalah 2";
} else if ($akhir == 5) {
    echo "Nilai $akhir adalah 5";
} else {
    echo "Tidak ada yang benar";
}
?>
```

## PERULANGAN

Perulangan digunakan untuk mengeksekusi satu atau beberapa baris pernyataan secara berulang-ulang selama syaratnya terpenuhi.

### 1. Do...While

#### dowhile.php

```
<?php
$awal = 1;
$akhir = 5;
$i = $awal;

do {
    echo "Cetakan ke $i\n";
    $i++;
} while ($i <= $akhir);
?>
```

### 2. While

#### while.php

```
<?php
$awal = 1;
$akhir = 5;
$i = $awal;
```

```
while ($i <= $akhir) {
    echo "Cetakan ke $i\n";
    $i++;
}
?>
```

---

### 3. For

for.php

```
<?php
$aawal = 1;
$akhir = 5;

for ($i = $awal; $i <= $akhir; $i++) {
    echo "Cetakan ke $i\n";
}
?>
```

---

## ARRAY

Array digunakan untuk menampung kelompok data kedalam satu variabel yang sama.

Variabel array akan terdiri dari key (kunci) dan value (nilai)

```
$nama_variabel[kunci] = nilai
```

array1.php

```
<?php
$mhs=array();

$mhs[1] = "Wati";
$mhs[2] = "Budi";
$mhs[5] = "Iwan";

echo "$mhs[1]<br>";
echo "$mhs[2]<br>";
echo "$mhs[5]<br>";
?>
```

---

Jika kunci tidak disebutkan maka secara otomatis kunci akan dimulai dari 0 dan akan diurutkan.

#### array2.php

```
<?php
$mhs=array();

$mhs[] = "Wati";
$mhs[] = "Budi";
$mhs[] = "Iwan";

echo "$mhs[0]<br>";
echo "$mhs[1]<br>";
echo "$mhs[2]<br>";
?>
```

#### array3.php

```
<?php
$mhs=array("Wati", "Budi", "Iwan");

echo "$mhs[0]<br>";
echo "$mhs[1]<br>";
echo "$mhs[2]<br>";
?>
```

## FUNCTION

PHP secara default sudah datang dengan begitu banyak built-in function (fungsi bawaan), seperti fungsi-fungsi untuk pemroses teks, tanggal, matematika, database, berkas, dan masih banyak lagi.

### 1. strlen

Digunakan untuk mendapatkan panjang string

Contoh:

```
strlen("Wirasto"); ---> Hasilnya: 7
```

```
strlen("A B C"); ---> Hasilnya: 5
```

## 2. substr

Digunakan untuk mengambil sebagian string berdasarkan index string. Index adalah nomor posisi huruf dalam string. Index dimulai dari 0 untuk posisi pertama dari string.

Contoh:

```
substr("Wirasto", 2); ---> Hasilnya: rasto
```

```
substr("Wirasto", 3, 2); ---> Hasilnya: as
```

Selain fungsi-fungsi yang sudah disediakan, kita juga bisa membuat fungsi sendiri. Ini biasa disebut sebagai UDF (user defined function). Tujuan pembuatan fungsi sendiri adalah untuk mengelompokkan sejumlah pernyataan php, sehingga bisa mempersingkat pengkodean ketika pernyataan digunakan beberapa kali dalam program.

### fungsi.php

```
<?php
function tambah($a, $b)
{
    $c = $a + $b;
    echo "Hasil Penambahan $a + $b adalah $c<br>";
}

tambah(2, 3);

tambah(6, 7);
?>
```

---

Fungsi buatan sendiri juga dapat dibuat untuk bisa mengembalikan nilai

### fungsi2.php

```
<?php
function tambah($a, $b)
{
    $c = $a + $b;
    return $c;
}

$hasil=tambah(2, 3);
echo "Hasil Penambahan : $hasil<br>";
?>
```

---

## PUSTAKA

Betha, Husni Iskandar Pohan, *Pemrograman Web Dengan HTML*, INFORMATIKA, Bandung 2002

PHP Documentation Group, *PHP Manual*, <http://php.net>, 2012