

Tutorial Cascading Style Sheet (CSS)

Abe Poetra

abe_poetra@yahoo.com

abe@go.fmipa.usu.ac.id

Lisensi Dokumen:

Copyright © 2003 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

(* Artikel ini juga telah dimuat di <http://medan.linux.or.id>

Kata-kata Introduksi

Pernah dengar istilah di atas? (bukan, bukan yang "Kata-kata Introduksi" tapi di atasnya lagi, nah bener yang itu, "Cascading Style Sheet" atau sering disebut CSS). Mungkin sebahagian besar akan menjawab "YA". Pun, kalau semisal belum sekarang jadi sudah. Karena Cascading Style Sheet adalah suatu teknologi yang digunakan untuk memperindah tampilan halaman website (situs). Singkatnya dengan menggunakan Metode CSS ini anda dengan mudah mengubah secara keseluruhan warna dan tampilan yang ada di situs anda, sekaligus memformat ulang situs anda (merubah secara cepat).

Jelas, dengan sedikit berfikir dan menggunakan logika maka anda akan mendapatkan hasil yang memuaskan, maka sedari itu baca semua tutorial ini sampai habis, dicoba langsung, kemudian jika bisa dikembangkan, sehingga kalau misalnya anda ingin sedikit memodifikasikan kode yang telah dibuat di dalamnya sangatlah gampang, apalagi bla...bla...bla... dan bla..bla...(wah... bakat ceramah saya tiba2 kambuh (") ma'af) ya udah pokoknya CSS itu digunakan untuk *memperindah tampilan situs*, terserah anda mo mendefinisikannya bagaimana? (lihat definisi saya yang di Italic-kan).

Bagi anda yang baru dalam bidang ini jangan khawatir, karena memang tutorial ini sangat mendasar, tetapi setidaknya sudah menguasai dasar-dasar HTML. Dan bagi anda yang sudah mahir jangan ngejek dunks, toh, saya sudah ngaku ini tutorial untuk yang "pemula saja". Hehehe... :-)
peace, bro's.

Ya udah, Baca aja deh... syukur-syukur nambah pengetahuan atawa temen dikala gak bisa tidur, gitu baca tutorial ini langsung (Zzz... Zzz... Zz... grok.. grok...) tertidur deh hehehe... kelihatannya yang belakangan disebut yang paling sering dipilih.

Anggap ini Mudah

Sekarang coba kita mulai dengan contoh-contoh yang ringan aja dulu (dalem hati : syukur-syukur gak tau

kalau ini alih-alih karena gak mampu buat yang ribet)

Note: Kalau misalnya malas menyalin kode HTML nya, gunakan tehnik 2 jurus yang sangat ampuh. Ya, bener "kopi dan paste" bukan kopi dan teh manis ya!

```
<HTML>
<HEAD>
</HEAD>
<BODY>
P: Mengapa jika kita anggap <B>suatu pekerjaan</B> itu mudah
   maka pekerjaan itu akan beneran menjadi<B> lebih mudah</B>?
<P>
J: Karena itu merupakan <B>sugesti</B> terhadap <B>diri kita
   sendiri</B>
</BODY>
</HTML>
```

Nah, seperti yang anda lihat beberapa huruf pada kalimat-kalimat di atas menjadi BOLD (ditebalin), karena fungsi dari tag `` dan `` adalah untuk menebalkan huruf. Lihat kembali catatan-catatan atau tutorial-tutorial untuk membuat HTML anda. Sekarang coba bayangkan bagaimana jika kita ingin membuat warna pada huruf-huruf yang ditebalkan itu? Katakan saja kita ingin mewarnainya dengan hijau? Mungkin.

Kebanyakan kita akan buat listing kodenya seperti ini :

```
<HTML>
<HEAD>
</HEAD>
<BODY>
P: Mengapa jika kita anggap <B><FONT COLOR="Lime">suatu
   pekerjaan</FONT></B> itu mudah maka pekerjaan itu akan
   beneran menjadi<B><FONT COLOR="Lime"> lebih mudah
   </FONT></B>?
<P>
J: Karena itu merupakan <B><FONT COLOR="Lime">sugesti
   </FONT></B> terhadap <B><FONT COLOR="Lime">diri kita
   sendiri </FONT></B>
</BODY>
</HTML>
```

Bagaimana hasilnya di browser? Huruf yang ditebalin sekarang sudah berwarna "hijau", sekarang coba bandingkan kemudahan yang didapat jika kita menggunakan Style-Sheet.

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    B { color : lime }
  </STYLE>
</HEAD>
<BODY>
P: Mengapa jika kita anggap <B>suatu pekerjaan</B> itu mudah
   maka pekerjaan itu akan beneran menjadi<B> lebih mudah</B> ?
<P>
J: Karena itu merupakan <B>sugesti</B> terhadap <B>diri kita
```

```
sendiri</B>
</BODY>
</HTML>
```

Kedua contoh di atas terlihat persis sama di browser, jadi apa untungnya buat kita? Hehehe... masih belum ngerti juga? Jika anda menggunakan style-sheet maka listing code anda setidaknya akan lebih mudah dibaca dan di atur. Keuntungan yang kedua, kita dapat melakukan ini dimana saja dan kapan saja. Ini akan anda pahami setelah anda berpindah ke tahap selanjutnya yang lebih berbobot. (wah salut... berarti ini masih "piece of cake" khan bro?).

Coba-coba? Silahkan!

Eh... tiba-tiba saja terfikir untuk menggantikan warna dengan kode RGB nya, layaknya perintah-perintah HTML dasar. bagaimana? Bisa tidak ya? Seperti terlihat di bawah ini kita cukup menggantikannya saja untuk mendapatkan variasi warna.

```
<STYLE TYPE="text/css">
  B { color : #CC6633 }
</STYLE>
```

Bagaimana? Sama mudahnya bukan? Bagi anda yang sudah familiar dengan beberapa perintah HTML, maka beberapa modifikasi dapat anda buat misalnya seperti contoh di bawah :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    I { color : red }
  </STYLE>
</HEAD>

<BODY>
P: Mengapa jika kita anggap <I>suatu pekerjaan</I> itu mudah
  maka pekerjaan itu akan beneran menjadi<I> lebih mudah</I>?
<P>
J: Karena itu merupakan <I>sugesti</I> terhadap <I>diri kita
  sendiri</I>
</BODY>
</HTML>
```

Anda lihat saya mengganti huruf "B" dengan Huruf "I", sehingga jika menurut kebiasaan browser (berlebihan tidak bahasanya?) akan menganggap baris yang diberi tag tersebut menjadi ITALIC. Sekarang lihat hasilnya pada browser. Timbul pertanyaan bagaimana jika untuk UNDERLINE atawa garis di bawah huruf? Ya benar, hal ini juga masih berlaku.

Nah.. sekarang saya ingin mencoba sedikit pemahaman anda ☺ !

Jika saya ingin kata "*suatu pekerjaan*" dari contoh di atas menjadi berwarna merah dan ITALIC, terus kata "*lebih mudah*" kita buat menjadi berwarna hijau dan BOLD, terus untuk menambah kebingungan anda jadikan kata "*sugesti*" dengan UNDERLINE dan berwarna merah, serta kata "*diri kita sendiri*" menjadi hijau dan BOLD.

Saya kehendaki tampilannya seperti itu pada browser bagaimana? ehm.. menurut anda, membuat kode HTML nya? Susah? Wah... payah deh (",) ---> ini senyum, tapi kalau mau yang lebih mantab lagi seperti ini c",) --> isn't it cool, huh? (ini macemnya gak ada hubungannya ya? Tapi cuek aja deh, wong ini

tutorial saya yang bikin kok, kalau anda yang buat terserah anda. Kan adil? hehehe)

Coba seperti ini :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    I,U { color: red }
    B { color : green }
  </STYLE>
</HEAD>

<BODY>
P: Mengapa jika kita anggap <I>suatu pekerjaan</I> itu mudah
   maka pekerjaan itu akan beneran menjadi<B> lebih mudah</B>?
<P>
J: Karena itu merupakan <U>sugesti</U> terhadap <B>diri kita
   sendiri</B>
</BODY>
</HTML>
```

Anda lihat saya hanya mengubah beberapa huruf serta menambah beberapa perintah saja, dan hasilnya wow... mengagumkan.

Sekarang sudah jelas mengapa saya katakan css dapat menghemat kode dan waktu anda dalam mendesain tampilan situs anda. Jika suatu kali anda bosan dengan warna atau bentuk tulisan pada tampilan situs anda, hanya dengan mengubah dan menambah variasi dari perintah css anda akan mendapatkan tampilan yang lebih menarik bukan? Dan juga benar-benar menghemat waktu.

Ketentuan Yang Mendasar

Sampai saat ini anda sudah dapat melihat kehebatan dan kemampuan yang dimiliki Style Sheet, mari kita kembali berkuat dikubangan untuk mengingat aturan dasar yang ada dalam css ini. Pada contoh di atas, baris ini!

```
B {color: lime}
```

dikenal sebagai "Style Rule" atau peraturan/perintah css, yang mana terdiri dari dua elemen dasar yaitu : "selector" dan "declaration"

sebuah "selector" biasanya adalah tag HTML (dalam hal ini B), sementara "declaration" adalah satu atau beberapa perintah/nilai dari css yang menunjukkan type bentuk yang diaplikasikan pada selector. Declaration ini biasanya di tandai dengan kurung kurawal, dan perintah/nilai css yang berbeda dipisahkan satu dengan yang lain dengan menggunakan titik-koma, Pasti bingung? Lihat contoh!

```
B {color: lime; text-decoration: underline; font-family: Arial}
```

ini baru dapat kita katakan sebagai css yang valid. Rumit? Belum tentu, anda jangan memvonis dulu sebelum disidangkan (lho... apa hubungannya ya?). Lengkapnya :

```
<HTML>
<HEAD>
<STYLE TYPE="text/css">
B {color: lime; text-decoration: underline; font-family: Arial}
</STYLE>
</HEAD>

<BODY>
P: Mengapa jika kita anggap <B>suatu pekerjaan</B> itu mudah
    maka pekerjaan itu akan beneran menjadi<B> lebih mudah</B>?
<P>
J: Karena itu merupakan <B>sugesti</B> terhadap <B>diri kita
    sendiri</B>
</BODY>
</HTML>
```

css sendiri memiliki lebih dari enam-puluh keywords (kata-kunci), dan anda akan sering-sering bergaul dengan mereka seiring dengan pemahaman tutorial ini.

Selectors juga dapat di kelompokkan, seperti contoh di bawah, yang mengubah teks untuk H1, H2, dan H4, menjadi berwarna putih.

```
H1, H2, H3 {color: white} /* this rule turns H1, H2 and H3 blocks white */
```

Sebagai catatan, anda juga dapat memasukkan komentar didalam blok komentar `/*` dan `*/`.

Hal yang paling umum dalam memasukkan kode Style Sheet dengan menggunakan tag `<STYLE>`, tag `<STYLE>` ini selalu tampil dalam bagian `<HEAD>` dari dokumen anda, garis besarnya seperti ini :

```
<HEAD>
  <STYLE TYPE="text/css">
    ... aturan-aturan css disini ...
  </STYLE>
</HEAD>
```

Jika anda lihat ulang contoh-contoh sebelumnya, maka akan terlihat beginilah garis besar dari metode yang digunakan. TAPI INI HANYA BERLAKU UNTUK DOKUMEN INI SAJA (lihat pakai huruf kapital untuk menegaskan hehehe...), yang menimbulkan sedikit masalah jika kamu ingin menerapkan css untuk keseluruhan halaman web. Tidak perlu khawatir... pemecahannya semudah membalikkan beras segoni (lho? Susah ya!), saya jadi teringat waktu ibu saya nyuruh saya beli beras ke tokonya pak bedoel, beras udah saya bayar eh.. pak bedoel ngotot uangnya kurang. Padahal tau nggak saya ngasih udah pas, malah... hehehe.. tuh khan udah mulai ngelantur ("), dasar payah aku ya?

Sekali Tulis, Pakai Bersama

Satu dari beberapa kelebihan teknologi css ini - dan merupakan alasan banyak orang menyukai penggunaannya - adalah memperbolehkan kita untuk mendefinisikan sebuah style-sheet global yang dapat menerapkan aturan-aturan css tersebut untuk keseluruhan dokumen-dokumen HTML pada halaman web site anda.

Keuntungannya jelas sekali, jika kita ingin mengubah tampilan situs, kita tinggal mengedit satu file saja

dan hasilnya akan di refleksikan pada keseluruhan dokumen HTML yang dilink ke situ.

Anggap saja anda mempunyai Style-Sheet yang diberi nama "global.css" yang diletakkan pada server "www.situskamu.com/global.css". Untuk memberikan pengaruh aturan Style-Sheet kesemua atau ke spesifik dokumen HTML, gampangnya tambahkan saja kode sederhana ini di dalam tag <HEAD> pada dokumen HTML anda, seperti ini :

```
<HEAD>
  <LINK REL="stylesheet" HREF="http://www.situskamu.com/global.css"
    TYPE="text/css">
</HEAD>
```

dan semua aturan Style-Sheet dalam "global.css" secara otomatis akan diterapkan kedalam dokumen HTML yang ingin diberi Style-Sheet.

Kamu juga dapat mengimport Style-Sheet dengan menggunakan keyword (kata-kunci) "*@import*", mau diberi contoh? Nih coba :

```
<STYLE TYPE="text/css">
  @import url(http://www.situskamu.com/global.css);
  P {color: yellow}
</STYLE>
```

kamu juga dapat menerapkan aturan CSS berdasarkan "case to case", dengan menambahkan atribut CSS ke dalam kode HTML itu sendiri. Contohnya :

```
<HTML>
<HEAD>
</HEAD>

<BODY>
P: Mengapa jika kita anggap <B STYLE="color: lime; background-
  color: black"> suatu pekerjaan</B> itu mudah
  maka pekerjaan itu akan beneran menjadi <B STYLE="color:
  lime; background-color: black"> lebih mudah</B>?
<P>
J: Karena itu merupakan <B STYLE="color: lime; background-
  color: black"> sugesti</B> terhadap <B STYLE="color: lime;
  background-color: black"> diri kita
  sendiri</B>
</BODY>
</HTML>
```

bisakan? Nah sekarang baru percaya.

Lebih lanjut tentang Pseudo-Giberish

Melanjutkan kembali konsep dasar yang sudah dipelajari : pewarisan (inheritance), kelas (classes), pseudo-classes, dan selectors-kontekstual (Contextual Selector). Coba kita bahas satu persatu :

Pewarisan (Inheritance) :

Pewarisan pada dasarnya berarti bagian-bagian HTML yang kekurangan untuk spesifikasi aturan CSS dapat mengikuti/dibuat menurut aturan CSS yang menutupinya (@#\$\$%... pasti bingungkhan dengan penjabarannya? Sama, saya juga gak ngerti apa maksudnya ☺ tulisan ini). Pokoknya seperti inilah, setiap aturan yang tidak ada pada HTML atau ingin menggantikannya dengan aturan CSS, tetap akan mempengaruhi huruf yang ada didalam kurungan CSS (waduh... kok kayaknya makin ribet aja sih? Hehehe... dasar gak professional). Udah coba aja lihat contohnya mana tau aja ngerti? Tapi janji kalo' udah ngerti kasih tau saya (lho.. dasar edan!) :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    B { color : green }
  </STYLE>
</HEAD>

<BODY>
<B>
P: Mengapa jika kita anggap <FONT SIZE="+1">suatu
  pekerjaan</FONT> itu mudah
  maka pekerjaan itu akan beneran menjadi lebih mudah?
<P>
J: Karena itu merupakan sugesti terhadap diri kita
  sendiri</B>
</BODY>
</HTML>
```

Nah, sekarang baru ngertikan? Tag mendefinisikan warna dan menebalkan huruf dengan aturan CSS, sedangkan tag yang notabene berada didalam lingkup CSS dapat menerapkan aturan HTML dasar tetapi tetap terpengaruh dengan kondisi warna dan tebalnya huruf. Sesekali ini juga dapat digunakan sebagai kombinasi sesuai dengan keperluan anda sendiri. (hmm... kali ini lebih jumawa dan berwibawa sekali ya?)

Classes :

CSS juga mengijinkan kita untuk menyatukan elemen-elemen secara bersamaan didalam sebuah kelas (classes) dan menerapkannya aturan Style-Sheet kedalam sebuah class. Sebagai contoh :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .tanya {color: red}
    .jawab {color: blue}
  </STYLE>
</HEAD>

<BODY>
<P CLASS="tanya">
P: Mengapa jika kita anggap suatu pekerjaan itu mudah
  maka pekerjaan itu akan beneran menjadi lebih mudah?
<P CLASS="jawab">
J: Karena itu merupakan <FONT CLASS="tanya">sugesti</FONT>
  terhadap diri kita sendiri
</BODY>
</HTML>
```

Kali ini kita mendefinisikan dua kelas yaitu "tanya" dan "jawab" dengan mengaplikasikan aturan CSS. Dari sini kita bisa lihat kalau kedua kelas itu tidak harus sama dan bisa di modifikasi sesuai dengan kepentingan keinginan dan kebutuhan kita (udah berapa kali kata-kata ini diulang, tapi lagi-lagi cuek aja deh). Yang terpenting sekarang setidaknya kita sudah bisa membuat sebuah kelas.

Selektor Kontekstual (Contextual-Selector)

Selektor Kontekstual hampir mirip dengan pernyataan-pernyataan kondisional - deklarasi Style-Sheet yang ada akan berpengaruh jika kondisi/keadaan tertentu dijumpai. Sebagai contoh, anggap anda ingin membuat semua teks yang bold dan italic tampil dengan huruf arial dan berwarna biru. Selektor-Kontekstual nya akan seperti ini :

```
<HEAD>
  <STYLE TYPE="text/css">
    B I {color: blue; font-family: Arial} /* selector ini kesemuanya untuk
      bold+italic text */
  </STYLE>
</HEAD>
```

Ayo sekarang dicoba! Apakah saya kelihatan sedang berbohong? Tentu tidak. Kebiasaan saya kalau sedang berbohong mengedip-ngedipkan mata (abe: sedang berusaha agar matanya tidak berkedip-kedip c",) hehehe gak ding cuman canda. Suer!

CSS menawarkan berbagai macam cara untuk mengganti warna. Paling mudah adalah dengan menggunakan salah satu dari pre-defined warna yang ada - aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, dan yellow. Kamu juga dapat menggunakan kombinasi warna heksadesimal dengan menambahkan atribut # (tanda kreiss).

```
I {color: #0000FF}
```

Atau nilai kombinasi RGB dalam nilai absolut terhadap persen, seperti ini :

```
I {color: rgb (0, 0, 255)}
```

```
I {color: rgb (0%, 0%, 100%)}
```

Kita dapat mengubah kombinasi warna sesuai dengan gaya yang kita sukai.

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .blue {color: cyan; background-color: #FF8000}
    .green {color: lime; background-color: black}
  </STYLE>
</HEAD>

<BODY>
<P CLASS="blue">
  Sebenarnya semua pekerjaan itu mudah!
<P CLASS="green">
  Setuju, tapi tak semudah mengatakannya ☺
</BODY>
</HTML>
```

Lihat! seandainya kita lebih punya kreasi seni maka kita akan dapat membuat jutaan warna dan memperkaya situs kita.

Mencoba Berbagai Kemungkinan

Hmm, kita telah mempelajari ketentuan huruf dan bagaimana mengubahnya sehingga kita mendapati beberapa variasi huruf. *Dus*, contoh di bawah ini tidak ada yang istimewa tetapi 'ntar saya tunjukkan perbedaannya.

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    P {font-family: "Verdana"}
  </STYLE>
</HEAD>

<BODY>
<P CLASS="tanya">
P: Mengapa jika kita anggap suatu pekerjaan itu mudah
   maka pekerjaan itu akan beneran menjadi lebih mudah?
<P CLASS="jawab">
J: Karena itu merupakan sugesti terhadap diri kita sendiri
</BODY>
</HTML>
```

Anda mungkin sudah tahu jika huruf yang ingin ditampilkan tidak ada dalam font-properties di komputer client, maka browser akan menampilkan default-font (biasanya Times New Roman). Dan disinilah CSS mendapat nilai lebih, karena dengan menggunakan CSS kita dapat mencoba berbagai kemungkinan huruf pada tag . - CSS mengijinkan anda untuk membuat lebih dari satu kemungkinan sebagai alternatif font.

```
P {font-family: "Verdana", "Arial", "Arial Black"} /* first try Verdana,
then Arial, then Arial Black */
```

Sudah mengerti kan? Jika font dengan type *verdana* tidak dijumpai, maka type *arial* lah yang digunakan. Tetapi jika type *arial* ini juga tidak ada maka CSS akan memilih alternatif ketiga yaitu *arial black*.

Anda juga punya sejumlah pilihan untuk menentukan ukuran dari font. Karena CSS juga mempunyai "font-size" properties yang dapat dipakai untuk berbagai jenis tampilan huruf. Sequence? Ya bener berdasarkan urutan!

Pertama sekali yang paling penting saya jabarkan adalah penggunaan salah satu dari tujuh nilai dasar berikut : "xx-small", "x-small", "small", "medium", "large", "x-large" and "xx-large". Coba contoh ini untuk memahirkan :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .tanya {font-size: xx-small}
    .jawab {font-size: x-large}
  </STYLE>
</HEAD>
```

```
<BODY>
<P CLASS="tanya">
P: Mengapa jika kita anggap suatu pekerjaan itu mudah
   maka pekerjaan itu akan beneran menjadi lebih mudah?
<P CLASS="jawab">
J: Karena itu merupakan sugesti terhadap diri kita sendiri
</BODY>
</HTML>
```

Kemudian kita juga dapat menambahkan atribut "larger" atau "smaller" kedalam kondisinya.

```
.tanya {font-size: larger} /* ukuran huruf akan lebih besar dari huruf parent */
.tanya {font-size: smaller} /* ukuran huruf akan lebih kecil dari huruf parent */
```

Juga, kita dapat mengubah dengan menentukan "point" atau "nilai persen". Supaya tidak bingung dicoba aja contoh berikut (saya hanya ngasih petunjuknya saja) :

```
.tanya {font-size: 16pt} /* ukuran dengan 16 point */
.tanya {font-size: 300%} /* ukuran font diubah hingga tiga kali lebih besar */
```

Menambahkan Sedikit Style

Kita juga dapat menambah sedikit style pada teks dengan properti "font-style", dengan nilai "normal", "oblique", dan "italic". Perhatikan ini :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    P {font-family: serif; font-style: italic} /* font italic */
  </STYLE>
</HEAD>

<BODY>
<P CLASS="tanya">
Q. Kenapa contoh yang diberikan tidak pernah diubah?
<P CLASS="jawab">
A. Karena si Abe sudah gak mau susah membuat contoh baru
</BODY>
</HTML>
```

Coba juga untuk "normal" dan "oblique"!

karena ini adalah contoh yang terakhir (lho... udah selesai toh), maka akan saya berikan contoh properties yang lain untuk semakin memperkaya pengetahuan anda yaitu "font-weight". Dimana properties ini memberikan tekanan dengan kata kunci seperti "normal", "bold", "bolder" dan "lighter", atau dengan memberikan skala antara 100 sampai 900. Lihat kembali:

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .tanya {font-weight: bolder}
    .jawab {font-weight: 900}
  </STYLE>
</HEAD>

<BODY>
<P CLASS="tanya">
Q. Kenapa contoh pertanyaan tutorial ini tidak bervariasi?
<P CLASS="jawab">
A. Lho, jangan tanya saya. Tanya penulisnya.
</BODY>
</HTML>
```

Uhm... satu lagi, mungkin ini tidak begitu penting tetapi perlu (cemana sih? Kok jadi gak tegas begini). Properti yang lain yaitu "font-variant" yang mengatur tingkat KAPITAL huruf. Pilihan yang dapat diambil yaitu "small-caps" dan "normal". note : saya tidak jamin hal ini juga berlaku pada Netscape browser.

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    P {font-variant: small-caps}
    /* mengganti semua karakter lower-case dengan karakter upper-case
*/
  </STYLE>
</HEAD>

<BODY>
<P>
Q. Sekarang saya tanya tentang saran kamu kepada penulis?
<P>
A. Bagaimana kalau contohnya dibuat yang lain.
<P>
Q. Maksudnya bagaimana sih? Khan sekarang pertanyaannya sudah
saya ubah.
<P>
A. Yee... dasar tetep aja ngotot (sambil berlalu dengan cuek)
</BODY>
</HTML>
```

Phiuh... akhirnya selesai juga bro (sambil menarik nafas sedalam-dalamnya dengan penuh kelegaan). Mungkin kalau masih sanggup kita akan mempelajari bagaimana membuat "margins", "padding", "borders", "padding", dan "background-images" pada tutorial CSS yang kedua.

Lanjut Pak Supir.. Terima Kasih Ya!

Jika anda juga baca artikel ini berarti anda setidaknya sudah tahu bagaimana style sheet dasar, apa dan bagaimana menggunakannya, serta bagaimana menerapkannya pada halaman web anda. Kalau anda belum tau, wajar, toh penulis tidak menyalahkan pembaca, penulis akui kok artikel pertama itu menyesatkan dan tidak perlu dibaca hihhi.. apalagi artikel ini? Semakin kloplah kebingungan anda.

Dalam artikel yang kedua ini, penulis akan bawa anda dari pulo gadung ke medan.. lho? Maaf, maksudnya penulis akan bawa anda *memahami* bukan untuk *mempelajari*. Setuju? 1 di kali 7, setuju atau gak setuju harus setuju. Maksudnya begini lho! Jika anda telah membaca artikel yang pertama berarti sekarang anda penulis ibaratkan sudah sampai pada terminal *transit*, dengan arti anda harus naik angkot terakhir sekali lagi untuk tiba di rumah anda. Anda juga bisa memutuskan tidak naik angkot? Anda bisa memutuskan untuk naik ojek kali? atau naik bajaj? Ataupun anda bisa telephone orang di rumah untuk menjemput anda. *Lah? Jadi apa hubungannya dengan artikel ini? (Tenang saja kali ini penulis akan jawab ada hubungannya, hehehe.. jarang serius sih! Jadi kurang dipercaya)*

Hubungan yang bisa kita ambil adalah, mengapa kita harus *naik angkot* yang menyesakkan dan panas jika ada *ojek* yang cepat dan kena angin terbuka? Mengapa harus *naik angkot* yang penuh copet dan tidak aman kalau ada *orang dirumah* yang mau menjemput anda dengan nyaman dan tentram?

Lalu? Apa hubungannya dengan artikel ini. Hihhi tidak mengerti juga, begindang lo nek, (baca:begini loh!) anda tidak harus mempelajari artikel ini jika anda rasa ini membosankan dan tidak sesuai dengan minat anda. Anda bisa pelajari hal lain untuk membangun situs anda, katakanlah bahasa pemrograman server side atau database, atau anda ingin menguatkan basic HTML anda. Semua jawaban tergantung anda. Tapi yang pasti ada banyak alternatif lain untuk membangun situs anda. Masih mau lanjut? Lanjut Pak Supir!

Pada akhir artikel, penulis juga akan menjelaskan properties CSS yang mengendalikan gambar latar belakang (background image), memposisikan (positioning), jarak penglihatan dan perataan (visibility dan alignment).

Ciptakan Dekorasinya?

Sebagai tambahan terhadap banyak properties font yang telah kita diskusikan pada artikel pertama, CSS juga *merestui* anda untuk mengatur secara luarbiasa masalah spasi (spacing), perataan (alignment) dan tampilandari text anda (appearance).

Properties “text-decoration” mengijjinkan anda untuk membuat penekanan pada text. Sebuah garis dibawah, di atas, maupun garis yang melaluinya. Coba lihat :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .question {text-decoration: underline} /*tampil garis di bawah text */
    .answer {text-decoration: overline} /*tampil garis di atas text */
    .repeat {text-decoration: line-through}/*tampil garis melalui text */
    .no-imagination {text-decoration: blink; font-weight: bolder}
    /* text berkedip */
  </STYLE>
</HEAD>

<BODY>
<P CLASS="question">
Q. Kenapa why, tidak pernah never, selalu always?
<P CLASS="answer">
A. Karena because selalu always kapan-kapan sometimes?
```

```
<P CLASS="repeat">
Q. Apakah anda tau, siapa orang yang terluca di Indonesia?
<P CLASS="repeat">
A. I Don't know, Kasino and Indro.
<P CLASS="no-imagination">
Maksudnya? Dono, Kasino, dan Indro (Warkop DKI)
</BODY>
</HTML>
```

Catat, bahwa keyword “blink” hanya bekerja pada browser Netscape Navigator. Salah satu penggunaan umum dari property “text-decoration” adalah untuk memberi links garis bawah ketika pointer mouse lewat di atas huruf tersebut (on mouse – over), sebagai kombinasi dari *A:hover pseudo – class* (masih ingat pseudo – classes kan?). Begini contohnya :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    A {text-decoration: none}
    A:hover {text-decoration: underline}
  </STYLE>
</HEAD>

<BODY>

<A HREF="#">Ini Sebuah Link</A>. Yahoo!

</BODY>
</HTML>
```

Berikan Mereka Ruang (Space)

Anda juga dapat mengendalikan jumlah spasi antara karakter-karakter dengan property “letter-spacing”

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    P {letter-spacing: 30px}
  </STYLE>
</HEAD>

<BODY>
<P>
Q. Kenapa Pekerjaan Jika Kita Anggap Mudah Akan Menjadi Benar-benar Mudah?
<P>
A. Kok, Sepertinya ini pertanyaan artikel yang kemarin?

</BODY>
</HTML>
```

Berantakan? Tidak mengapa, nanti anda bisa sesuaikan dengan kebutuhan. Sekarang anda perhatikan contoh di bawah yang berguna membuat indent untuk awal paragraf!

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    P {text-indent: 10px}
  </STYLE>
</HEAD>

<BODY>
<P>
Q. Kenapa Pekerjaan Jika Kita Anggap Mudah Akan Menjadi Benar-benar Mudah?
<BR>
A. Kok, Sepertinya ini pertanyaan artikel yang kemarin?

</BODY>
</HTML>
```

Kerjakan dan Selesaikan Semua Contoh

Perataan horizontal berlangsung melalui property “text-align”, yang menerima harga “left”, “right”, “center” dan “justify”, ketika perataan vertical terjadi melalui property “vertical-align”, yang dapat membawa harga “baseline”, “text-top”, “text-bottom”, “middle”, “sub”, “super”, “top”, dan “bottom”.

Kebanyakan ini adalah self-explanatory (anda harus bisa memakai dan memilikinya sendiri), akan tetapi contoh di bawah akan membuat semuanya lebih jelas :

```
<HTML>
<HEAD>
</HEAD>

<BODY>

<P STYLE="text-align: center; font-weight: bolder">
Judul Di atas!

<P STYLE="text-align: left">
Buku Adalah Dunia!

<P STYLE="text-align: right">
<IMG SRC="bunga.gif" HEIGHT="25" WIDTH="25"> Jangan Menilai Buku Dari Sampulnya

<P STYLE="text-align: justify">
<IMG SRC="bunga.gif" HEIGHT="25" WIDTH="25">Apakah Anda Sadar Kalau Kita Sebenarnya Lemah!

<P STYLE="vertical-align: sub">
<IMG SRC="bunga.gif" HEIGHT="25" WIDTH="25">Iwan Fals? Padi? Sheila On 7? Dan Dewa 19?

<P STYLE="vertical-align: super">
<IMG SRC="bunga.gif" HEIGHT="25" WIDTH="25">Mari Kita Mulai Dari Permulaan.
```

```
<P STYLE="vertical-align: top">
<IMG SRC="bunga.gif" HEIGHT="25" WIDTH="25">Let's We Begin From The
Beginning.

</BODY>
</HTML>
```

Bagaimana? Sudah jelas? Ada yang error? Hehehe.. “bunga.gif” anda tidak nongol ya? Duduls.. terang aja itukan bunga penulis, copy dan pastekan bunga anda ke dalam direktori anda menyimpan script di atas. Perhatikan ekstensinya, jangan salah namanya.

Oke.. cukup masalah bunga, sekarang kita lanjutkan lagi.

Property “text-transform” juga mengizinkan anda untuk mengganti besar huruf, maaf, maksudnya *Kapitalisasi* dari seluruh teks. Pilih antara “uppercase”, “lowercase”, “capitalize”, dan “none”.

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .question {text-transform: capitalize} /*Karakter Pertama Huruf Besar*/
    .answer {text-transform: uppercase} /* Semua Karakter Huruf Besar */
  </STYLE>
</HEAD>

<BODY>

<P CLASS="question">
Q. sebutkan sebuah kata bijak yang anda ketahui?
<P CLASS="answer">
A. never judge a book from its cover!!!

</BODY>
</HTML>
```

Pada contoh di atas, anda lihat properti “capitalize” membuat kalimat “sebutkan sebuah kata bijak yang anda ketahui?” menjadi “Sebutkan Sebuah Kata Bijak Yang Anda Ketahui?” Beda? Tentu saja. Perhatikan pada setiap awal kata menjadi Huruf Besar. Untuk kalimat “never judge a book from its cover!!!” akan berubah menjadi huruf besar (kapital) semuanya!

Property “line-height” mengizinkan anda untuk mengubah space di antara dua garis, dengan demikian anda bisa bebas menumpang tindih kan teks satu dengan yang lain.

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .question {font-size: 30pt; text-transform: capitalize; line-height: 60px}
    .answer {font-size: 50pt; color: red; text-transform: uppercase;
      line-height: 20px}
  </STYLE>
</HEAD>

<BODY>

<P CLASS="question">
Q. Apakah semua properti CSS ini bisa dimanfaatkan?
```

```
<P CLASS="answer">
A. Lho? Kok tanya penulis? Tanya tuh si Abe Poetra!

</BODY>
</HTML>
```

Konsep Watermark

CSS juga hadir dengan properties yang memungkinkan anda untuk mendefinisikan cara bagaimana gambar latar (background) anda di tampilkan. Pertama, property "background-image" memungkinkan anda untuk menetapkan sebuah image background untuk setiap elemen HTML.

```
.question {background-image: url ("http://www.mysite.com/back.gif");
           /*Menampilkan image background dari url*/}
```

Jika anda memerlukan latar belakang ini sebagai *watermark*, yang tidak dapat discroll ketika anda men-scroll ke bawah suatu halaman, anda harus menambahkan property "background-attachment" – Nilai-nilai yang diterima adalah "fixed" dan "scroll"

Anda juga dapat mengatur Ya/Tidak nya suatu image/gambar yang melewati satu halaman dengan properti "background-repeat". Properti ini dapat diambil satu dari empat bagian : "repeat" (letak secara horizontal dan secara vertikal), "repeat-x" (letak secara horizontal saja), "repeat-y" (letak secara vertikal saja), dan "no-repeat" (tidak ada pengaturan letak)

Contoh di bawah ini kita ambil untuk menunjukkan keyword "repeat-y", silahkan dicoba :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .question {font-size: 20pt; background-image:
      url("http://www.mysite.com/bunga.gif"); background-repeat: repeat-y}
  </STYLE>
</HEAD>

<BODY>

<P CLASS="question">
Q. Hmm... ternyata anda sekarang sudah mengerti tentang konsep watermark?
A. Ya, Belum Sepenuhnya lah Pak!

</BODY>
</HTML>
```

Perhatikan, url(<http://www.situs.com/bunga.gif>) mengambil image "bunga.gif" sebagai background yang diatur secara vertikal ☺

Sekarang tugas anda!

Buatlah dan gantilah image diatas dengan nilai "repeat-x". Perhatikan apa yang berubah? Kalau memang tidak berubah, penulis sarankan anda untuk mengganti image atau gambar anda. Sehingga perbedaannya terasa.

Kupas Sampai Tuntas

Model format CSS berasumsi bahwa tiap-tiap elemen dikelilingi oleh tiga area yang berbeda. Dimulai dari dalam keluar, area ini adalah *padding*, *border*, dan *margin*. Masing-masing entitas ini dapat diatur melalui spesial properties CSS, mengijinkan para developer untuk secara singkat mengatur tampilan dan posisi dari setiap elemen HTML.

Nilai margin (garis tepi) dapat kita atur besar dan letaknya melalui properti “margin-top”, “margin-bottom”, “margin-right”, dan “margin-left”, dan ditetapkan seperti ini:

```
DIV {margin-top: 10px; margin-bottom: 10px; margin-right: 5px; margin-left: 5px} /* 10px width for horizontal margins, 5px width for vertical margins */
```

You can also use the catch-all "margin" property

```
DIV {margin: 10px 5px 5px 10px} /* specify widths clockwise */
```

or set a uniform margin width with

```
DIV {margin: 10px} /* equal width for all margins */
```

Anda dapat menyesuaikan lebar border dengan properties self-explanatory “border-top-width”, “border-left-width”, dan “border-right-width”, atau dengan menata suatu border dengan shortcut properti “border-width”. Coba perhatikan sekali lagi!

```
DIV {border-top-width: 50px; border-right-width: 100px; border-bottom-width: 75px; border-left-width: 125px} /* different width for each border */
```

```
DIV {border-width: 50px} /* equal width for all borders */
```

You can also specify border widths with the keywords "thick", "medium", "thin" and "none", like this:

```
DIV {border-top-width: thick; border-right-width: medium; border-bottom-width: thin; border-left-width: none}
```

```
DIV {border-width: thick}
```

Dan akhirnya, padding dapat diatur dengan.. ya benar, anda telah menduganya! Properti “padding-top”, “padding-bottom”, “padding-right”, dan “padding-left”. Untuk ini penulis tidak usah mencontohkannya, anda coba saja sendiri!

Hehehe.. bukan malas, menghemat kertas adalah jawaban bijaknya ☺

CSS juga mengijinkan anda untuk mengatur warna border anda dengan properties “border-color”. Contoh di bawah akan menunjukkan bagaimana “border-color” bekerja!

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .question {border-color: black; border-width: thick}
```

```
</STYLE>
</HEAD>

<BODY>

<P CLASS="question">
Q. How many doctors does it take to change a light bulb?
A. It depends on what kind of insurance you have

</BODY>
</HTML>
```

Mari Belanja Eh.. Belajar!

Jika anda orang yang suka membuat daftar (lists), CSS juga memiliki sesuatu untuk anda – dua properties yang memungkinkan anda untuk mengubah tampilan dari item-item pada penanda daftar anda. Pertama sekali, kita harus mengetahui properti “list-style-type”, yang menerima nilai “disc”, “circle”, “square”, “decimal”, “lower-roman”, “upper-roman”, “lower-alpha”, “upper-alpha” atau “none”. Lihat Contoh :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    LI {list-style-type: square}
  </STYLE>
</HEAD>

<BODY>
<center><h3>:: Daftar Belanja Siang Ini :: </h3></center>
<OL>
<LI>Jeruk 1 Kg.
<LI>Ikan Asin Rebung 2 Kg.
<LI>Beras 2 Goni Plastik Besar 60 Kg.
<LI>Bayam 4 Ikat
<LI>Mie Instant 4 Kardus
</OL>

</BODY>
</HTML>
```

anda dapat juga menggunakan suatu image/gambar biasa selain menggunakan bullet, circle dan disc default dengan properti “list-style-image” – Coba lihat ini :

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    LI {list-style-image: url("http://www.situs.com/bunga.gif" )}
  </STYLE>
</HEAD>

<BODY>
<center><h3>:: Daftar Belanja Siang Ini :: </h3></center>
<OL>
```

```
<LI>Jeruk 1 Kg.  
<LI>Ikan Asin Rebung 2 Kg.  
<LI>Beras 2 Goni Plastik Besar 60 Kg.  
<LI>Bayam 4 Ikat  
<LI>Mie Instant 4 Kardus  
</OL>  
  
</BODY>  
</HTML>
```

Mohon Diperhatikan dan jangan protes, Browser Netscape Navigator tidan mendukung properties di atas.

Lihat Dari Atas Posisinya!

Mungkin salah satu hal yang terbaik dari CSS adalah fakta, untuk pertama kalinya, seorang developer web memiliki kemampuan secara cepat dan tepat mengatur posisi (position) elemen-elemen yang berbeda pada sebuah halaman web. Tiga properti yang memungkinkan anda untuk melakukannya adalah “position”, “left”, dan juga “top” – ingat baik-baik, karena anda akan sering, maaf bukan hanya sering, tapi sangat sering dipergunakan pada script CSS anda.

Properti “position” mengijinkan anda untuk menggambarkan jenis dari positioning ini pada sebuah elemen – pilihannya adalah “absolute” atau “relatives”. Properti ini digunakan secara bersamaan dengan “top” dan “left”, yang menetapkan koordinat bagian atas (top) dan bagian kiri (left) untuk elemen-elemen yang dibahas.

Posisi Absolute mengijinkan anda untuk menempatkan sebuah elemen dimanapun pada sebuah halaman web, tanpa mengindahkan aturan elemen yang berlaku pada elemen didalamnya, sementara Posisi Relative mengijinkan anda untuk meletakkan posisis relative untuk elemen yang lain pada sebuah halaman situs. Bingung? Jangan khawatir, kalau sampai sini anda masih bingung berarti penulis telah sukses. Lho? Tujuan artikel ini sebenarnya bukan untuk mengajari anda. Akan tetapi agar anda semakin bingung.

Di bawah ini adalah sebuah contoh bagaimana anda dapat menggunakan properties ini untuk menempatkan posisi pada block teks:

```
<HTML>  
<HEAD>  
</HEAD>  
  
<BODY>  
  
<DIV STYLE="position: absolute; top:10; left:50; font-family: Verdana;  
font-size: 35pt; color: green">  
Q. Apakah semua orang yang membaca artikel ini sudah mengerti?  
</DIV>  
  
<DIV STYLE="position: absolute; top:140; left:300; font-family: Verdana;  
font-size: 20pt; color: blue">  
A. Belum Tentu!  
</DIV>  
  
<DIV STYLE="position: absolute; top:180; left:50; letter-spacing:5px;  
font-family: Verdana; font-size: 33pt; color: red">
```

```
Fikiran semuanya belum tentu sama.  
</DIV>
```

```
</BODY>  
</HTML>
```

Bagaimana? Masih bingung juga? Lah, tapi sudah diberi contoh? Ya udah sekarang coba perhatikan contoh yang penulis buat. Kopi dan pastekan ke editor favourite anda, kemudian lihat hasilnya! Berubah? Tidak juga? Sekarang coba anda *restore down* jendela browser anda. Atau begini saja bahasa singkatnya, coba anda kecilin ukuran windowsnya. Sudah? Kalo sudah berarti sekarang anda menjumpai kalau teks-teks yang anda tentukan pada script di atas akan mengikuti seberapa besar jendela anda. Ok, sampai sekarang sudah jelas bukan!

Anda juga dapat membuat elemen tertentu *kelihatan* atau tidak, dengan properti “visibility”, yang menerima nilai “visible” dan “hidden”. Gak usah bingung, lihat saja:

```
<HTML>  
<HEAD>  
</HEAD>  
  
<BODY>  
  
<DIV STYLE="position: absolute; top:10; left:50; font-family: Verdana;  
font-size: 35pt; color: green">  
Q. Apakah semua orang yang membaca artikel ini sudah mengerti?  
</DIV>  
  
<DIV STYLE="visibility: hidden">  
A. Belum Tentu!  
</DIV>  
  
<DIV STYLE="position: absolute; top:180; left:50; letter-spacing:5px;  
font-family: Verdana; font-size: 33pt; color: red">  
Fikiran semuanya belum tentu sama.  
</DIV>  
  
</BODY>  
</HTML>
```

Anda perhatikan pada browser anda, kata “A. Belum Tentu!” tidak *tertampil* pada browser karena properti “hidden” tidak menampilkannya.

Tentang Z – Faktor

Dan akhirnya, setelah kita panjang lebar membicarakan *memposisikan* (positioning), kita tidak bisa melupakan faktor Z yaitu properti “z-index”, yang fungsinya untuk meletakkan “stacking order” dari layar-layar yang diletakkan antara satu dengan yang lain. Nilai z – index yang tertinggi akan menekan elemen yang terendah, dan nilai yang terendah akan secara pasti mengikuti aturan di atasnya. Oke, jangan bingung dulu please! Kopi dan pastekan aja scriptnya!

Salah satu penggunaan properti “z-index” adalah untuk menciptakan bayangan (drop shadow) – contoh berikut untuk lebih jelasnya:

```
<HTML>
<HEAD>
  <STYLE TYPE="text/css">
    .level1 {position: relative; top: 15; left: 25; z-index: 10; color: #000000;
    font-family: Arial; font-size:25pt}
    .level2 {position: relative; top: -22; left: 28; z-index: 6; color:
    #ADADAD; font-family: Arial; font-size:25pt}
    .level3 {position: relative; top: -59; left: 29; z-index: 6; color:
    #ADADAD; font-family: Arial; font-size:25pt}
  </STYLE>
</HEAD>

<BODY>
<DIV CLASS="level1">Apakah Berbayang, huh?</DIV>
<DIV CLASS="level2">Apakah Berbayang, huh?</DIV>
<DIV CLASS="level3">Apakah Berbayang, huh?</DIV>
</BODY>

</HTML>
```

Sekarang penulis telah memberi contoh semua properti style sheet yang akan sering anda gunakan. Sebenarnya ini adalah beberapa contoh dari semua properti yang ada. Sekali lagi penulis bilang “Silahkan di Googling aja sendiri” untuk mendapatkan hasil yang lebih dan maksimal sesuai dengan kebutuhan.

Maaf jika contoh di atas tidak maksimal hasilnya pada browser. Satu kelemahan pada CSS ini adalah ketidak adanya *kesamaan* antara para vendor browser. Jika anda benar-benar tertarik dan serius mempelajari CSS ini silahkan kunjungi <http://www.w3.org/TR/REC-CSS2>

Salam dari Medan,

Abe Poetra

- ✓ Masih bego’ pemrograman dan tidak tau apa-apa tentang komputer
- ✓ Berarti dengan hadirnya artikel ini maka penulis telah memenuhi permintaan para pembaca yang merasa penasaran ingin mempelajari “borders”, “padding”, “position”, dan “konsep watermark”.
- ✓ Kunjungi <http://www.ilmukomputer.com/penulis/penulis-abepoetra.php> untuk melihat jati diri penulis ☺

BIOGRAFI PENULIS



Abe Poetra. Dilahirkan dengan nama asli **Abdi Januar Putra.** Lahir di Binjai, 15 Januari 1981 sebagai anak yang bersahaja dan berbakti kepada Orang Tua. Menamatkan Bangku Kuliah di D3 Ilmu Komputer FMIPA-USU pada tahun 2002 dengan penuh perjuangan dan untung saja tamat dengan Nilai yang seadanya (lho kok?). Pernah dipercaya sebagai Assisten Lab. Untuk Fortran, Pascal, dan Bahasa C. Memiliki banyak teman dan lebih banyak lagi musuh hehehe.. ☺

Baru seumur jagung belajar Linux di KPLi-Medan, walaupun begitu, terpilih sebagai Sekretaris Umum untuk periode 2002-2004, dan saat ini sedang belajar serius LAMP (Linux Apache MySQL dan PHP) bersama beberapa sesepuh di kawah candradimuka nya KPLi Medan (PoliTekhnik Negeri Medan). Bersama

rekan-rekan patungan modal mendirikan sebuah usaha yang bergerak dalam usaha warung internet (GO-Internet, juga sebagai sekretariat KPLi-Medan), penjualan komputer, software komputer dan Jaringan, dan penulis dipercaya sebagai ketua Divisi LITBANG.

Abe Poetra juga ditunjuk sebagai Manager Marketing PT. Gayo Belangi Coffee yang berdomisili di Medan. Beberapa bulan terakhir waktunya dihabiskan untuk meneliti dan mempelajari kopi yang siap dipasarkan keseluruh dunia melalui internet.

Informasi lebih lanjut tentang penulis ini bisa didapat melalui:

URL: <http://go.fmipa.usu.ac.id>

Email: abe_poetra@yahoo.com, abe@go.fmipa.usu.ac.id