

PEDOMAN PRAKTIKUM

PEMROGRAMAN WEB

HTML-CSS-JAVASCRIPT-DOM-PHP-MYSQL-AKSESIBILITAS-FINDABILITAS-HOSTING

STANDAR WEB

L. Erawan

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER

2015

Kata Pengantar

Pemrograman web sangat dipengaruhi oleh perubahan dan perkembangan teknologi web. Saat ini perubahan dan perkembangan tersebut berlangsung dengan sangat cepat. Dalam periode waktu yang amat singkat berbagai teknologi web baru bermunculan. Situasi ini memerlukan respon yang cepat dan bijaksana dari dunia pendidikan ranah Teknologi Informasi. Ketersediaan waktu pembelajaran yang terbatas di satu sisi dan materi belajar yang melimpah di sisi lain memerlukan pengelolaan materi belajar secara tepat. Pedoman praktikum ini disusun sebagai jawaban terhadap situasi tersebut. Selain itu, dengan perubahan susunan mata kuliah dalam kurikulum tahun 2014 memerlukan pembaharuan atas materi pembelajaran selama ini. Dalam perkuliahan Pemrograman Web, usaha ini dilakukan dengan antara lain memperbaharui buku Pedoman Praktikum lama yang isinya sudah tertinggal teknologi web.

Pedoman praktikum ini disusun sebagai panduan bagi mahasiswa yang mengambil mata kuliah Pemrograman Web. Dalam pedoman praktikum ini dibahas mengenai praktek berbagai bahasa web yang penting untuk mengembangkan situs web, termasuk diantaranya HTML, CSS, Javascript, dan PHP. Selain itu, ada penerapan berbagai konsep yang sangat penting bagi desainer web untuk mengembangkan situs web yang profesional dan memenuhi berbagai standar web. Diantara konsep-konsep tersebut adalah prinsip dasar desain web, DOM, aksesibilitas dan usability, serta findabilitas. Praktek menempatkan situs kedalam internet atau disebut hosting juga ada dalam bagian terakhir pedoman praktikum ini.

Dengan banyaknya materi yang perlu dipelajari, kurang tepat jika satu materi dibahas secara menyeluruh dan mendalam. Dengan alasan tersebut, materi praktikum disini disusun sedemikian rupa sehingga mahasiswa akan mengetahui berbagai teknologi yang mendasari web saat ini. Keberhasilan menguasai pengetahuan dan teknologi web yang dipraktekkan dalam pedoman praktikum ini sangat tergantung kepada usaha setiap mahasiswa untuk mengembangkannya lebih jauh, bukannya berhenti mempelajari setelah menyelesaikan praktikum Pemrograman Web ini. Di sisi lain, pedoman praktikum ini disusun semudah tetapi selengkap mungkin dengan harapan mahasiswa dapat mengikuti kegiatan praktikum dengan mudah tetapi memperoleh banyak pengetahuan yang diperlukan.

Akhir kata, seperti kata pepatah, Tiada Gading yang Tak Retak. Demikian pula dengan pedoman praktikum ini, tentunya memiliki berbagai kekurangan. Untuk itu, kami mengharapkan saran dan masukan yang positif untuk meningkatkan kualitas pedoman praktikum ini. Dengan kualitas yang

baik sebagai konsekuensi logisnya mahasiswa akan memperoleh materi pembelajaran praktikum yang berkualitas pula yang pada akhirnya meningkatkan kualitas diri mereka sebagai generasi intelektual penerus yang diharapkan bangsa dan negara ini.

Demikian sedikit pemikiran kami sebagai penulis buku pedoman praktikum ini. Semoga pedoman ini berguna dan bermanfaat bagi mahasiswa maupun dosen pengampu mata kuliah Pemrograman Web. Mari kita bersama-sama memajukan kualitas diri kita masing-masing dengan bekerja keras sesuai bidang kita masing-masing demi masa depan yang cerah dan demi jayanya Universitas Dian Nuswantoro kita khususnya Program Studi Sistem Informasi Fakultas Ilmu Komputer. Amin.

Semarang, Agustus 2015

Penulis

L. Erawan

Daftar Isi

Kata Pengantar	i
Daftar isi	ii
PRAKTIKUM 1: Pengenalan Lingkungan Praktikum	1
Penjelasan	1
Praktikum	1
Tugas	3
PRAKTIKUM 2: Teknologi Dasar Internet	4
Penjelasan	4
Praktikum	5
Tugas	5
PRAKTIKUM 3: Bahasa HTML: List, Link, Image	6
Penjelasan	6
Praktikum	13
Tugas	13
PRAKTIKUM 4: Bahasa HTML: Tabel dan Form	14
Penjelasan	14
Praktikum	22
Tugas	22
PRAKTIKUM 5: Mendesain Situs Web	23
Penjelasan	23
Praktikum	30
Tugas	45
PRAKTIKUM 6: Bahasa Style CSS	46
Penjelasan	46
Praktikum	51

Tugas	75
PRAKTIKUM 7: Bahasa Javascript	76
Penjelasan	76
Praktikum	77
Tugas	79
PRAKTIKUM 8: DOM (Document Object Model)	80
Penjelasan	80
Praktikum	81
Tugas	85
PRAKTIKUM 9: Bahasa PHP dan MySQL	86
Penjelasan	86
Praktikum	88
Tugas	90
PRAKTIKUM 10: Aksesibilitas	92
Penjelasan	92
Praktikum	94
Tugas	98
PRAKTIKUM 11: Findabilitas	99
Penjelasan	99
Praktikum	103
Tugas	103
PRAKTIKUM 12: Hosting Situs	104
Penjelasan	104
Praktikum	108
Tugas	108
LAMPIRAN	109



Praktikum 1

Pengenalan Lingkungan Praktikum

Tujuan Pembelajaran

Mengenalkan kepada mahasiswa lingkungan tempat bekerja membuat dan menyelesaikan tugas praktikum

Pokok Bahasan

Mengenal Dokumen HTML

Indikator Keberhasilan

Mahasiswa dapat menggunakan komputer di laboratorium untuk bekerja membuat dan menyelesaikan tugas

Penjelasan

Di kegiatan praktikum pertama ini mahasiswa akan diperkenalkan dengan lingkungan kerja mereka selama mengikuti perkuliahan praktikum Pemrograman Web.

Praktikum

1. Gunakan salah satu komputer yang disediakan di laboratorium computer untuk login ke jaringan menggunakan akun jaringan masing-masing.
(Format nama user akun jaringan mahasiswa adalah a12xxxxxx dengan dua huruf x pertama adalah tahun angkatan mahasiswa dan 5 digit sisanya nim terakhir mahasiswa. Contoh: a121405000. Jika akun jaringan belum dimiliki atau mengalami masalah, laporkan pada asisten laboratorium.)

Untuk selanjutnya aktifitas login ini akan selalu dilakukan mahasiswa ketika akan bekerja di laboratorium komputer.

2. Buka teks editor untuk membuat dokumen web yang ditentukan, yaitu Notepad++, kemudian ketikkan kode HTML berikut:

```
<!DOCTYPE html>
<html>
<head>
<title>Sedang Belajar Pemrograman Web</title>
</head>
<body>
<h1>Hello world!</h1>
</body>
</html>
```

3. Pilih menu **File - Save As** lalu pada kotak dialog pada bagian **file name** ketikkan nama dokumen, yaitu **testing.html**.

*(Sebelum menyimpan, mahasiswa perlu menyiapkan dahulu direktori penyimpanan pada home direktori masing-masing. Perlu diketahui, bahwa file-file web harus disimpan dalam direktori khusus, yaitu **htdocs**. Mahasiswa harus memastikan direktori tersebut sudah ada di dalam home direktori masing-masing. Jika belum ada, mahasiswa membuat atau meminta asisten membantu membuatkan. Kemudian didalam direktori htdocs dibuat sebuah direktori dengan nama **pemrogweb**. Dokumen yang dibuat akan disimpan pada direktori ini.)*

4. Jalankan browser dan ketikkan alamat berikut pada kotak alamat browser:

http://192.168.10.250/~a12xxxxxx/pemrogweb/testing.html, atau

http://dinus.net/~a12xxxxxx/pemrogweb/testing.html

(Jika ingin cepat, ketikkan alamat sampai nama user (~a12xxxxxx) saja kemudian tekan enter, maka akan tampak di layar browser daftar file yang ada didalam home direktori. Setelah itu, Anda tinggal mengklik file testing.html).

5. Ketika langkah sebelumnya sudah dilakukan, maka pada layar browser akan muncul tampilan berupa bidang putih dengan tulisan "Hello World!" yang besar.
6. Selanjutnya lakukan hal berikut, ganti kode "<h1>Hello World!</h1>" dengan kode-kode berikut ini secara bergantian dan setiap kali lihat hasilnya di layar browser:

- <h2>Hello World!</h2>
- <h6> Hello World!</h6>
- <p> Hello World!</p>
- <div> Hello World!</div>
- Hello World!
- Hello World!

7. Tambahkan kode berikut dibawah kode “<h1>Hello World!</h1>” dan lihat hasilnya:

Bangun tidur ku terus mandi
Tidak lupa menggosok gigi
Habis mandi kutolong ibu
Membersihkan tempat tidurku

8. Di akhir masing-masing baris kode tambahan diatas berikan kode **
** lalu amati hasilnya dan lihat perbedaannya.

Tugas

1. Browsing Internet untuk mencari informasi tentang bahasa HTML. Kemudian buat laporan tentang ciri-ciri dokumen HTML dan ciri-ciri kode HTML dengan susunan:
 - Halaman Judul
 - Halaman Daftar Isi
 - Pendahuluan
 - Pembahasan
 - Kesimpulan
 - Daftar Pustaka
2. Kumpulkan dalam bentuk terjilid rapi pada pertemuan praktikum berikutnya



Praktikum 2

Teknologi Dasar Internet dan Web

Tujuan Pembelajaran

Mendorong mahasiswa untuk mengeksplorasi sumber daya internet untuk memperoleh informasi atau pengetahuan yang diminta sehingga mahasiswa dapat memperoleh pengetahuan tersebut untuk dipelajari.

Pokok Bahasan

- User Agent, Web Browser, server web
- Jaringan client-server, client, server, TTL
- HTTP, TCP/IP, FTP, SMTP, UDP, OSI
- HTML, XHTML, XML, CSS, Javascript, AJAX, PHP, JAVA, Python, Ruby on Rails, .NET.
- Framework, SaaS

Indikator Keberhasilan

Mahasiswa dapat:

- menguraikan cara kerja internet
- menguraikan cara kerja web
- menjelaskan protokol-protokol dalam internet
- menjelaskan protokol-protokol dalam web
- membedakan berbagai jenis bahasa web
- menjelaskan konsep framework
- menjelaskan konsep SaaS

Penjelasan

Kegiatan praktikum kali bertujuan untuk membiasakan mahasiswa mencari sumber informasi dari Internet untuk menyelesaikan tugas praktikum yang diberikan.

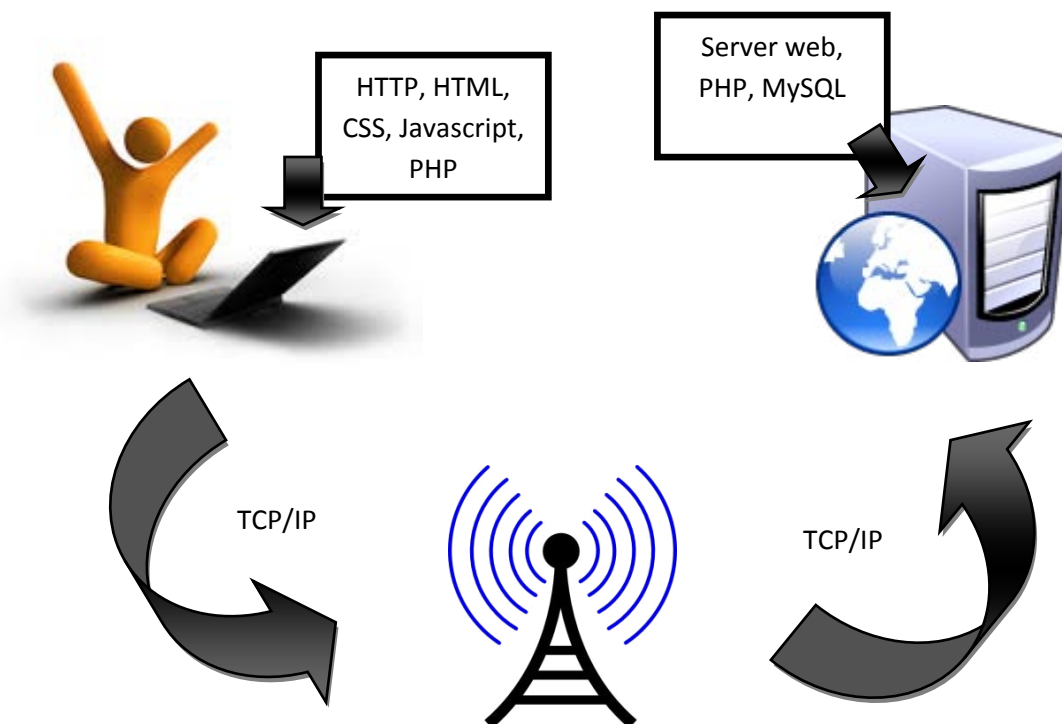
Praktikum

1. Browsing di Internet untuk mencari berbagai situs sumber informasi tentang hal-hal berikut ini:
 - user agent, browser, server web
 - jaringan client server
 - protokol HTTP, TCP/IP, dan OSI
 - HTML, XML, CSS, Javascript, PHP, dan MySQL
 - Tentang Framework dan SaaS
 - Cara kerja internet dan web
2. Pelajari informasi yang diperoleh didalamnya

Tugas

Buatlah laporan berisi:

1. Alamat URL dan ringkasan isi dari situs-situs yang Anda identifikasi berisi informasi-informasi tersebut
2. Kemudian buatlah 2 diagram buatan sendiri yang mengilustrasikan cara kerja Internet dan cara kerja web. Lengkapi diagram tersebut dengan mencantumkan protokol HTTP, TCP, IP, bahasa HTML, XML, Javascript, dan PHP pada tempatnya berfungsi, contoh:



3. Kumpulkan dalam bentuk terjilid rapi pada pertemuan praktikum berikutnya

Praktikum 3

Bahasa HTML: Link, List, Image

Tujuan Pembelajaran

Mempraktekkan pembuatan dokumen HTML serta penggunaan elemen-elemen link, list, dan image sesuai kebutuhan

Pokok Bahasan

- Pengenalan dokumen (X)HTML
- Jenis-jenis doctype (X)HTML: Strict, Transitional, Frameset, html (HTML 5)
- Elemen HEAD
- List
- Image
- Link

Indikator Keberhasilan

Mahasiswa dapat:

- menjelaskan pembagian struktur dokumen HTML
- menjelaskan fungsi bagian head
- menentukan doctype yang tepat untuk dokumen HTML
- menggunakan elemen list, link, dan image secara tepat sesuai semantiknya dan kebutuhan pengembangan situs/aplikasi web

Penjelasan

Dalam praktikum ini Anda akan membuat sebuah halaman web berisi profil diri Anda. Halaman ini merupakan salah satu halaman situs personal yang akan dibangun. Simpan baik-baik untuk digunakan pada praktikum selanjutnya.

Setiap halaman web yang Anda lihat di Internet dibuat dengan bahasa HTML. Bahasa ini dapat dianggap sebagai kerangka yang membentuk struktur setiap halaman web. HTML sendiri singkatan dari Hyper Text Markup Language yang artinya kurang lebih Bahasa Bertanda Teks Hiper. Teks hiper artinya teks yang memiliki link didalamnya. Jika teks ini diklik, akan membawa Anda ke halaman web yang lain. File HTML berjenis file teks, yang artinya dapat dibuat dengan sembarang teks editor yang dapat menghasilkan file teks seperti, NotePad++, Dreamweaver, bahkan MS Word. File HTML dijalankan menggunakan sebuah perangkat lunak browser seperti Chrome, FireFox, Internet Explorer, Opera, dan Safari. Browser akan menerjemahkan perintah-perintah HTML menjadi sebuah halaman web.

File HTML disebut dokumen HTML. Struktur dokumen HTML terdiri dari bagian kepala yang ditandai dengan elemen HEAD dan bagian tubuh yang ditandai dengan elemen BODY. Masing-masing mempunyai fungsinya sendiri. Bagian kepala digunakan untuk meletakkan informasi tentang dokumen seperti judul halaman, nama pembuat dokumen, kata kunci untuk mencari dokumen dalam mesin pencari, dan deskripsi tentang isi dokumen. Fungsi lain yang tidak kalah pentingnya adalah untuk menuliskan perintah pemanggilan file-file CSS dan Javascript yang akan digunakan dalam dokumen. Sedangkan fungsi dari bagian tubuh dokumen untuk meletakkan semua perintah yang menampilkan konten halaman dengan format tertentu ke layar browser. Format dapat berupa paragraf, heading, gambar, daftar, dan sebagainya. Nantinya, sebagian besar perintah HTML akan ditulis pada bagian tubuh ini.

Perintah-perintah bahasa HTML berfungsi menentukan bagaimana suatu teks atau gambar atau obyek halaman web lainnya tampil di layar browser. Perintah HTML mengatur bagaimana suatu teks ditampilkan sebagai judul bab (heading), paragraf, atau sebagai sebuah daftar dari sekumpulan item (list), bagaimana gambar ditampilkan dengan ukuran dan jenis tertentu, dan seterusnya.

Suatu perintah HTML terdiri dari kombinasi tanda 'lebih kecil' dan 'lebih besar' yang mengurung nama perintah HTML, disebut *tag*. Contoh tag HTML misalnya, <p>, yang berfungsi untuk menampilkan kumpulan teks dalam bentuk paragraf. Setiap perintah HTML terdiri dari tag awal dan akhir. Tag akhir sama dengan tag awal hanya ditambah dengan tanda garis miring didepan nama tag. Contoh, pasangan tag awal <p> adalah tag akhir </p>. Satu set tag awal, isi tag, dan tag akhir disebut sebagai elemen. Contoh, tag awal <p>, isi paragraf, dan tag akhir </p>, disebut sebagai elemen paragraf.

Perhatikan dokumen HTML dibawah ini yang akan menampilkan sebuah heading dan sebuah paragraf:

```

<!DOCTYPE html>
<html>
<head>
    <title>Ini Judul Halaman</title>
</head>
<body>
    <h1>Elemen Paragraf</h1>
    <p>Untuk menampilkan sebuah paragraf dalam suatu halaman web memerlukan elemen paragraf yang digunakan saat ini. Teks yang ditandai sebagai paragraf akan ditampilkan seperti sebuah paragraf. Ciri-ciri paragraf adalah memiliki perataan kiri dan kanan, dan memiliki jarak spasi antar paragraf yang lebih besar dari jarak spasi antar barisnya. Dalam penggunaannya, paragraf dalam halaman web biasanya ditampilkan dengan perataan kiri yang merupakan nilai defaultnya, seperti perataan paragraf ini. Perataan kiri paragraf bertujuan agar baris-baris paragraf mudah dibaca dan ini sesuai dengan aturan usability web.</p>
</body>
</html>

```

Perhatikan susunan dokumen yang dimulai <!DOCTYPE html>. Perintah ini untuk menentukan bahwa jenis dokumen adalah HTML. Baris selanjutnya tag <html> yang menandai awal dokumen. Pada akhir dokumen, ditandai dengan tag penutup </html>. Elemen *title* untuk membuat judul halaman yang akan ditampilkan pada bagian kiri atas jendela browser. Terakhir, penggunaan elemen heading tingkat 1, <h1>, dan elemen paragraf, <p>. Elemen heading dapat sampai dengan tingkat 6, <h2>, <h3>, <h4>, <h5>, <h6>. Heading tingkat 1 memiliki ukuran huruf paling besar. Elemen paragraf akan menampilkan paragraf dengan perataan rata kiri yang merupakan nilai default perataan. Sekarang tugas Anda adalah:

1. Buatlah sebuah dokumen HTML Anda sendiri yang akan menampilkan sebuah artikel. Buat sedemikian rupa sehingga artikel Anda terdiri dari beberapa bab dan sub bab. Untuk membuat judul artikel gunakan heading tingkat 1, untuk judul bab gunakan <h2>, dan untuk judul sub bab gunakan <h3>. Tema artikel tentang diri Anda, misalnya riwayat pendidikan, hobi, dan sebagainya. Gunakan elemen paragraf secara benar. Simpan dengan nama **tentang_kami.html**. Contoh strukturnya:

```

<!DOCTYPE html>
<html>
<head>
    <title>Judul halaman</title>
</head>
<body>
    <h1>Judul Artikel</h1>
    <h2>Judul bab 1</h2>

```

```

<p>isi paragraf</p>
<h3>Judul sub bab 1</h3>
<p>isi paragraf</p>
<h3>Judul sub bab 2</h3>
<p>isi paragraf</p>

<h2>Judul bab 2</h2>
<h3>Judul sub bab 1</h3>
<p>isi paragraf</p>
<h3>Judul sub bab 2</h3>
<p>isi paragraf</p>
</body>
</html>

```

Salah satu kelebihan dari web adalah kemampuannya untuk menampilkan gambar. Web biasanya menggunakan jenis file gambar *jpg*, *gif*, dan *png*. Jenis *jpg* biasanya digunakan untuk membuat gambar foto, jenis *gif* untuk gambar yang dapat dianimasikan, sedangkan *png* memiliki dua kelebihan tersebut. File gambar yang digunakan sebaiknya dipilih yang berukuran kecil sehingga tidak akan memberatkan halaman web ketika dimuat ke browser, tetapi harus tetap mempertahankan kejernihan dan kejelasan gambar agar tetap tampil bagus di browser.

Tag yang digunakan untuk menampilkan gambar adalah tag **. Tag ini memiliki atribut *src* yang harus digunakan, berfungsi untuk menentukan alamat file gambar yang akan ditampilkan. Tag ini tidak memiliki pasangan maka penulisannya harus diakhiri dengan tanda *backslash*.

```

```

Terkadang kita perlu menyesuaikan ukuran gambar agar pas ditampilkan pada halaman. Atribut yang dapat digunakan untuk ini adalah *height* dan *width*, masing-masing untuk mengatur tinggi dan lebar gambar. Satuan ukuran biasanya menggunakan *px* singkatan dari *pixel*. Gunakan salah satu saja dari atribut ini untuk mengatur ukuran gambar agar perbandingan tinggi dan lebar gambar tetap. Jika pengecilan ukuran gambar terlalu besar, lebih baik file gambar disimpan ulang dengan ukuran yang baru menggunakan perangkat lunak grafis seperti adobe photoshop, karena meskipun ukuran gambar sudah kecil dilayar, tetapi ukuran file gambar tetap besar.

```

```

Pengguna internet tidak hanya orang normal saja, orang-orang dengan indera yang “kurang” juga berhak menggunakan internet. Definisi “kurang” disini merujuk kepada kurang

pendengaran, penglihatan, gerakan tangan karena cacat atau usia tua, lingkungan yang terlalu bising atau kurang pencahayaan sehingga menyulitkan mengakses halaman web. Usabilitas merupakan teori dan teknik dalam membuat situs web yang mudah digunakan. Salah satu teknik usabilitas adalah menggunakan atribut *alt* pada tag **. Atribut ini akan menyediakan teks alternatif untuk gambar, yang akan dibaca oleh perangkat lunak *screen reader* untuk pengguna yang kekurangan penglihatan.

```

```

2. Kedalam dokumen **tentang_kami.html**, tambahkan foto Anda sebelum paragraf pertama artikel. Gunakan gambar yang jelas, ukuran file kecil, dan atribut alt untuk meningkatkan usabilitas. Contoh penempatan:

```
.....  
<body>  
<h1>Judul Artikel</h1>  
<img src="" width="" alt="" />  
<h2>Judul bab 1</h2>  
.....
```

Anda bisa menyediakan media bagi pengguna untuk menuju halaman lain situs Anda atau situs lain dengan menggunakan hyperlink atau disingkat link. Dengan link setiap halaman web dapat dihubungkan dengan halaman web yang lain. Tidak hanya halaman web saja, sumber daya web yang lain seperti file gambar, suara, video, dan file-file lain seperti file pdf dapat dihubungkan. Ini merupakan salah satu keistimewaan web.

Link dibuat menggunakan tag *<a>*. Tag ini memiliki sebuah atribut yang harus dituliskan, yaitu *href* yang berfungsi menentukan alamat halaman web lain atau file lain sebagai tujuan dari link. Penulisan tag ini sebagai berikut:

```
<a href="alamat_tujuan" >Teks Link</a>
```

Teks Link adalah teks yang mewakili link, atau teks yang akan ditampilkan pada browser sebagai perwujudan link. Jika teks ini diklik, halaman web lain yang ditentukan alamatnya pada atribut *href* akan ditampilkan.

Alamat tujuan link ada yang berbentuk relatif dan ada yang absolut. Alamat relatif digunakan pada link yang menuju ke halaman web dalam situs yang sama, sedangkan alamat absolut digunakan untuk mengalami tujuan link yang berada pada situs lain.

Contoh alamat relatif:

```
<a href="produk/produk_gadget.html">Produk Gadget</a>
```

Contoh alamat absolut:

```
<a href="http://www.dinus.ac.id">Universitas Komputer Terbaik</a>
```

Alamat relatif tidak perlu lengkap, sebaliknya alamat absolut harus lengkap terdiri dari protokol, domain, dan nama file yang dituju (Contoh alamat absolut diatas sebenarnya adalah *http://www.dinus.ac.id/index.php*, tetapi karena merupakan halaman homepage maka tidak perlu dituliskan nama filenya).

3. Tambahkan link pada foto diri Anda dalam dokumen **tentang_kami.html**. Alamat tujuan link menuju halaman facebook atau Twitter Anda.

Ketika menyusun konten halaman, ada saatnya Anda ingin menampilkan dalam bentuk daftar. Misalnya ingin menampilkan hobi Anda yang lebih dari satu. Untuk menampilkan daftar ini, tag yang diperlukan adalah tag `<ul>`, unordered list, atau `<ol>`, ordered list. Jenis unordered list, item-itemnya ditandai dengan bullet sedangkan ordered list dengan angka, huruf, atau romawi. Item-itemnya dibuat dengan tag `<li>`

Unordered list:

```
<ul>
    <li>...</li>
    <li>...</li>
</ul>
```

Ordered list:

```
<ol>
    <li>...</li>
    <li>...</li>
</ol>
```

List juga bisa ditampilkan secara bertingkat atau list didalam list:

```
<ul>
    <li>...</li>
    <li>
        <ol>
            <li>...</li>
        </ol>
    </li>
</ul>
```

4. Pada artikel tentang diri Anda, konten yang berupa daftar, misalnya hobi atau daftar prestasi, tampilkan dengan elemen list.

Pada suatu artikel sangat penting Anda menekankan bagian teks yang penting. Penekanan teks bisa menggunakan penebalan atau pemiringan huruf. Elemen HTML untuk menebalkan teks adalah `<strong>...</strong>` dan untuk pemiringan menggunakan `<em>...</em>`. Contoh:

`<strong>Teks ini ditebalkan</strong>`

Atribut `src` pada elemen `image`, `href` pada elemen `link`, adalah contoh atribut yang dapat ditambahkan pada suatu elemen HTML. Setiap elemen HTML memiliki atribut berbeda. Atribut adalah perluasan fungsi yang dapat ditambahkan pada suatu elemen HTML. Penulisan atribut didalam tag awal dan jika ada lebih dari satu atribut, penulisannya dipisah dengan satu spasi antar atribut.

Teks yang tampil pada suatu halaman web dapat diubah baik ukuran, warna, maupun jenis hurufnya. Untuk itu kita perlu menggunakan atribut *style*. Atribut ini dapat digunakan pada semua elemen HTML. Nilai dari atribut ini adalah properti elemen HTML yang hendak diubah dan nilainya. Untuk mengubah ukuran huruf, properti yang digunakan adalah *font-size*. Contoh:

`<p style="font-size:14px">...</p>`

Satuan ukuran dalam web adalah *px* (pixel), *pt* (point), *in* (inchi), *em*, dan % (prosentase). Elemen paragraf diatas akan ditampilkan dengan teks berukuran 14 piksel.

Sedangkan untuk mengubah warna, dan jenis huruf, masing-masing menggunakan *font-color* dan *font-family*. Nilai dari *font-color* berupa nama warna seperti *blue*, *lightyellow*, atau kode warna dalam hexadesimal seperti *#ffffff* (putih), *#ff0000* (merah), *#00ff00* (hijau), *#0000ff* (biru), *#555555* (abu-abu). Kode warna itu terdiri dari 6 digit angka yang terbagi dalam tiga bagian: dua digit pertama mewakili warna merah, dua digit berikutnya mewakili warna hijau, dan sisanya warna biru. Masing-masing bagian bisa bernilai dari rentang 00 sampai dengan ff. Mencampur ketiga warna tersebut dengan kuantitas yang berbeda-beda akan menghasilkan jutaan warna yang bisa Anda dapatkan.

5. Pada artikel tentang diri Anda, ubahlah warna teks judul bab dan sub bab dengan warna yang berbeda, dan ubahlah ukuran hurufnya menjadi sedikit lebih besar. Ubahlah juga warna teks item-item dari list. Pilihlah warna yang masih bisa dilihat dengan mudah oleh mata (warna teks kuning dengan latar putih sulit untuk dibaca). Beri penekanan pada bagian teks yang penting dengan menggunakan penebalan atau pemiringan huruf.

Praktikum

Kerjakan apa yang diminta dalam penjelasan

Tugas

Kerjakan perintah-perintah nomor 1 sampai dengan nomor 5 sampai selesai. Baca petunjuk dan penjelasannya baik-baik

Bahasa HTML: Tabel dan Form

Tujuan Pembelajaran

Mempraktekkan penggunaan elemen tabel dan form sesuai fungsinya

Pokok Bahasan

- Table
- Form (X)HTML
 - ❖ Fungsi Form
 - ❖ Mekanisme pengiriman data pada form
 - ❖ Jenis-jenis input form: field, textarea, radio button, checkbox, select menu, submit button
 - ❖ Membuat validasi form
 - ❖ Elemen baru form HTML 5

Indikator Keberhasilan

Mahasiswa dapat menggunakan elemen tabel dan form HTML secara tepat sesuai fungsinya dan kebutuhan pengembangan situs/aplikasi web

Penjelasan

Tabel

Seringkali konten yang akan ditampilkan berupa sekumpulan data, misalnya produk yang dijual, statistik penjualan, dan sebagainya. Jenis konten ini perlu ditampilkan dalam bentuk tabular (tabel) agar mudah dibaca. Tabel terdiri dari baris dan kolom. HTML menyediakan elemen *table* untuk menampilkan jenis konten ini. Untuk membuat baris-baris tabel menggunakan tag `<tr>`, singkatan dari *table row*. Didalam baris-baris tabel dibuat kolom-kolom tabel menggunakan tag `<td>`, singkatan dari *table data*. Susunan penulisan dari tag-tag tabel tersebut adalah:

```

<table>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
</table>

```

Diantara tag <td> dan </td> adalah tempat untuk meletakkan konten yang akan ditampilkan dalam tabel. Konten dapat berupa teks, gambar, video, maupun tabel yang lain.

Tabel didalam Tabel

Jika konten sel berisi tabel yang lain, maka penulisannya sebagai berikut:

```

<table>
<tr>
  <td>
    <!--Ini sel tabel yang berisi tabel juga -->
    <table>
      <tr>
        <td>Sel ke-1.1</td>
      </tr>
      <tr>
        <td>Sel ke-1.2</td>
      </tr>
    </table>
  </td>
  <td>Sel ke-2</td>
</tr>
</table>

```

Sel pertama dari tabel luar berisi sebuah tabel yang lain (bercetak tebal). Tabel didalam tabel ini dapat dibuat sampai beberapa tingkat tidak terbatas.

Tag <th>

Jika Anda melihat pada berbagai bentuk tabel yang telah dibuat orang, entah menggunakan web atau dalam bentuk cetak, biasanya tampilan baris pertama yang berisi judul kolom dibuat berbeda untuk memudahkan orang membaca tabel.

Selain tag <td>, ada sebuah tag yang fungsinya untuk membuat sel tabel juga. Tetapi sel yang dibuat akan menampilkan teks isi sel dengan cara berbeda yaitu: teks dicetak rata tengah tebal. Tag ini

adalah tag <th> singkatan *table header*. Dibawah ini contoh penggunaan tag <th> untuk membuat kolom-kolom baris pertama menjadi judul kolom:

```
<table>
  <tr>
    <th>...</th>
    <th>...</th>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
</table>
```

Tag <thead> dan <tbody>

Sebagaimana halnya dokumen HTML, yang terbagi menjadi bagian kepala dan tubuh, sebuah tabel dalam web juga perlu memiliki struktur yang jelas, yaitu memiliki bagian kepala dan tubuh. Untuk masalah ini, HTML telah menyediakan tag <thead> dan <tbody>. Tag yang pertama digunakan untuk menandai bagian kepala tabel, sedang yang kedua untuk menandai bagian tubuh tabel. Letak penulisannya sebagai berikut:

```
<table>
  <thead>
    <tr>
      <th>...</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>...</td>
    </tr>
  </tbody>
</table>
```

Menggabungkan sel

Sel-sel tabel secara normal memiliki lebar dan tinggi yang sama. Jika kita ingin membuat sebuah sel memiliki lebar atau tinggi yang berbeda dari sel-sel lainnya, maka satu-satunya cara yang bisa kita lakukan adalah dengan menggabungkan beberapa sel menjadi satu.

Cara ini disebut merge atau penggabungan sel. Terdapat dua jenis penggabungan. Pertama, sel-sel tabel dapat digabungkan secara vertikal, yaitu sel-sel yang terletak di kolom yang sama digabung.

Kedua, sel-sel tabel digabungkan secara horisontal, yaitu sel-sel yang terletak di baris yang sama digabungkan.

Untuk menggabungkan sel-sel tabel ini diperlukan atribut rowspan atau colspan. Atribut rowspan digunakan untuk menggabungkan sel-sel tabel pada kolom yang sama. Atribut colspan untuk menggabungkan sel-sel tabel pada baris yang sama. Berikut contoh penggabungan kedua jenis:

1. Secara Vertikal (Rowspan)

Tabel dengan kode HTML dibawah ini sel-sel kolom pertama akan digabung:

```
<table>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
</table>
```


Setelah digabung maka kondisi kode HTML menjadi seperti berikut:

```
<table>
  <tr>
    <td rowspan="3">...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
  </tr>
</table>
```


2. Secara Horizontal (Colspan)

Tabel dengan kode HTML dibawah ini sel-sel baris pertama akan digabung:

```
<table>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
</table>
```


Setelah digabung maka kondisi kode HTML menjadi seperti berikut:

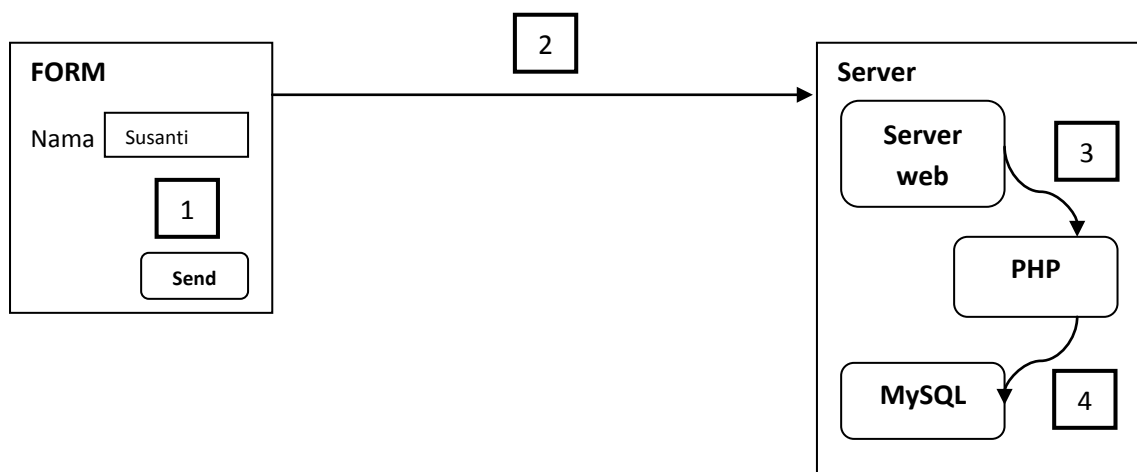
```
<table>
  <tr>
    <td colspan="2">...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
</table>
```


Form

Form merupakan alat yang digunakan pemilik situs untuk mengumpulkan data yang dibutuhkan dari para pengguna atau pengunjung situsnya. Form HTML dapat Anda pahami sebagai sebuah formulir seperti di dunia nyata, misalnya formulir pendaftaran mahasiswa baru, yang berisi serangkaian pertanyaan yang harus diisi dengan data oleh calon mahasiswa baru. Data yang diisikan harus sesuai dengan pertanyaan agar dapat diterima oleh pemilik formulir. Isian yang tercantum dalam formulir tersebut bentuknya bermacam-macam. Ada isian yang harus diisi dengan cara ditulis langsung pada area yang disediakan, ada juga yang tinggal memilih salah satu dari beberapa pilihan yang tersedia, misalnya isian jenis kelamin. Sebagai pemilik formulir, data yang berada dalam formulir akan diolah lebih lanjut sesuai kebutuhan.

Cara Kerja Form

Penggunaan form akan melibatkan server web, script CGI (PHP), dan database (MySQL). Data akan dikirim dari form ke file pengolah data yang ditentukan dengan atribut *action* ketika tombol *submit* form dikirim. Berikut ilustrasi cara kerja form:



Penjelasan gambar:

1. Tombol submit diklik
2. Data nama 'Susanti' akan dikirim ke server web

3. Server web mencari file PHP yang ditentukan dalam form untuk mengolah data kemudian menyerahkannya pada PHP untuk diterjemahkan dan dieksekusi.
4. PHP menghubungi MySQL ketika data 'Susanti' akan disimpan ke database

Tag Form

Tag yang diperlukan untuk membuat form pada dasarnya terbagi menjadi 2 bagian:

- bagian untuk mendefinisikan form
Menggunakan tag `<form>`. Atribut yang harus digunakan method, untuk menentukan jenis pengiriman data dari form ke file pengolah datanya. Ada dua jenis metode yaitu post dan get. Metode post digunakan apabila data yang dikirim berjumlah banyak, akan disimpan ke database, dan bersifat sensitif. Sedangkan get digunakan bila data sedikit, tidak sensitif, dan hanya untuk mengambil data dari database atau tidak mengubah isi database. Contoh data yang perlu dikirim menggunakan metode post misalnya data diri pengguna yang mengandung data nomor kartu kredit.
- bagian untuk membuat isian form
Isian form ada beberapa jenis, yaitu: text, password, hidden, file, radio, checkbox, textarea, select, submit, reset. Tag untuk membuat isian form ada beberapa macam tag: `<input>`, `<select>`, dan `<textarea>`. Berikut tabel jenis-jenis isian form dan tag yang diperlukan untuk membuatnya:

Jenis Isian form	Tag
Text	<code><input></code>
Password	<code><input></code>
Hidden	<code><input></code>
File	<code><input></code>
Radio	<code><input></code>
checkbox	<code><input></code>
Textarea	<code><textarea></code>
Kotak pilihan	<code><select></code>
Submit	<code><input></code>
Reset	<code><input></code>

Untuk tag `<input>`, untuk menentukan isian form mana yang akan dibuat ditentukan dengan atribut *type*.

Memberi Nama Setiap Isian Form

Salah satu hal terpenting dari form adalah memberi nama pada setiap isian form. Nama ini yang akan digunakan pada file PHP untuk mengolah data. Untuk memberi nama isian form gunakan atribut *name*. Anda bisa menyamakan nama isian form ini seperti nama variabel pada suatu program.

Atribut-atribut

Selain atribut *type* dan *name* diatas, terdapat beberapa atribut lain yang sering digunakan dalam form, yaitu:

- **Size**
Digunakan pada tag `<input>` untuk menentukan panjang kotak masukan.
- **Value**
Pada tag `<option>` dalam isian form jenis select, isian jenis radio dan checkbox, digunakan untuk menentukan nilai pilihan. Pada pembuatan tombol submit digunakan untuk menentukan teks diatas tombol.
- **Rows dan cols**
Digunakan pada tag `<textarea>` untuk menentukan tinggi dan lebar dari kotak isian textarea.

Untuk memahami setiap jenis isian form, pelajari dan coba kode HTML untuk membuat form dibawah ini:

```
<form method="post" action="sv_barang.php">
  Kode barang: <input type="text" name="kdbrg" size="5" /><br />
  Nama Barang: <input type="text" name="nmbrg" size="35" /><br />
  Harga: <input type="text" name="hrgrg" size="15" /><br />
  Satuan:
  <select name="satbrg">
    <option value="kg">Kilogram</option>
    <option value="liter">Liter</option>
  </select><br />
  Jumlah: <input type="text" name="kdbrg" size="5" /><br />
  Gudang:
  <input type="radio" name="gudang" value="A">Gudang A
  <input type="radio" name="gudang" value="B">Gudang B<br />
  Keterangan:
  <textarea name="ket" rows="5" cols="75"></textarea><br />
  <input type="submit" value="Send" />
</form>
```

Praktikum

1. Buatlah tabel berisi data jadwal kuliah Anda mencakup informasi hari, jam, ruang kuliah, kelompok kuliah, dan nama dosen pengampu.
2. Gunakan tag `<th>`, `<thead>`, `<tbody>` secara benar pada tabel Anda. Perlu diketahui, tag `<thead>`, dan tag `<tbody>` fungsinya tidak berhubungan dengan tampilan tabel, tetapi untuk memberikan struktur yang jelas kepada tabel. Struktur yang jelas akan berpengaruh terhadap aspek usabilitas tabel. Dampak penggunaan kedua tag ini barangkali tidak 'kasat mata' bagi pengguna yang normal. Tetapi bagi pengguna yang memiliki keterbatasan penglihatan yang melengkapi komputernya dengan perangkat lunak pembaca layar (*screen reader*), kedua tag ini sangat penting bagi mereka. Ini karena *screen reader* mereka akan memberikan informasi (membaca tabel) didalam tabel dengan benar jika tabel berstruktur jelas.
3. Tambahkan tabel ini pada halaman "Tentang Kami" yang sudah Anda buat pada praktikum sebelumnya.
4. Buat form yang berisi inputan NAMA, JENIS KELAMIN, EMAIL, ALAMAT, KOTA, PESAN. Isian JENIS KELAMIN menggunakan jenis *radio*, isian KOTA menggunakan jenis *select*, isian PESAN menggunakan jenis *textarea*, isian lain gunakan jenis *text*. Simpan dengan nama **kontak.html**
5. Jika sudah selesai dan Anda coba jalankan, maka form tersebut belum dapat digunakan untuk menyimpan data. File PHP yang diperlukan untuk menyimpan data akan ditambahkan pada materi tentang PHP

Tugas

Tugas praktikum ini adalah menghasilkan tabel yang disisipkan kedalam file **tentang_kami.html** dan halaman berisi form yang disimpan dengan nama **kontak.html** .



Praktikum 5

Mendesain Situs Web

Tujuan Pembelajaran

Mempraktekkan pembuatan desain situs menggunakan perangkat lunak grafis untuk membantu membuat desain situs.

Pokok Bahasan

Penggunaan Photoshop untuk membantu mendesain halaman web

Indikator Keberhasilan

Mahasiswa dapat menghasilkan desain halaman homepage dari situs yang akan dikembangkan dari hasil mengerjakan langkah-langkah praktikum

Penjelasan

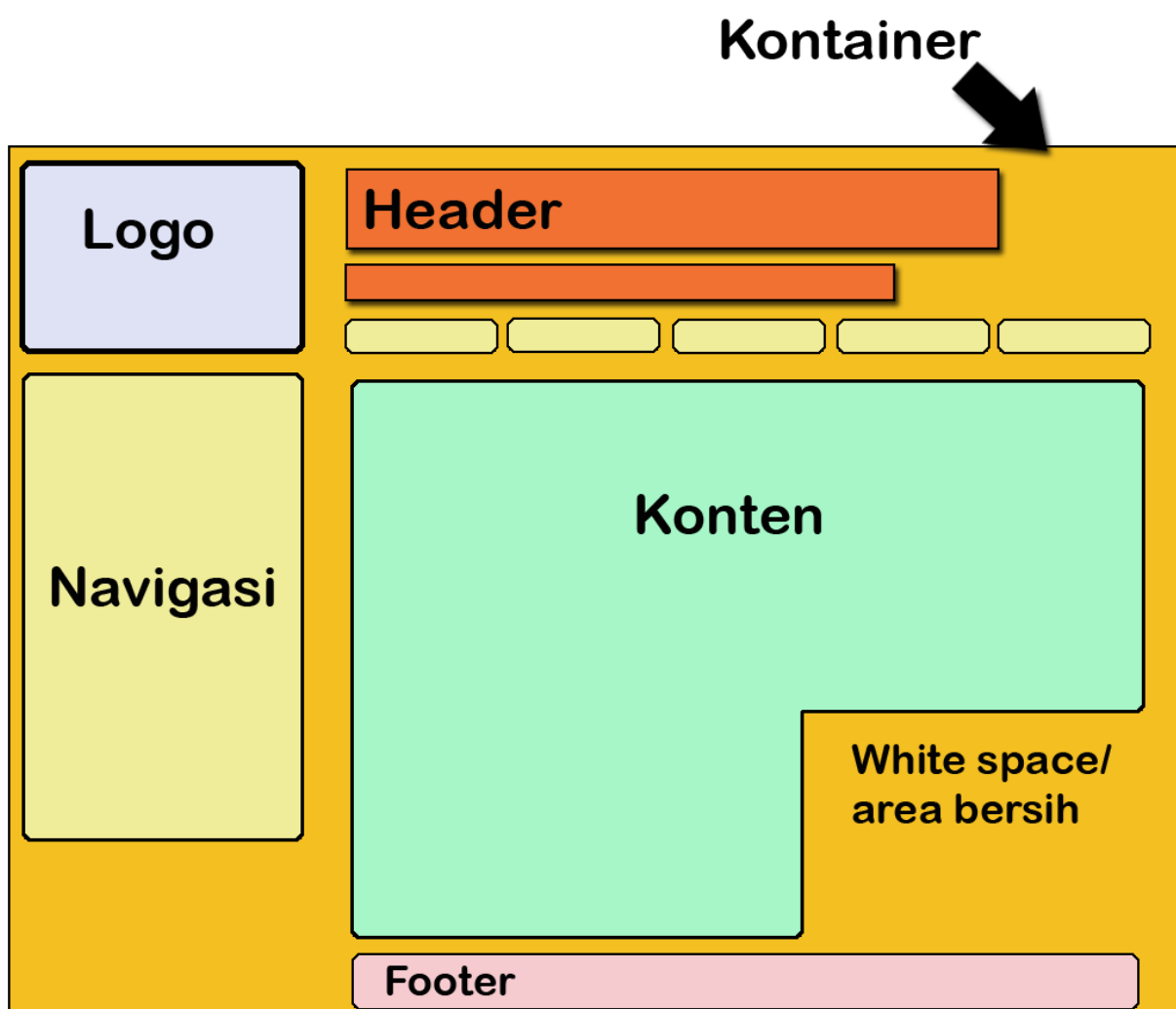
Situs Web

Situs web atau *website* adalah serangkaian halaman web yang memiliki tema yang sama dan saling berhubungan yang berisikan kumpulan informasi yang disediakan secara perorangan, kelompok, atau organisasi. Situs web biasanya ditempatkan pada sebuah server web yang dapat diakses melalui jaringan seperti Internet, ataupun jaringan wilayah lokal (LAN) melalui alamat Internet yang dikenali sebagai URL (Uniform Resources Locator). Gabungan atas semua situs yang dapat diakses publik di Internet disebut sebagai World Wide Web atau lebih dikenal dengan singkatan WWW. Meskipun situs web umumnya dapat diakses publik secara bebas, pada prakteknya tidak semua situs memberikan kebebasan bagi publik untuk mengaksesnya. Beberapa situs web mewajibkan

pengunjung untuk melakukan pendaftaran sebagai anggota, atau bahkan meminta pembayaran untuk dapat menjadi anggota agar dapat mengakses isinya misalnya situs-situs yang menampilkan pornografi, situs-situs berita, layanan surel (*e-mail*), dan lain-lain. Pembatasan-pembatasan ini umumnya dilakukan karena alasan keamanan, menghormati privasi, atau karena tujuan komersil tertentu.

Anatomi Halaman Web

Sebagian besar halaman web situs web pada umumnya memiliki komponen-komponen berikut ini:



Kontainer membungkus semua komponen halaman. Logo berisi identitas pemilik situs. Navigasi atau menu berisi sejumlah link untuk berpindah-pindah halaman. Sebagian besar informasi halaman diletakkan di area konten yang merupakan bagian terbesar dan terpenting halaman. White space atau diindonesiakan area bersih adalah elemen penting dari layout halaman. Elemen ini berperan sebagai pembatas antar komponen halaman. Berguna untuk memberi ruang napas bagi layout

halaman agar tidak tampak berjejal, juga sebagai ruang berhenti bagi pengguna saat melihat halaman. Footer berisi informasi singkat pemilik situs dan hak cipta serta sejumlah link.

Layout Situs Web

Desain situs web yang baik dan tepat akan membuat pengunjung merasa nyaman dengan tampilan situs web tersebut, dan tetap dapat fokus pada isi (konten). Desain situs web seharusnya tidak mengganggu kejelasan bagian isi.

Pada desain layout yang tepat, pengunjung dapat berpindah ke halaman lain dengan mudah melalui navigasi yang baik, mudah dan informatif. Pengunjung merasa bahwa setiap halaman dari situs merupakan satu kesatuan. Setiap halaman memiliki posisi dan kesamaan tampilan (layout), yang berbeda hanya isi (konten).

Adanya layout yang baik dapat membuat halaman web lebih cantik dan seimbang, enak dilihat dan mudah dibaca. Desain layout halaman web meliputi usaha:

- pembagian tempat pada halaman
- pengaturan jarak spasi antar komponen
- pengelompokan teks dan grafik
- penekanan pada bagian tertentu halaman web

Pedoman Layout

Ada 4 prinsip dasar dalam menyusun desain halaman-halaman dari situs web:

1. Keseimbangan Visual

Konsep keseimbangan visual sama dengan konsep keseimbangan dalam permainan jungkat-jungkit. Seperti obyek fisik, elemen layout juga mempunyai bobot. Jika bobot elemen pada kedua sisi sama maka dikatakan mereka seimbang satu sama lain. Ada dua bentuk keseimbangan visual: simetris dan asimetris.

Keseimbangan simetris terbentuk apabila kedua sisi halaman, kiri dan kanan, memiliki komposisi elemen yang sama. Keseimbangan asimetris sedikit lebih abstrak dan secara visual pada umumnya lebih menarik. Tidak hanya membuat sisi satu halaman menjadi seperti bayangan cermin sisi yang lain, asimetris melibatkan penggunaan perbedaan ukuran, bentuk, gaya, dan penempatan.

Desain Simetris



Desain asimetris



2. Kesatuan

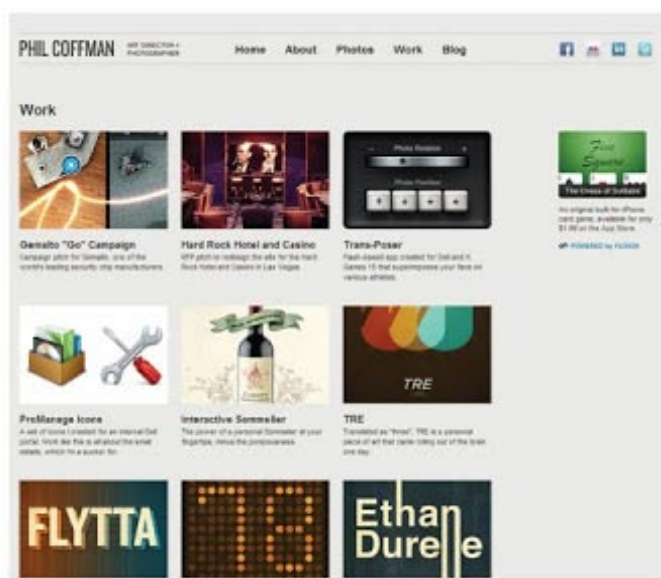
Kesatuan merupakan hal yang sangat penting dalam desain. Kesatuan mampu membentuk dan membangun karakter, menciptakan perbedaan, dan memunculkan suatu ingatan yang kuat tentang suatu produk.

Secara definitif, kesatuan berarti :

"kesinambungan antara beberapa elemen yang tampak menyatu dan harmonis di dalam sebuah desain"

Untuk mencapai kesatuan dalam layout menggunakan konsep kedekatan (proximity) dan pengulangan (repetition).

- Kedekatan merupakan cara yang paling sederhana untuk membuat sebuah desain tampak menyatu. Cukup dengan mendekatkan elemen-elemen tertentu satu sama lain untuk mendefinisikannya sebagai suatu kelompok, dan memberi jarak yang cukup dengan elemen-elemen yang lain diluar dari kelompok. Contoh kesatuan dengan menggunakan kedekatan (gambar-gambar diletakkan berdekatan):

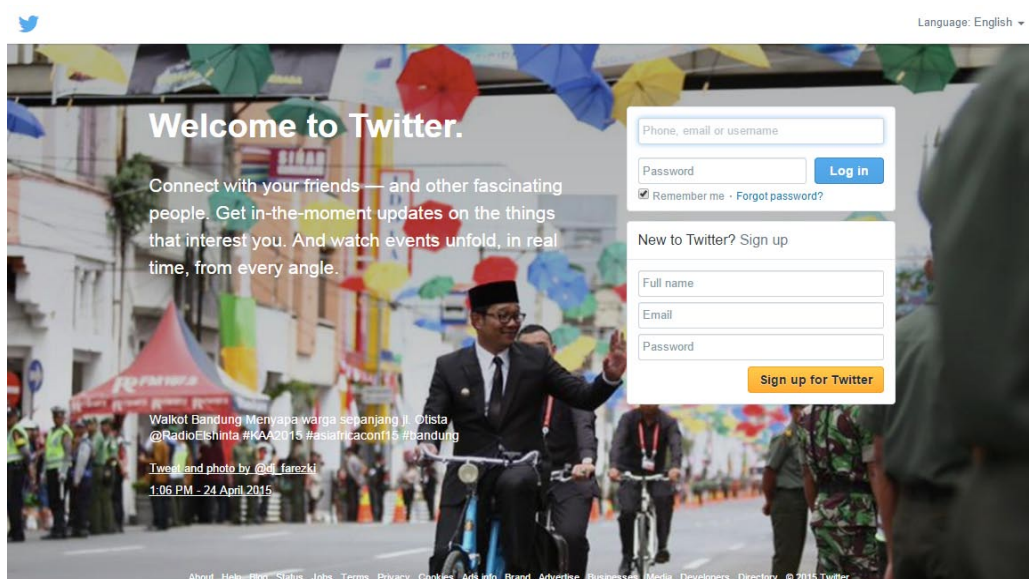


- Pengulangan merupakan cara lain yang bisa digunakan untuk menciptakan suatu kesatuan dalam desain. Elemen yang diulang dapat berupa warna, bentuk, garis, tekstur, bahkan objek. Dalam suatu brosur tentang alam, pengulangan warna hijau pada setiap halaman brosur akan memperkuat kesatuan pada desain. Contoh kesatuan dengan pengulangan:



3. Kontras

Kontras adalah metode yang paling umum digunakan untuk membuat penekanan. Konsepnya sederhana: semakin besar perbedaan antara elemen grafis dengan sekitarnya, semakin menonjol elemen tersebut. Kontras dapat dibuat dengan menggunakan perbedaan warna, ukuran, dan bentuk.



Homepage twitter tampaknya ingin sekali kita untuk mendaftar, atau login jika sudah punya akun, dengan menonjolkan tombol sign up dan log in.

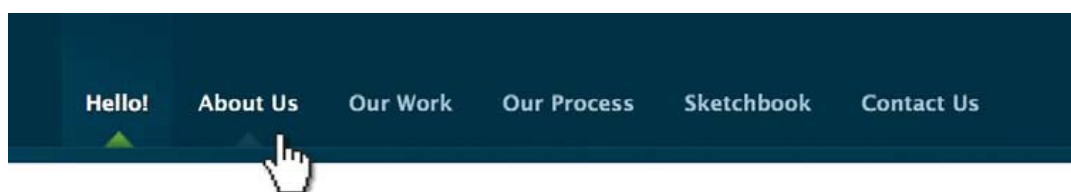
Cara Menyusun Layout

Banyak cara yang dapat dilakukan untuk menyusun layout halaman situs web. Salah satu cara yang dapat dilakukan adalah menggunakan menggunakan Photoshop untuk mendesain layout halaman-halaman situs. Kemudian menambahkan markup menggunakan bahasa HTML dan CSS. Tag yang digunakan untuk mengatur layout adalah tag <div>. Tag <div> digunakan untuk mengatur struktur halaman web, memasukkan komponen-komponen halaman seperti teks, paragraf, dan gambar pada halaman web. Kemudian dengan aturan-aturan CSS, termasuk penggunaan properti float, width, position, dan sebagainya, komponen-komponen tadi diatur posisi dan bentuk tampilannya.

Desain Situs Web yang Baik

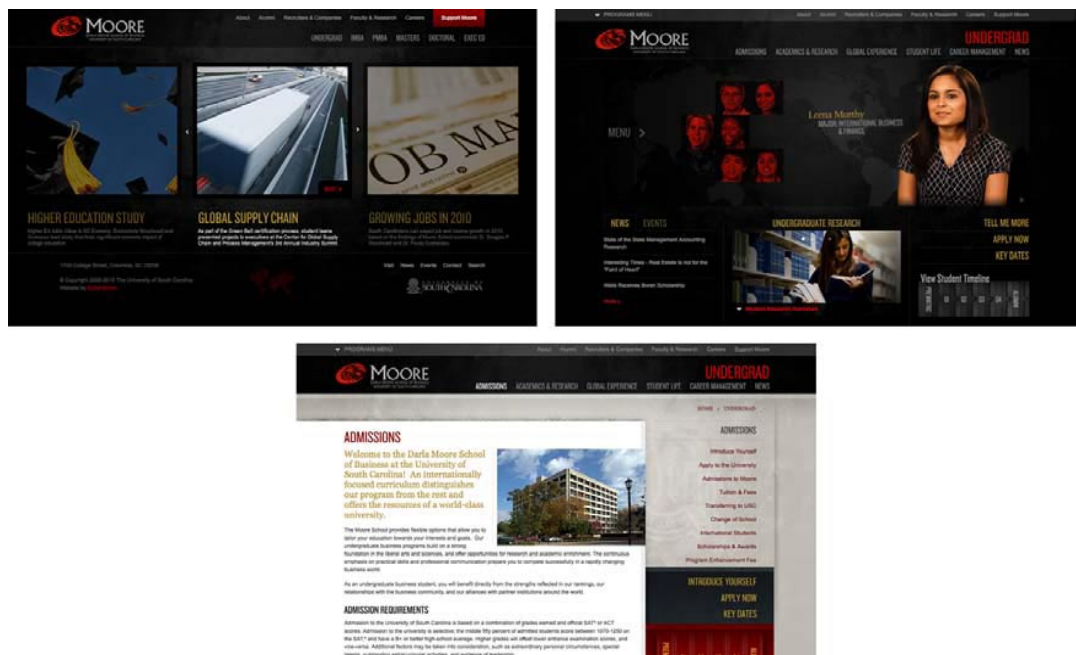
Ada dua perspektif utama dalam menentukan apakah suatu situs baik atau buruk. Pertama perspektif usability atau kegunaan, yang berfokus pada fungsionalitas situs, efektifitas penyajian informasi, dan efisiensi. Kedua, perspektif estetika murni, yang berkisar tentang nilai seni dan daya tarik visual dari desain. Beberapa desainer terjebak dalam estetika dan grafis dan mengabaikan pengguna, sementara desainer lain tersesat dalam pengujian kegunaan mereka dan melupakan daya tarik visual. Untuk itu penting untuk memaksimalkan keduanya. Selain itu, hal paling penting untuk diingat adalah bahwa desain itu adalah tentang komunikasi. Jika Anda membuat situs web yang bekerja secara efisien dan menyajikan informasi secara baik tetapi tampilannya tampak buruk, tidak ada seorangpun yang akan menggunakannya. Sebaliknya, jika situs web Anda cantik tetapi sulit untuk digunakan atau diakses, orang-orang akan meninggalkannya. Maka:

- Upayakan pengguna menyenangi desain dan tertarik dengan konten
Desain tidak boleh menjadi penghalang, melainkan harus bertindak sebagai penghubung antara pengguna dan informasi.
- Pengguna dapat bergerak dengan mudah menggunakan navigasi
Blok navigasi utama harus terlihat jelas pada halaman, dan setiap link harus memiliki judul yang deskriptif. Selain membuat link akan berubah tampilan saat kursor melayang di atasnya, juga dapat menunjukkan halaman yang aktif. Hal ini akan membantu pengguna untuk mengenali dimana mereka berada, dan bagaimana untuk pergi ke tempat yang mereka inginkan.



- Pengguna mengenali setiap halaman sebagai bagian situs

Desain setiap halaman situs harus dibuat seragam. Jika ada perbedaan dramatis pada halaman tertentu, perlu ada tema atau gaya yang mengikat yang tetap ada. Contoh dibawah ini meskipun blok konten pada halaman ini dibagi secara berbeda, ada beberapa indikator visual yang memungkinkan pengguna mengetahui bahwa ini adalah halaman dari situs yang sama. Komponen pemersatu disini adalah pengulangan identitas dan navigasi blok. Penggunaan yang konsisten dari palet warna (hitam, abu-abu, kuning, dan merah) juga membantu menyatukan halaman.

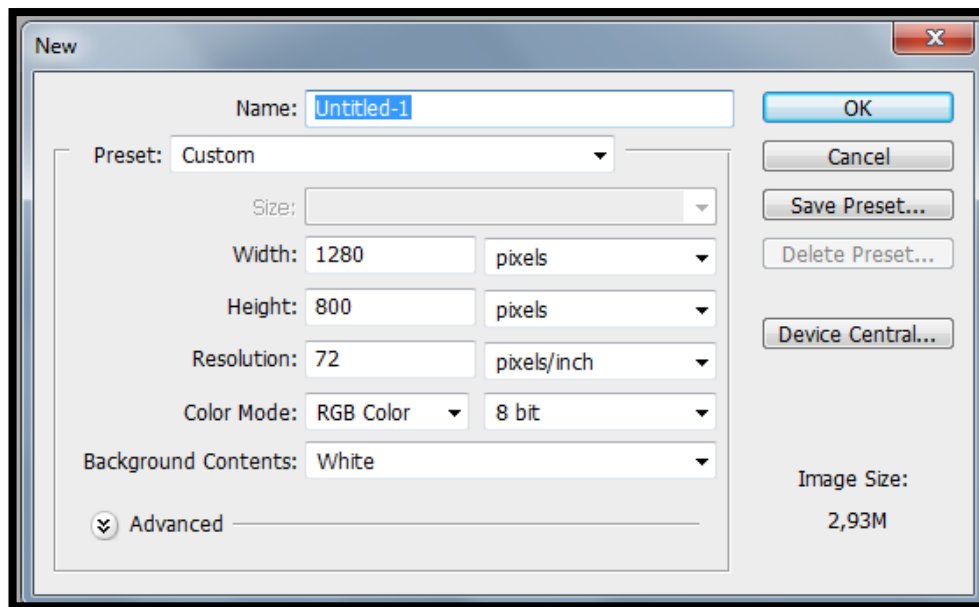


Praktikum

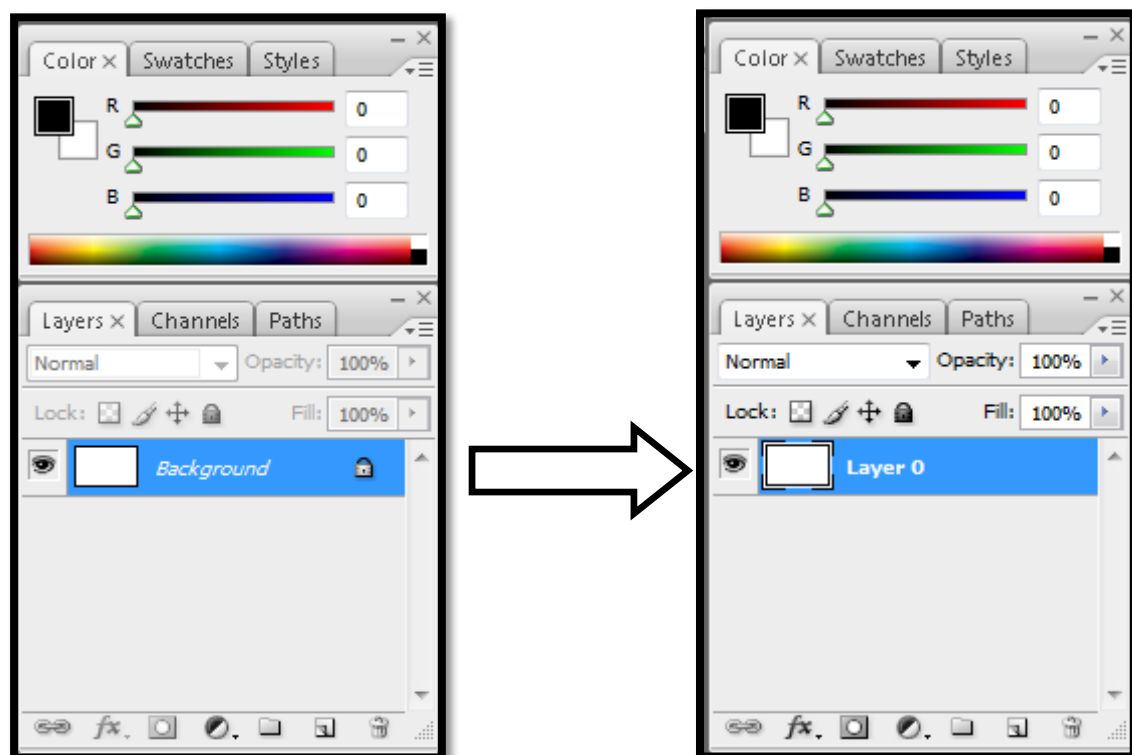
Dalam praktikum kali ini Anda akan membuat desain halaman situs web menggunakan adobe photoshop dan dreamweaver CS3. Jika menggunakan versi lain bisa saja, tetapi perlu sedikit penyesuaian atas langkah-langkah prakteknya.

Ikuti langkah-langkah praktikum berikut ini:

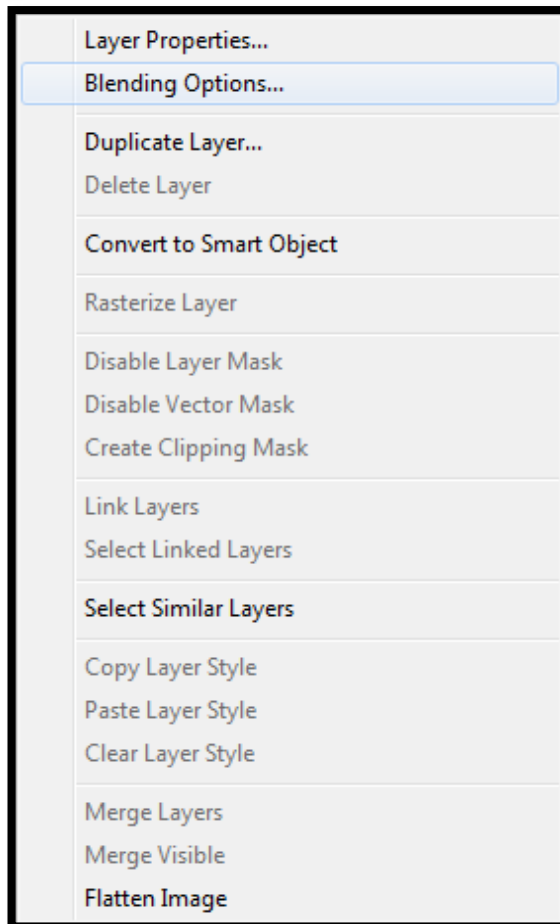
1. Buat file baru photoshop dengan ukuran 1280 x 800 piksel dengan background white, setingan lain tetap.



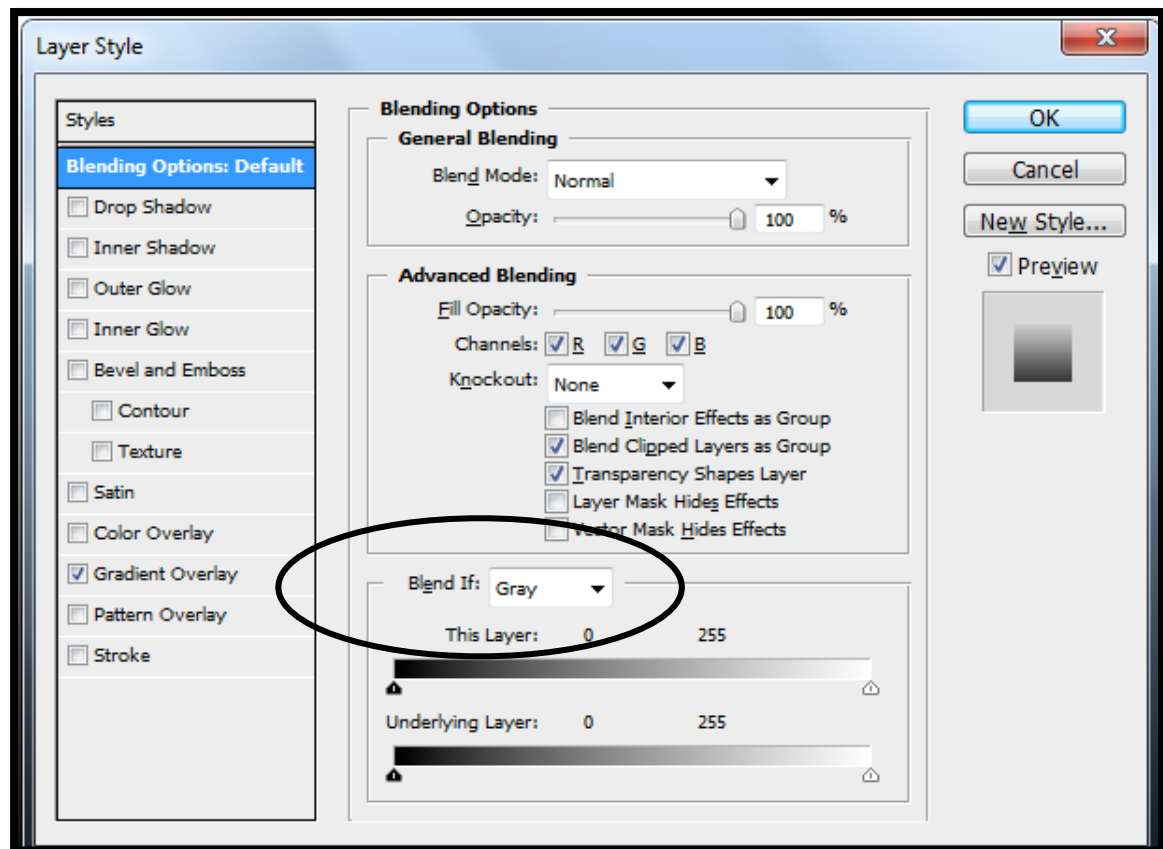
2. Double klik pada layer background yang terbentuk lalu langsung klik tombol OK, tujuannya untuk membuat layer background yang terkunci menjadi layer biasa.



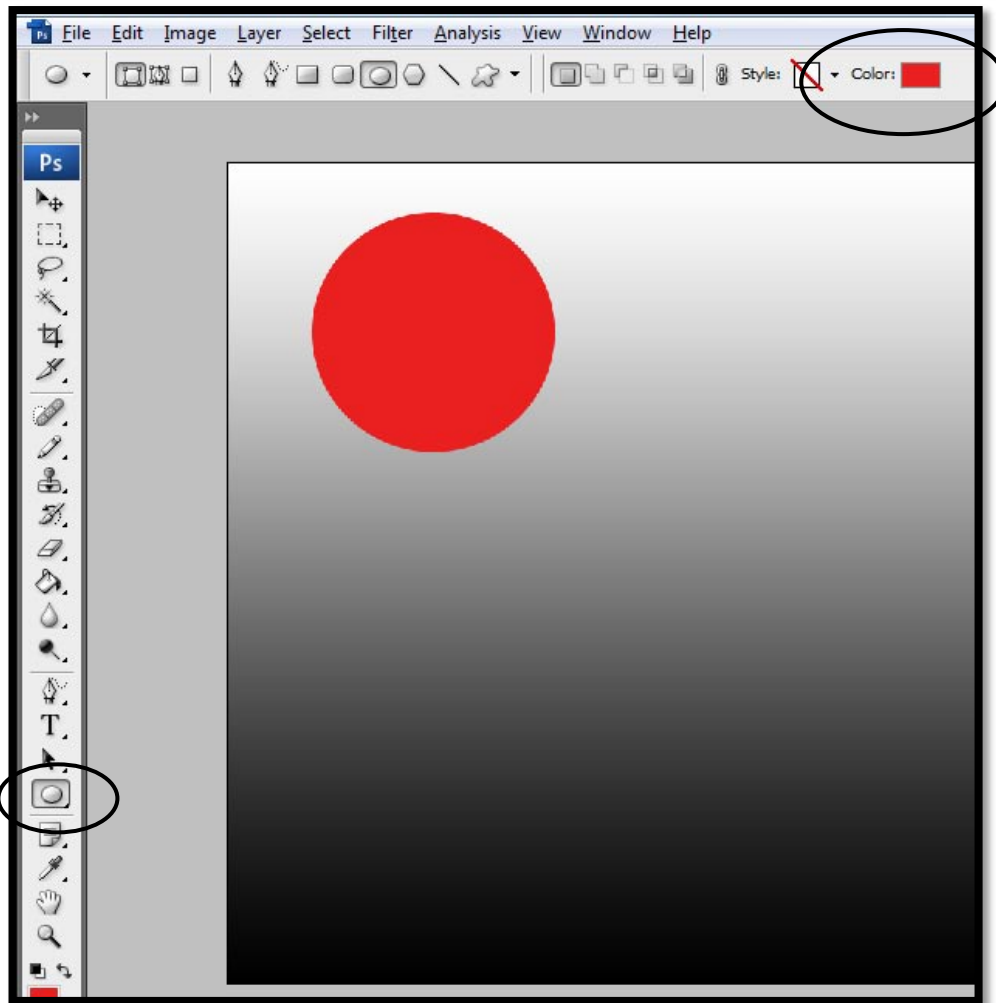
3. Klik kanan pada layer 0 dan pilih **Blending options...**



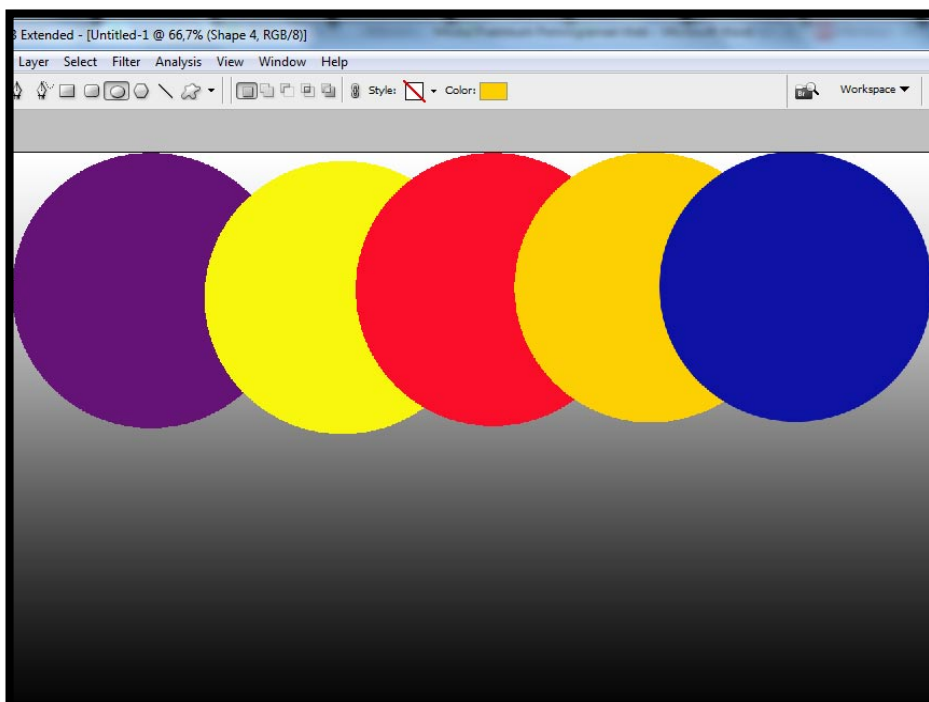
4. Pada kotak dialog blending options centang dan klik **Gradient Overlay** kemudian atur pilihan **Blend if** ke gray:



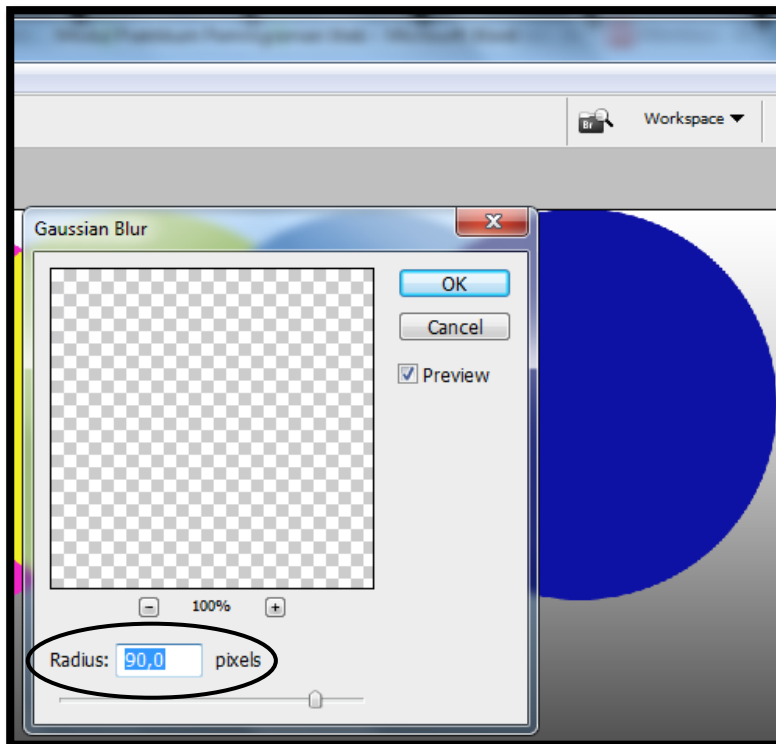
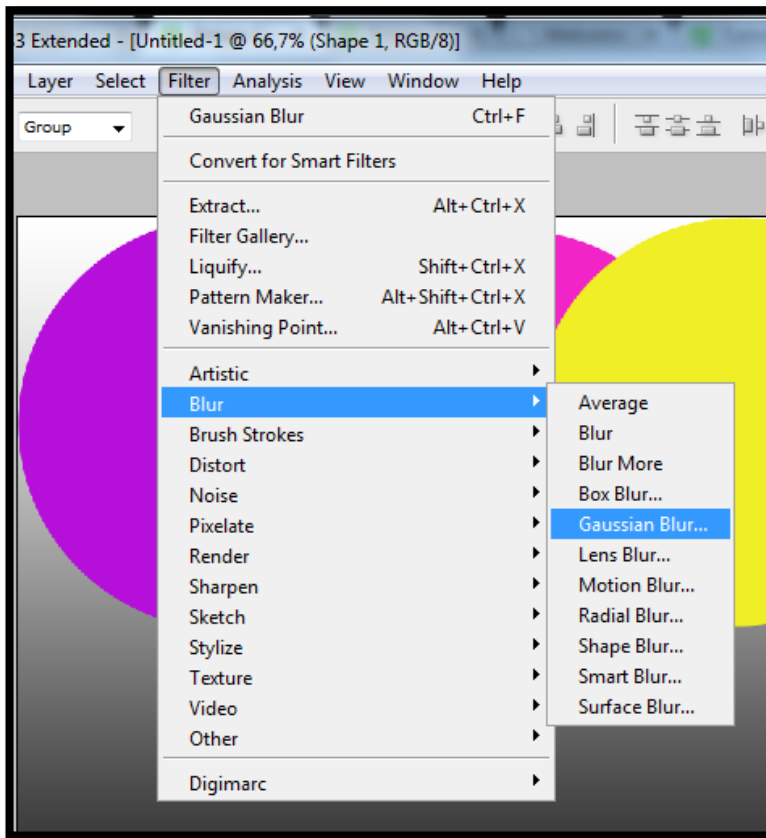
5. Pilih **Ellipse tool** kemudian buat bulatan pada layer 0 yang telah bergradasi hitam putih seperti berikut:



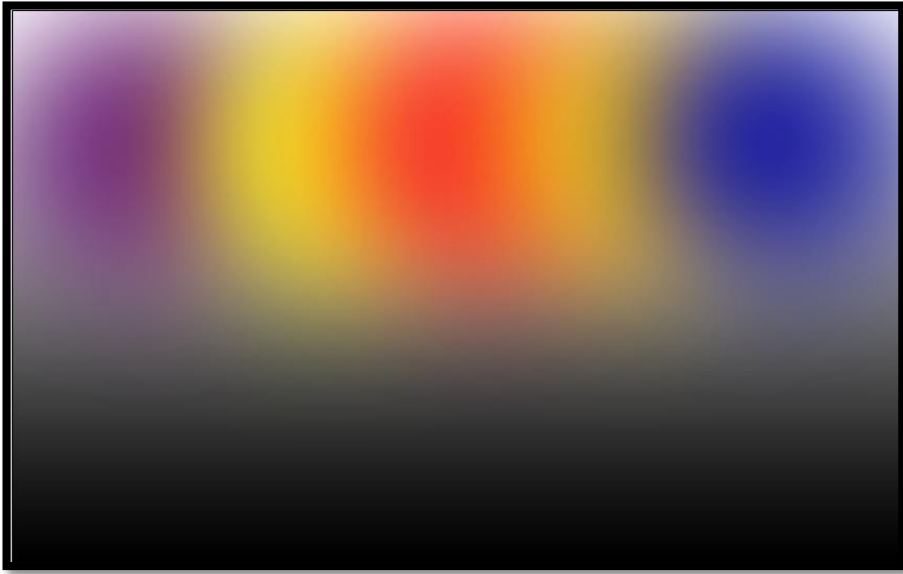
6. Buatlah bulatan yang sama sebanyak lima dengan warna yang berbeda:



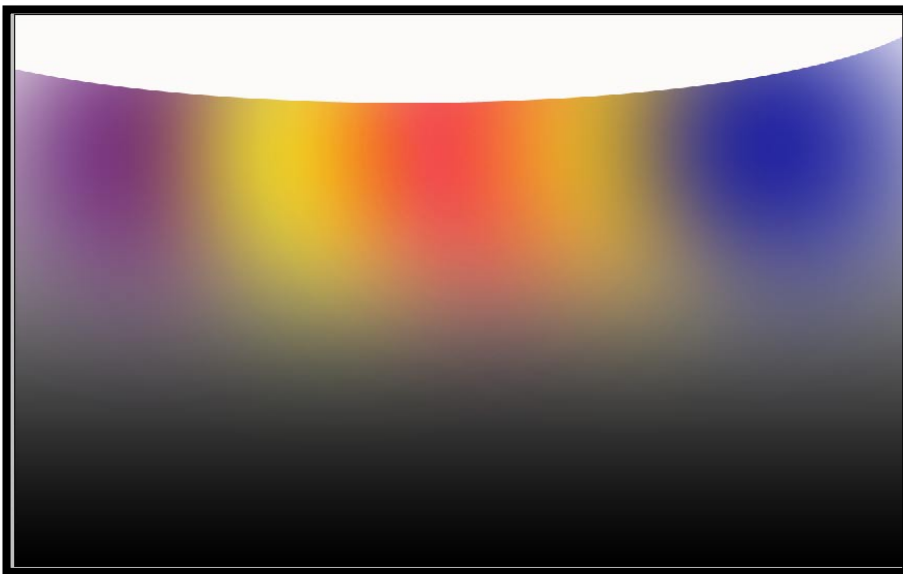
7. Klik salah satu bulatan kemudian pilih menu **Filter – Blur – Gaussian Blur**. Ubah radius Gaussian blur menjadi 90. Lakukan kemudian pada semua bulatan.



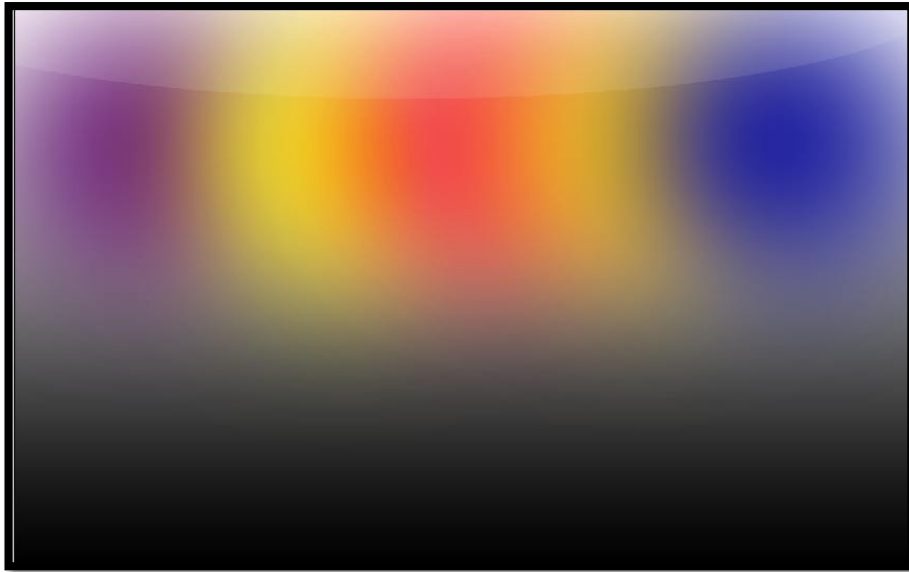
Hasilnya:



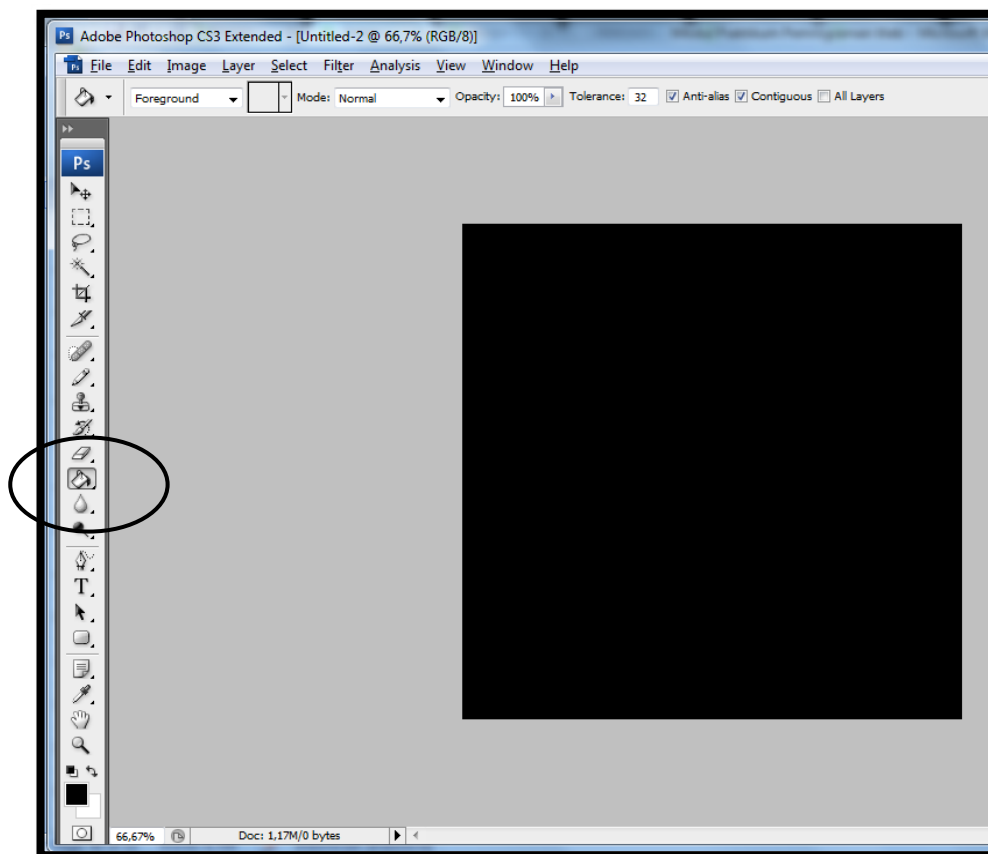
8. Buat lagi bulatan berwarna putih dengan Ellipse tool, kali ini agak lonjong dan diletakkan disebelah atas.



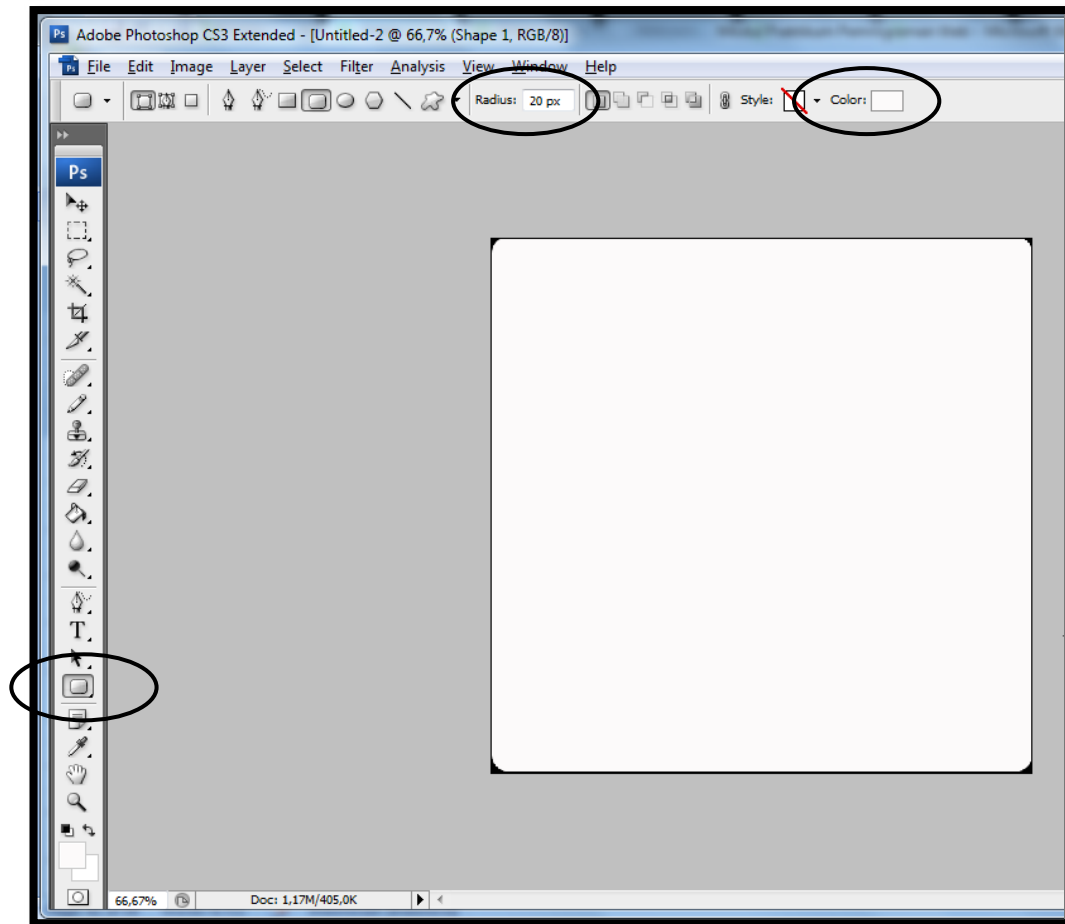
9. Atur **Fill Opacity**-nya pada blending option menjadi 10 sehingga menjadi seperti berikut:



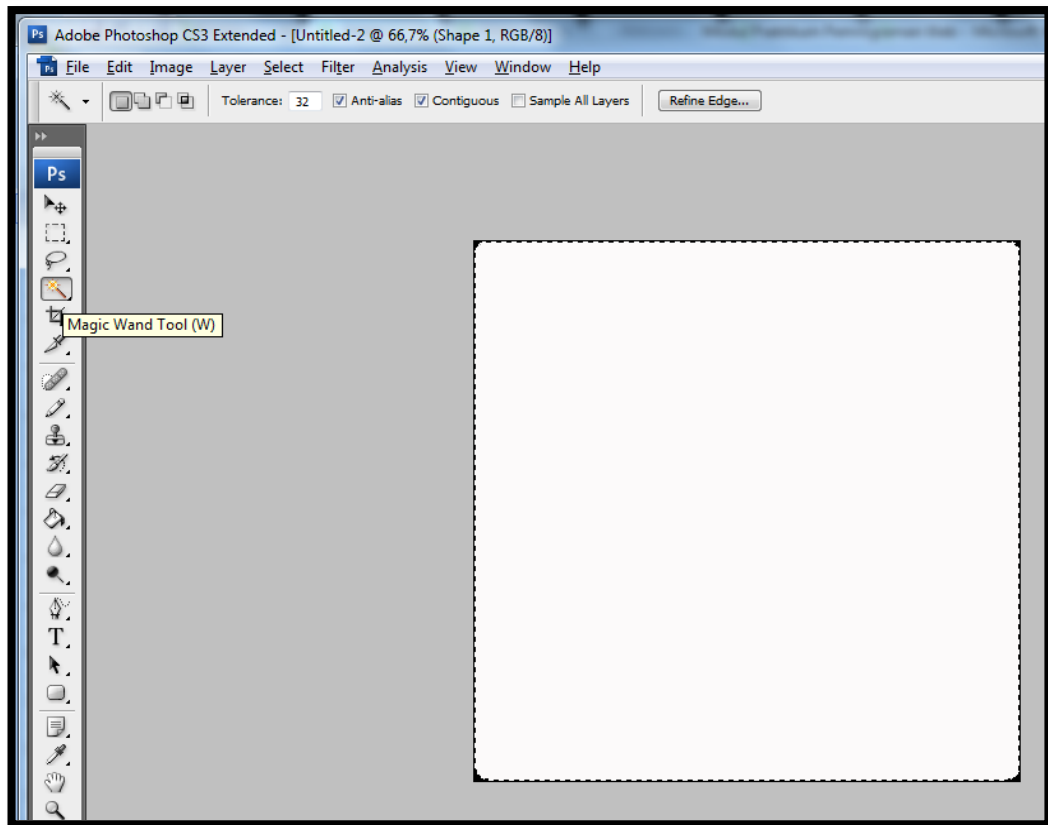
10. Sampai disini background untuk situs telah selesai dibuat. Selanjutnya membuat layout untuk konten situs. Buatlah file baru dengan ukuran 640 x 640 piksel. Beri warna latar belakang hitam dengan **Paint Bucket Tool**.



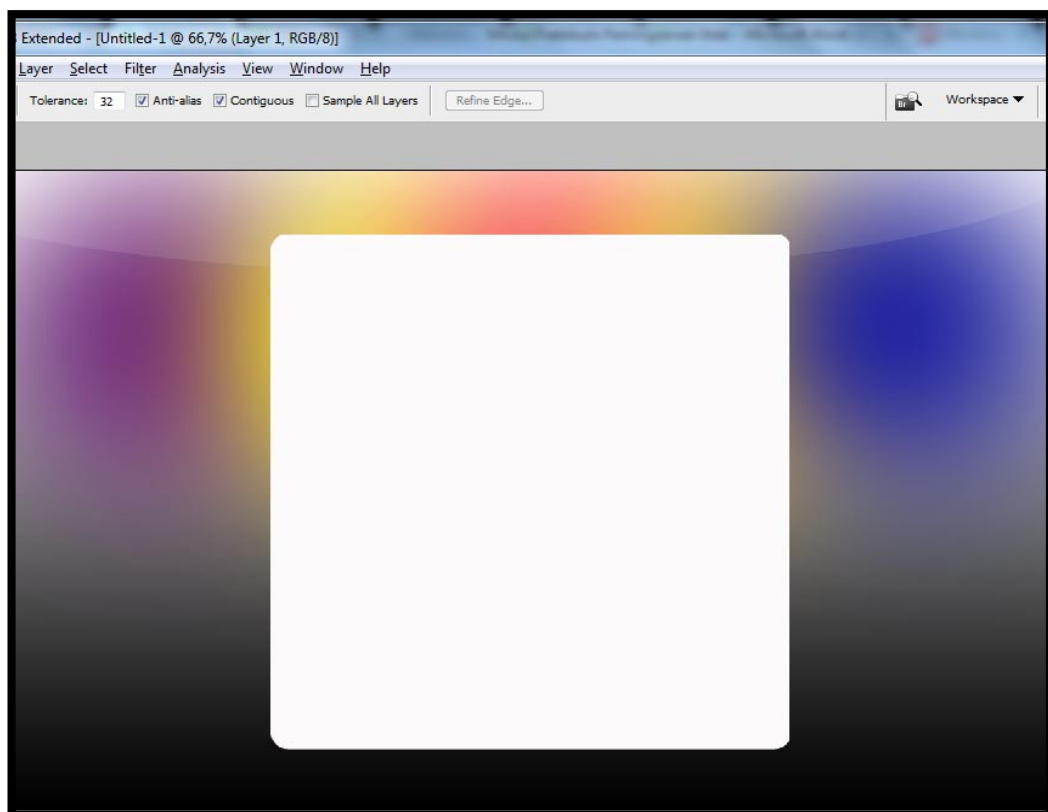
11. Kemudian pilih **Rounded Rectangle Tool** ubah radiusnya menjadi 20px lalu buat bidang berwarna putih.



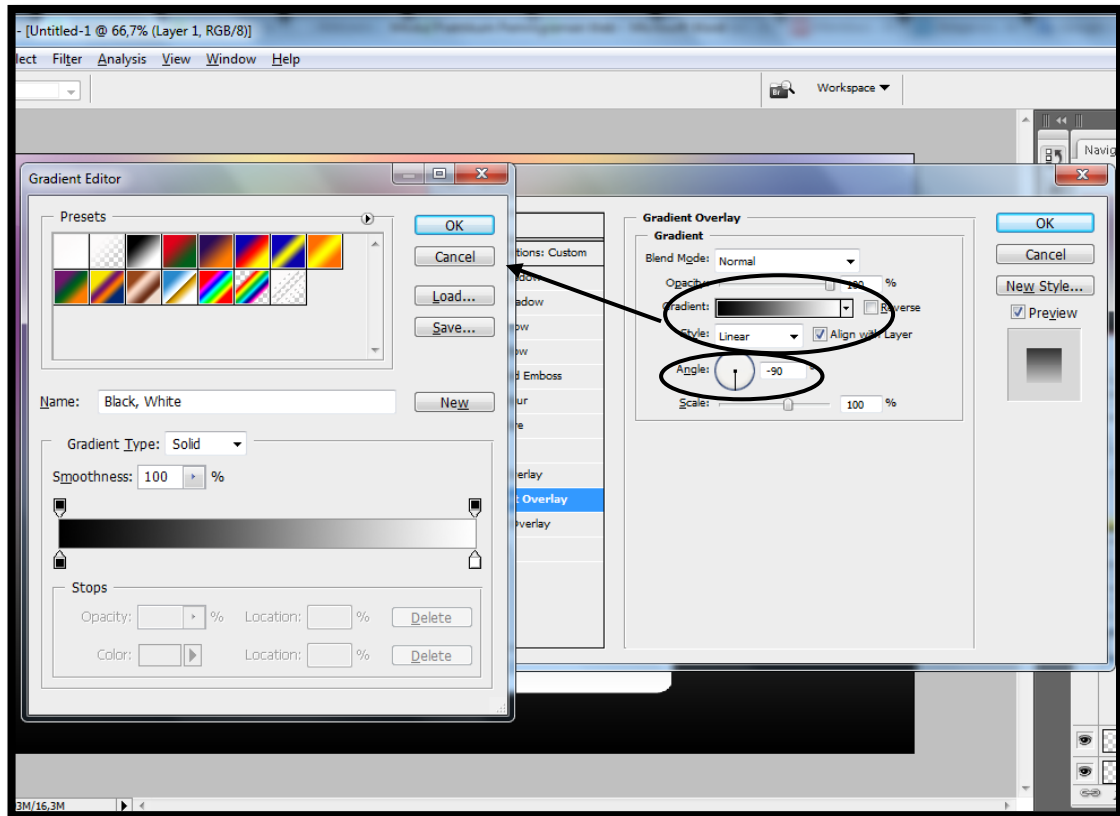
12. Dengan **Magic Wand Tool**, pilih bidang putih yang baru dibuat kemudian tekan Ctrl+C. Ini untuk menggandakan bidang putih ini.



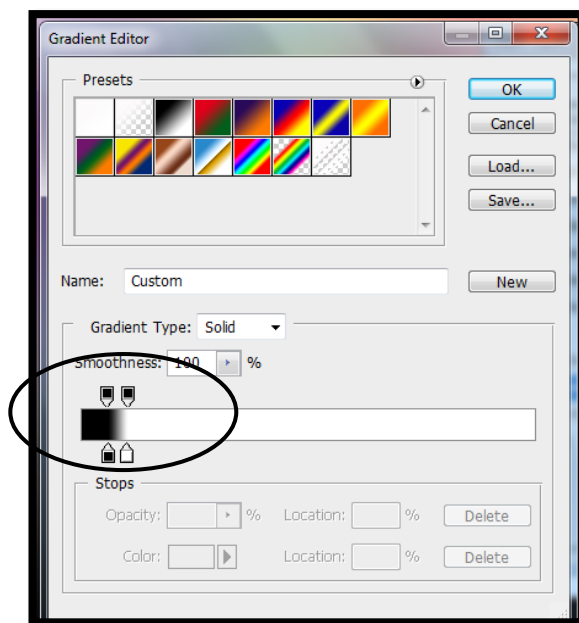
13. Menggunakan menu **Window**, pilih file background yang tadi lalu tekan Ctrl+V untuk menyalin bidang putih ke bidang background.



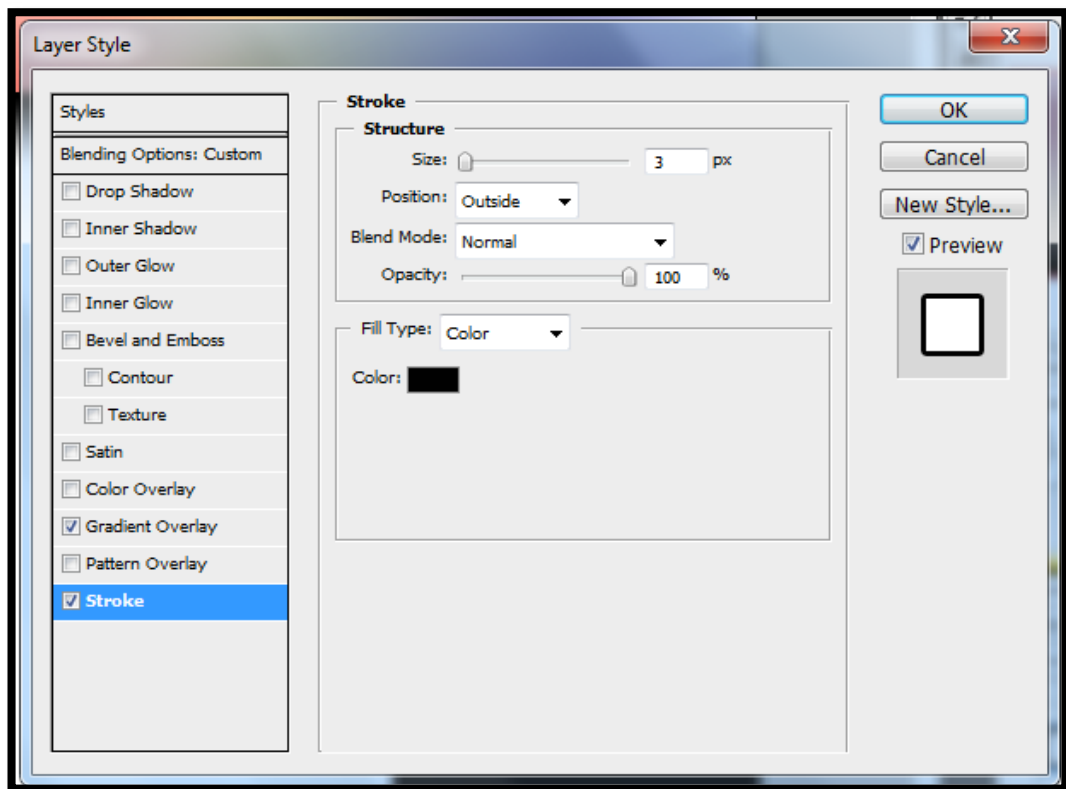
14. Pilih layer berisi bidang putih yang baru terbentuk lalu klik dua kali untuk membuka kotak dialog **Blending Options**. Tentukan Angle menjadi -90. Pilih dan centang pada **Gradient Overlay** lalu klik pada combo box **Gradient** untuk membuka kotak dialog **Gradient Editor**.



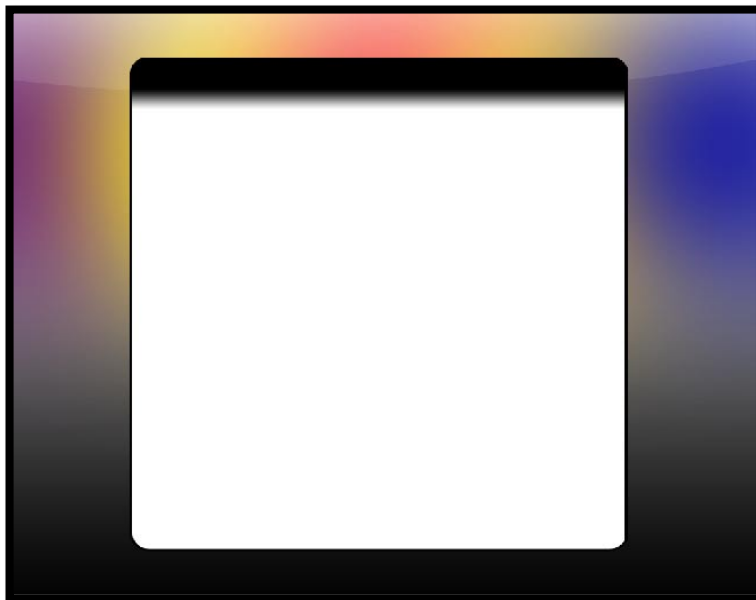
15. Atur slider gradient sehingga posisinya seperti terlihat pada gambar dibawah ini:



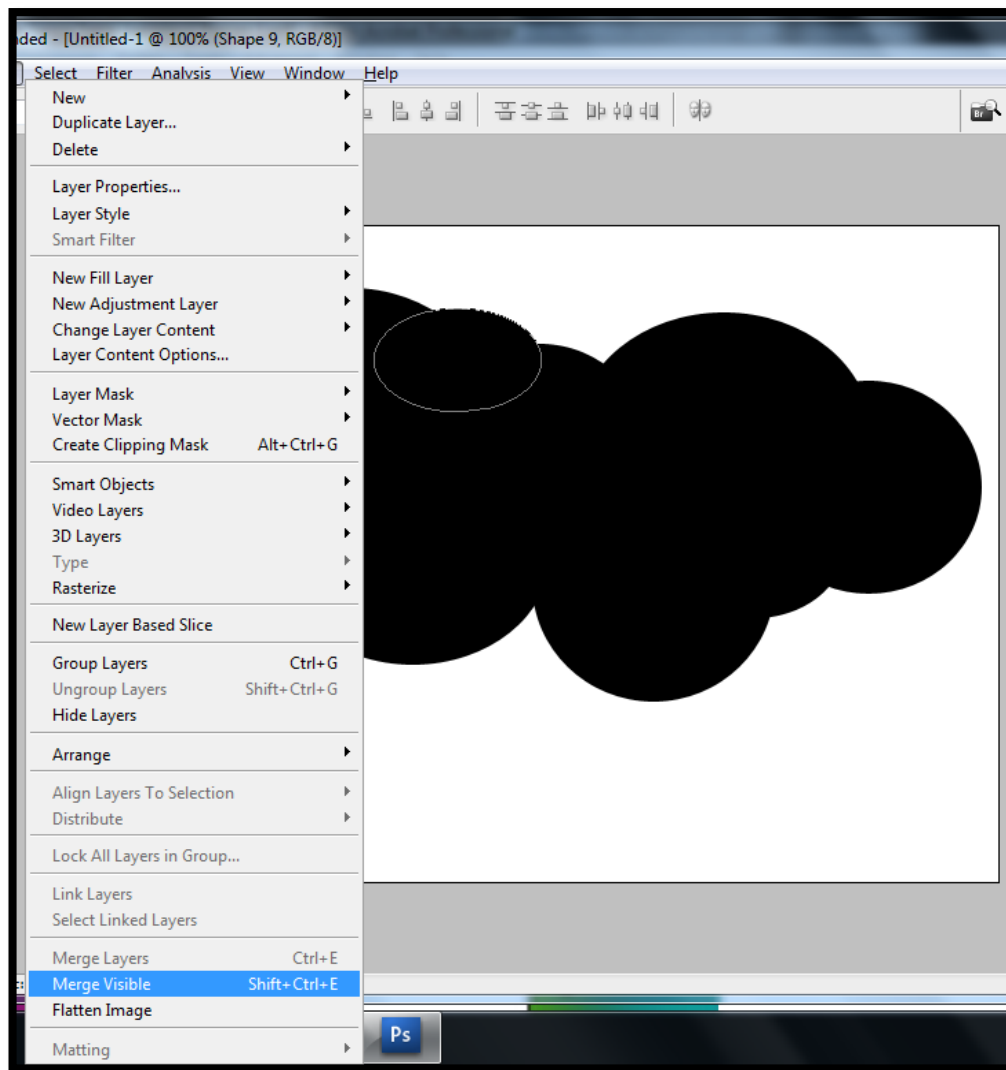
16. Buka lagi kotak dialog **Blending Options**, kali ini pilih dan centang **Stroke** lalu tentukan nilai-nilainya seperti ini:



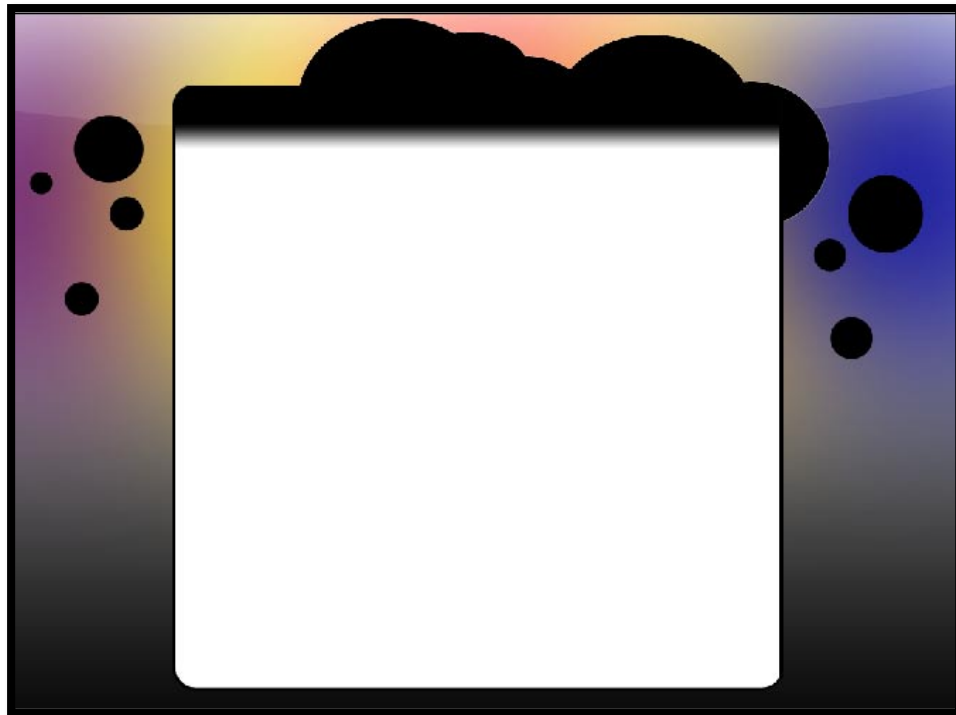
17. Hasil penambahan **Stroke** dan **Gradient Overlay** sebagai berikut:



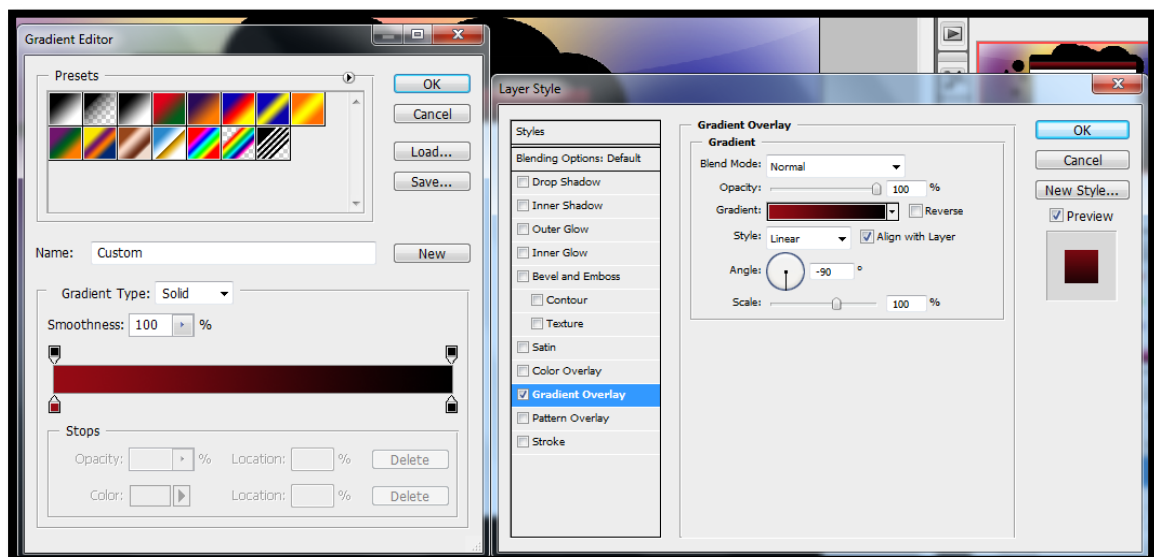
18. Selanjutnya buat sebuah file baru dan buat bulatan-bulatan hitam dengan **Ellipse Tool** sehingga membentuk gambar seperti awan hitam. Lalu pilih menu **Layer** kemudian pilih **Merge Visible** untuk menggabungkan layer-layer bulatan tadi menjadi satu.



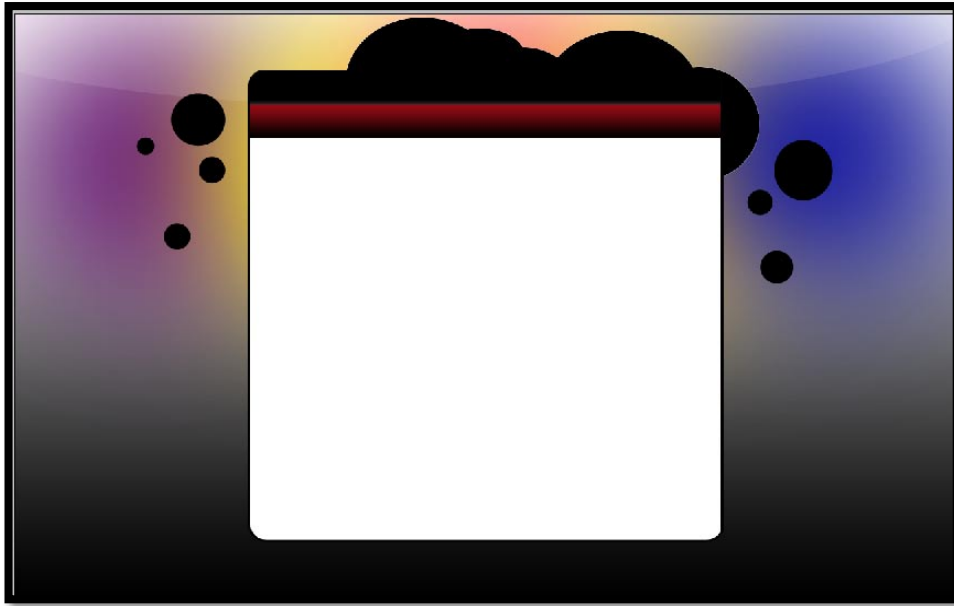
19. Gabungkan gambar awan hitam ini dengan desain background menggunakan cara salin dan gandakan pada pembuatan layout konten situs sebelumnya. Setelah digabungkan pada desain background, tambahkan bulatan-bulatan hitam kecil disekitarnya sehingga diperoleh desain seperti:



20. Sekarang desain background dan konten situs sudah jadi. Selanjutnya membuat desain menu. Langkah pertama buat bidang persegi dengan **Rectangle Tool** dan letakkan di bagian atas desain konten. Lebarnya selebar desain konten yang 640 piksel. Kemudian tambahkan **Gradien Overlay** pada **Blending Option** dengan variasi warna merah tua dan hitam. Caranya sama seperti sebelumnya.



Hasilnya:



21. Tambahkan sekali lagi bidang tetapi kali ini menggunakan **Rounded Rectangle Tool**. Letakkan di pojok kiri atas yang nanti akan menjadi kotak pencarian. Beri gradasi warna seperti sebelumnya dan tambahkan ikon pencarian diatas bidang. Selanjutnya dibagian atas tambahkan logo situs Anda sehingga bagian atas menu tampak seperti berikut:



Sampai disini layout situs sudah jadi denga hasil berikut:



22. Simpan file Photoshop layout situs Anda menggunakan menu **File – save**. Pastikan file Anda tersimpan dengan akhiran **.psd**.

Tugas

Buat desain untuk halaman homepage situs web personal yang sedang dikembangkan. Gunakan Adobe Photoshop seperti cara diatas. Gunakan imajinasi dan kreatifitas Anda. Semakin **berbeda** layout halaman homepage Anda dengan contoh layout praktikum ini, nilai yang didapat akan semakin tinggi! Simpan file layout Anda dengan nama bebas untuk digunakan pada praktikum berikutnya.

Praktikum 6

Bahasa Style CSS

Tujuan Pembelajaran

Mempraktekkan penggunaan CSS menggunakan perangkat lunak *authoring* untuk memarkup desain situs kedalam HTML dan CSS.

Pokok Bahasan

Penerapan CSS terhadap desain situs hasil rekayasa menggunakan Photoshop

Indikator Keberhasilan

Mahasiswa dapat menerapkan css untuk mengatur gaya tampilan halaman situs web menggunakan alat bantu *authoring*

Penjelasan

Ibarat manusia, CSS adalah pakaian dan aksesorisnya yang membuat penampilan seseorang menjadi menarik. Manusia tidak bisa memperindah tampilannya hanya menggunakan tubuhnya sendiri. Apakah Anda mau tubuh Anda dicat menyerupai pakaian sehingga tampak berbaju dan bercelana sementara Anda tidak mengenakan sepotong kainpun?

HTML tidak dapat memperindah tampilan suatu halaman web tanpa bantuan CSS dan ini sudah merupakan standar dari W3C. HTML memang memiliki berbagai atribut untuk mengatur beberapa aspek tampilan seperti warna latar belakang (*background*), tetapi sebaiknya tidak digunakan dan sebagai gantinya gunakan properti *background* CSS.

Cara Penerapan CSS

Terdapat 3 cara yang dapat digunakan:

1. Inline stylesheet. Aturan CSS ditulis langsung pada tag HTML yang akan diatur tampilannya menggunakan atribut *style*:

```
<p style="text-align:justify;">Paragraf ini diatur CSS secara inline</p>
```

2. Internal stylesheet. Aturan CSS ditulis pada bagian HEAD dokumen HTML menggunakan tag *<style>*:

```
<head>
  <title>CSS secara internal</title>
  <style type="text/css">
    P {text-align:justify;}
  </style>
</head>
<body>
  <p>Paragraf yang ini diatur CSS secara internal</p>
</body>
```

3. Eksternal stylesheet. Aturan CSS disimpan pada suatu file sehingga terpisah dari dokumen HTML. Kemudian tambahkan kode pemanggilan file CSS dalam dokumen HTML. Akhiran file CSS adalah .css:

File CSS (misalnya styleku.css) berisi:

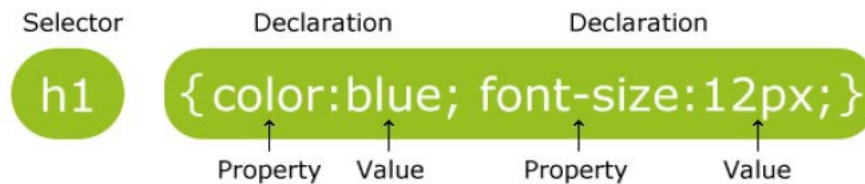
```
P {text-align:justify;}
```

Dokumen HTML berisi:

```
<head>
  <title>CSS secara eksternal</title>
  <link rel="stylesheet" type="text/css" href="styleku.css" />
</head>
<body>
  <p>Paragraf yang ini diatur CSS secara eksternal</p>
</body>
```

Sintaks

Sintaks penulisan CSS sebagai berikut:



Penjelasan:

Aturan CSS terdiri 2 bagian:

- **Selector**
Biasanya berupa tag HTML
- **Declaration**
Berisi aturan-aturan css yang terdiri dari properti dan nilainya yang dipisahkan oleh tanda titik dua. Setiap aturan css harus diakhiri dengan tanda titik koma.

Properti-properti CSS

Properti CSS jumlahnya sangat banyak, berikut beberapa diantaranya:

Properti	Fungsi	Nilai	Contoh
color	Mengatur warna teks	Nama warna, kode hexa warna (#ffffff:putih, #000000:hitam, #ff0000:merah), rgb(0,250,100)	Color:#ff5590;
Background-color	Mengatur warna latar	Nama warna, kode hexa warna, rgb(200,200,200)	Background-color:rgb(200,0,55);
Background-image	Mengatur gambar latar	Nama file gambar	Background-image:url(banner.jpg);
Text-align	Mengatur perataan teks	Left, right, center, justify	Text-align:justify;
Text-decoration	Mengatur dekorasi teks	Underline, none	Text-decoration:underline;
Line-height	Mengatur tinggi baris	Piksel, prosentase, em	Line-height:120%;
Font-family	Mengatur jenis font	"times new roman", arial, georgia	Font-family:arial;
Font-size	Mengatur ukuran karakter	Piksel, prosentase, em, pt	Font-size:12pt;
Margin			
Padding			
Border-width	Mengatur ketebalan garis batas	Piksel, prosentase, thin, thick	Border-width:1px;
Border-style	Mengatur jenis garis batas	Solid, dotted, dashed, double, none	Border-style:solid;
float	Mengatur obyek agar mengambang	Left, right, none	Float:left;
clear	Menghentikan	Left, right, both, none	Clear:both;

	pengembangan sebelumnya		
--	----------------------------	--	--

Kelas CSS

Disamping menggunakan tag HTML sebagai selektor, sering selektor juga berupa kelas. Kegunaan kelas adalah agar aturan CSS dapat diterapkan tidak hanya pada satu jenis tag HTML saja. Untuk membuat kelas CSS menggunakan tanda *titik* sebelum nama kelas. Contoh:

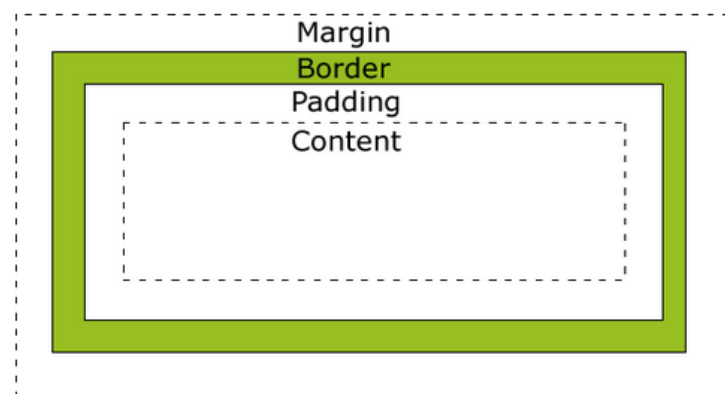
```
.kontenPenting {color:#ff00ff; font-size:150%;}
```

```
.....  
.....
```

```
<div class="kontenPenting">Universitas Dian Nuswantoro Menjadi Universitas  
Komputer nomor satu se-Indonesia</div>
```

Konsep Box-Model

Dalam CSS dikenal istilah 'Box Model'. Perhatikan gambar berikut ini:



CSS menggunakan konsep ini dalam mengatur tag-tag HTML. Pada gambar, bayangkan area 'Content' misalnya adalah sebuah paragraf. Obyek paragraf ini akan dianggap CSS memiliki area padding, border, dan margin disekitarnya. Keberadaan area-area ini berguna untuk pengaturan tata letak. Misalnya ingin diatur agar 2 buah gambar yang terletak berdampingan tidak terlalu rapat, maka kita dapat memperbesar lebar dari area margin agar jarak antara gambar lebih lebar.

Float dan Clearing

Suatu saat kita ingin membagi halaman web menjadi dua bagian kiri dan kanan masing-masing lebarnya 60% dan 40%. Untuk mengatur ini kita memerlukan properti *float* CSS. Properti ini akan membuat, seperti arti namanya, obyek HTML melayang. Karena melayang, maka obyek HTML (misalnya gambar) tidak terikat dengan ketentuan yang berlaku, seperti lebar layar browser, sehingga dapat diletakkan melewati batas layar browser, atau saling ditumpuk satu dengan lainnya.

Sedangkan proses clearing untuk menghentikan proses pengembangan akibat aturan CSS sebelumnya.

Contoh:

```
<div style="float:left; width:200px;"></div>
<div style="float:left; width:200px;"></div>
<div style="float:left; width:200px;"></div>
<div style="clear:both"></div>
<div style="float:left; width:200px;"></div>
```

Gambar ke-4 diatas tidak akan ditampilkan disisi kanan gambar ketiga karena ada properti clear, tetapi akan tampil dibaris selanjutnya.

CSS Shorthand

Properti CSS sangat banyak, beberapa merupakan satu keluarga properti, misalnya keluarga properti background. Untuk mempersingkat penulisan kode CSS dimungkinkan untuk menentukan seluruh properti tersebut dalam satu properti tunggal. Properti ini disebut properti shorthand. Properti shorthand untuk background yaitu *background*.

Contoh:

```
body {background:#ffff url('img_tree.png') no-repeat right top;}
```

Ketika menggunakan properti shorthand urutan properti harus diperhatikan, tidak boleh terbolak-balik. Untuk background, urutan propertinya adalah:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

Tidak masalah jika hanya beberapa dari nilai properti tersebut yang digunakan, tetapi penulisannya tetap harus mengikuti urutan diatas.

Tips

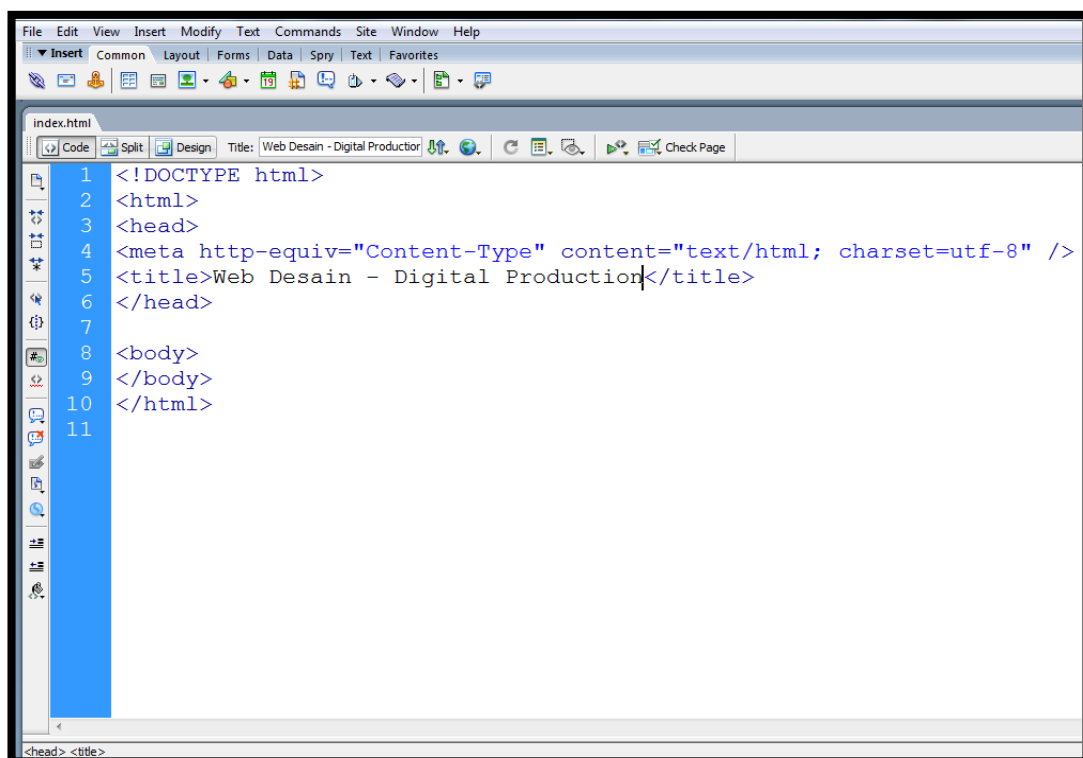
Menerapkan CSS pada situs web tidak bisa dilakukan dengan sembarangan karena akan menimbulkan masalah perawatan situs web dikemudian hari (susah untuk memperbaiki, memodifikasi, atau meningkatkan tampilan situs web). Hal yang perlu Anda perhatikan adalah

mulailah mengatur tampilan situs web Anda dari yang umum ke khusus. Maksudnya disini mulailah mengatur tampilan tag <body> terlebih dahulu sebelum mengatur bagian header, footer, penampilan paragraf, gambar, dan seterusnya. Sebagai panduan, pelajari tentang susunan aturan CSS yang baik di lampiran modul praktikum ini (**lampiran 1**).

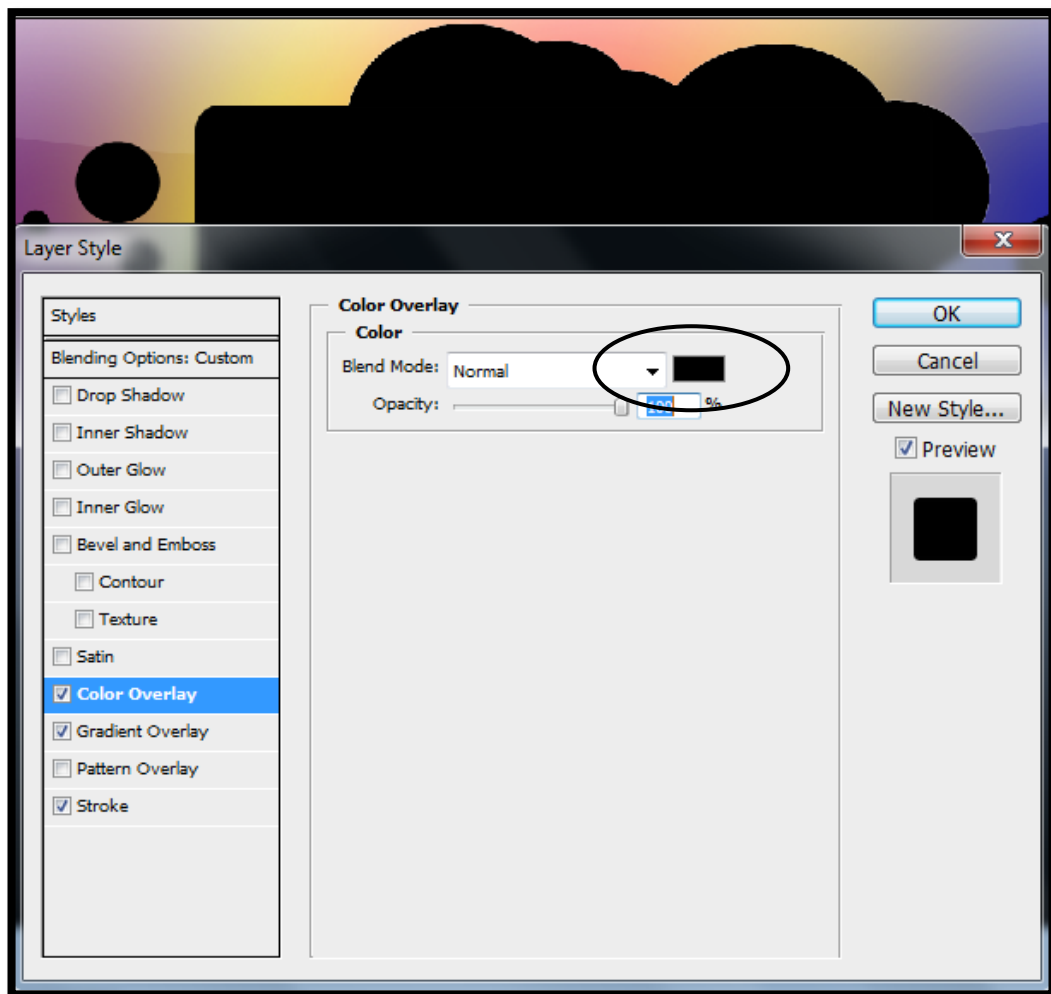
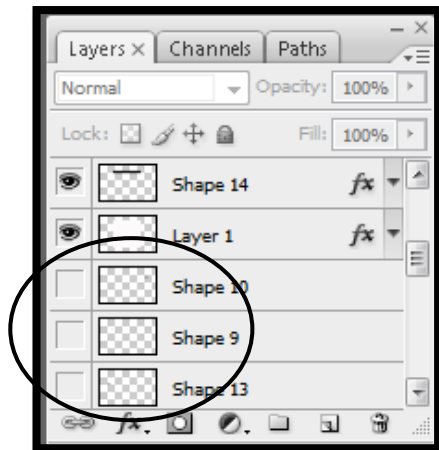
Praktikum

Pada praktikum sebelumnya Anda telah membuat tugas untuk membuat layout situs yang sedang Anda kembangkan menggunakan Photoshop. Pada praktikum ini file photoshop berisi layout situs Anda tersebut akan digunakan untuk dimarkup kedalam HTML dan CSS. Untuk melakukan markup kita akan menggunakan software Dreamweaver. Ikuti langkah-langkah berikut ini:

1. Buka Deamweaver dan buat sebuah file HTML lalu simpan dengan nama index.html
2. Ubah DOCTYPE dokumen HTML tersebut menjadi <!DOCTYPE html>. Ganti judul pada tag <title> sesuai dengan keinginan Anda.



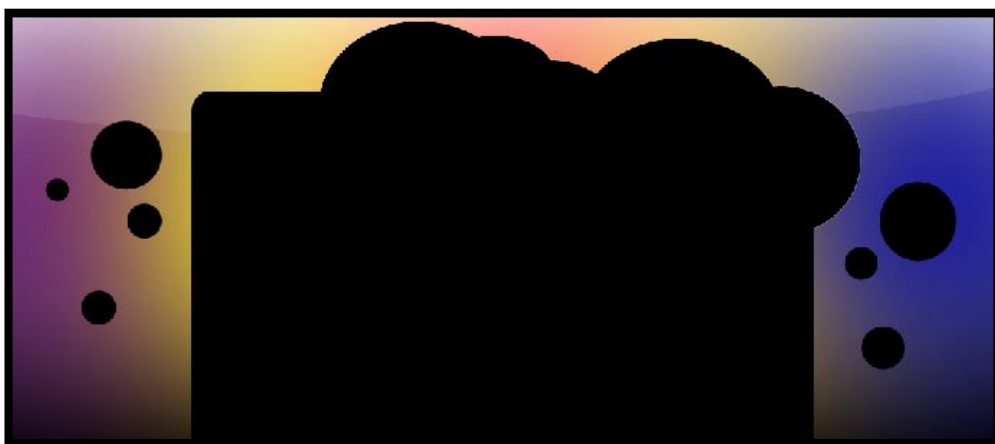
3. Buka file layout situs Photoshop Anda. Kita akan membuat gambar latar belakang untuk situs kita nanti. Untuk keamanan, buat salinan file layout photoshop Anda dengan menyimpan menggunakan menu **File – Save as**. Gunakan salinan file ini untuk langkah-langkah selanjutnya. Sembunyikan semua image menu. Uncentang tanda mata disetiap layer image menu. Pada **Blending Option** untuk layer layout konten (layer yang berisi kotak konten), centang dan ubah color overlay menjadi hitam pekat (#000000).



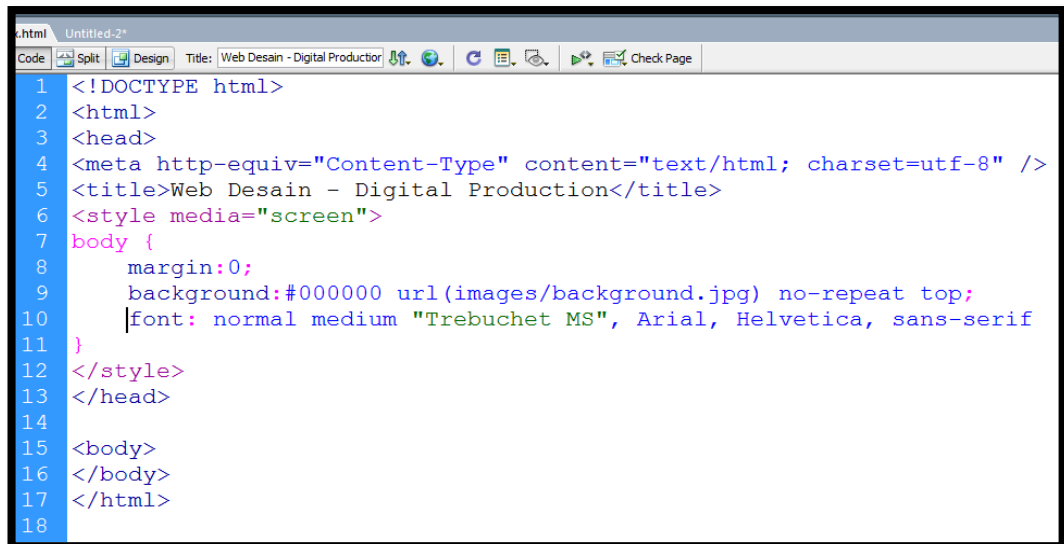
Hasilnya:



4. Potong atau crop gambar sehingga ukuran filenya dalam kb menjadi lebih kecil. Buat persis seperti dibawah ini dan pastikan layout hitam berada ditengah-tengah.

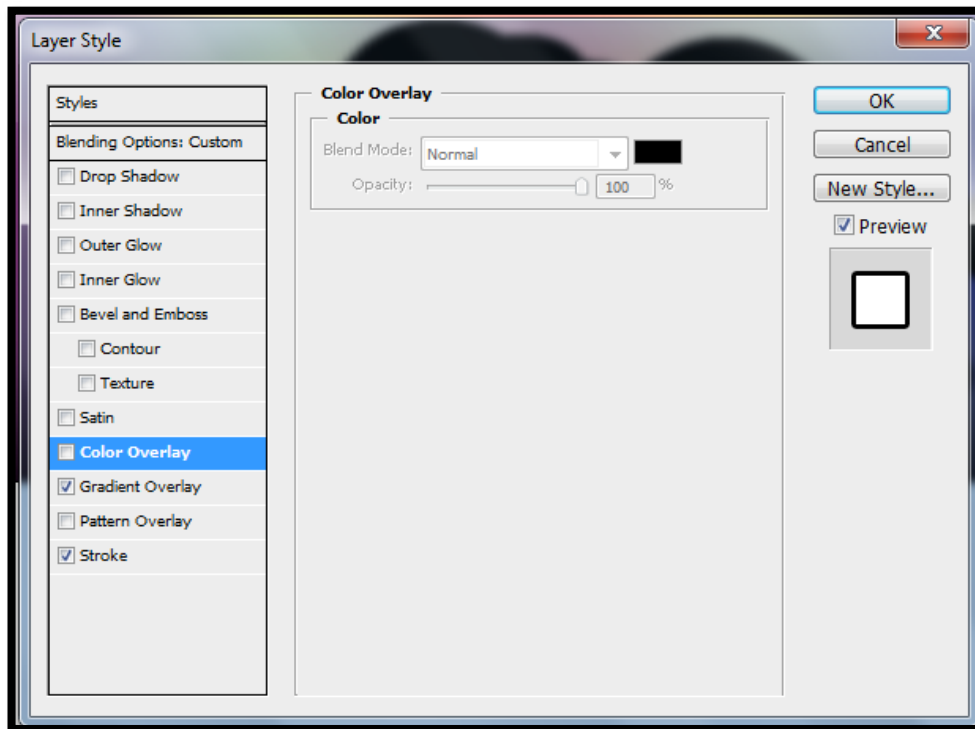


5. Buat direktori dengan nama images di lokasi file index.html berada. Simpan gambar menggunakan **menu File – Save for Web & Device**. Simpan sebagai file JPG dengan nama **background.jpg**.
6. Pindah ke Dreamweaver, buka file index.html. Kita akan menerapkan aturan css untuk tag `<body>`. Untuk sementara ini, css kita gunakan dengan cara internal stylesheet agar lebih mudah. Nanti jika sudah selesai, baru pindahkan aturan-aturan css ke file terpisah. Dibawah tag `<title>` tambahkan aturan css berikut:

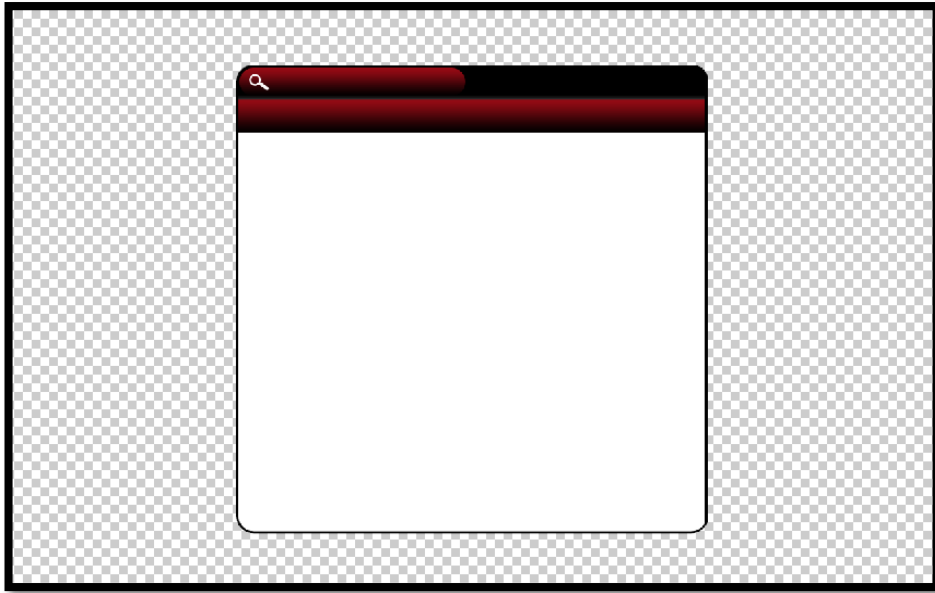


```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <title>Web Desain - Digital Production</title>
6 <style media="screen">
7 body {
8     margin:0;
9     background:#000000 url(images/background.jpg) no-repeat top;
10    |font: normal medium "Trebuchet MS", Arial, Helvetica, sans-serif
11 }
12 </style>
13 </head>
14
15 <body>
16 </body>
17 </html>
18
```

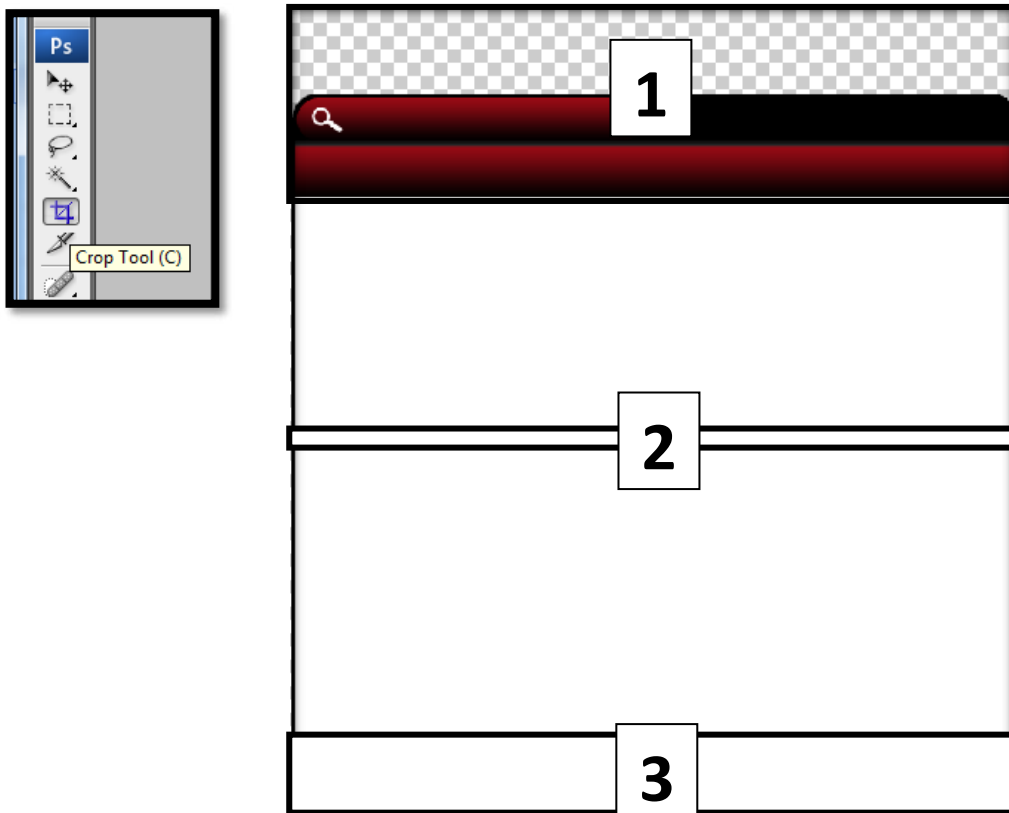
7. Pindah ke Photoshop, kembalikan layout konten menjadi putih kembali dengan menghilangkan centang pada **Color Overlay** (pada layer berisi layout konten).



8. Sembunyikan layout background sehingga diperoleh gambar seperti ini:



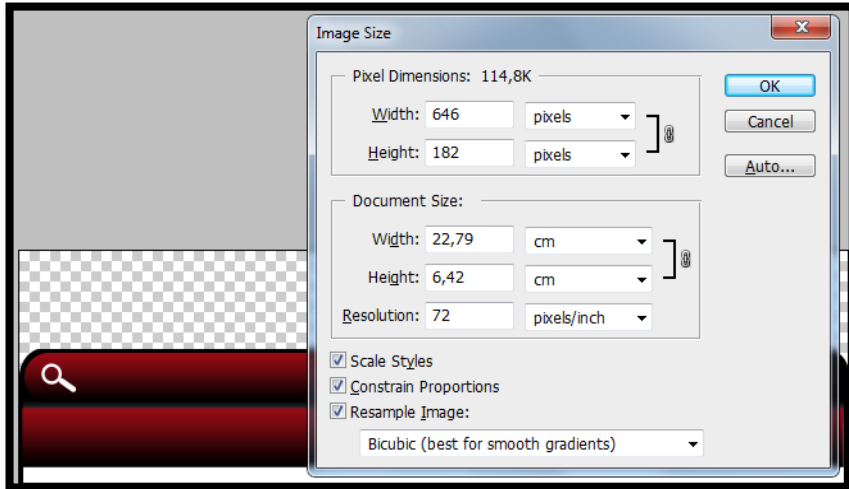
9. Klik kanan pada layer berisi layout konten, lalu pilih **Merge Visible** untuk menggabungkan semua layer menjadi satu. Ambil 3 potongan gambar seperti gambar dibawah ini. Untuk memotong gunakan Crop Tool. Potong satu demi satu.



10. Simpan masing-masing bagian sebagai file **PNG** pada direktori **images** dengan nama:
- Bagian pertama simpan dengan nama **bgheader.png**
 - Bagian kedua dengan nama **bgkonten.png**

- Bagian ketiga dengan nama **bgfooter.png**

11. Catat ukuran masing-masing potongan karena akan digunakan pada aturan css berikutnya. Untuk melihat ukuran masing-masing potongan, buka setiap potongan pada Photoshop. Kemudian pilih menu Image – Image size. Lihat pada bagian Pixel Dimensions.



Untuk gambar bgheader.jpg tampak ukurannya adalah 646 x 182 piksel.

Berikutnya, tambahkan aturan-aturan css tulis setelah aturan css untuk tag body. Kode css setelah ini adalah:

```
<style media="screen">
body {
    margin:0;
    background:#000000 url(images/background.jpg) no-repeat top;
    font: normal medium "Trebuchet MS", Arial, Helvetica, sans-serif
}
.header{
    width: 646px;
    height: 182px;
    margin: 0 auto;
    background:url(images/bgheader.png) bottom right no-repeat;
}
.content{
    width: 646px;
    margin: 0 auto;
    background:url(images/bgcontent.png);
}
.container{
    padding : 10px;
}
.footer{
    width: 646px;
    height: 48px;
    margin: 0 auto;
    background:url(images/bgfooter.png) top right no-repeat;
}
</style>
```

Nilai properti **width** dan **height** sesuai ukuran potongan gambar Anda masing-masing.

12. Tambahkan tag HTML berikut pada bagian body dokumen:

```
<div class="header"></div>
<div class="content">
    <div class="container"> </div>
</div>
<div class="footer"></div>
```

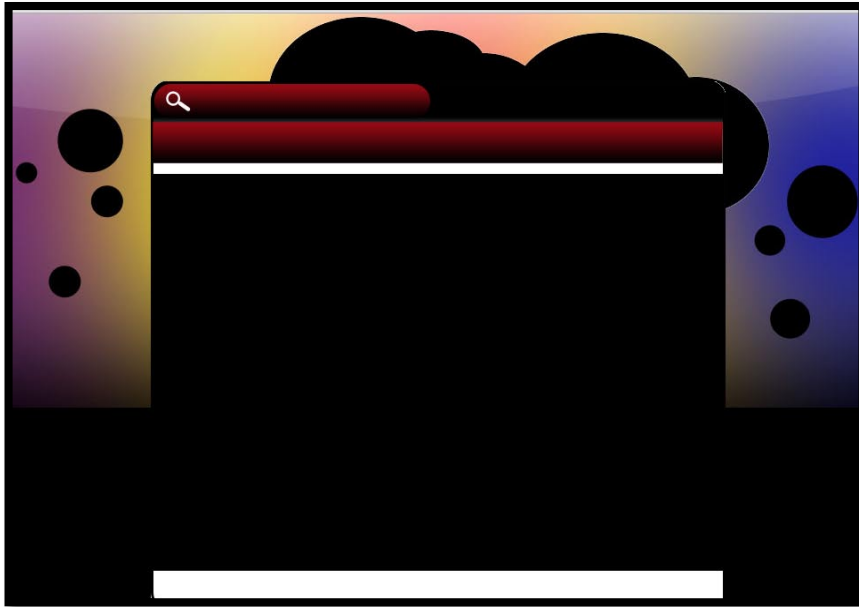
13. Dengan demikian maka untuk sementara dokumen HTML berisi kode sebagai berikut:

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Web Desain - Digital Production</title>
<style media="screen">
body {
    margin:0;
    background:#000000 url(images/background.jpg) no-repeat top;
    font: normal medium "Trebuchet MS", Arial, Helvetica, sans-serif
}
.header{
    width: 646px;
    height: 182px;
    margin: 0 auto;
    background:url(images/bgheader.png) bottom right no-repeat;
}
.content{
    width: 646px;
    margin: 0 auto;
    background:url(images/bgcontent.png);
}
.container{
    padding : 10px;
}
.footer{
    width: 646px;
    height: 48px;
    margin: 0 auto;
    background:url(images/bgfooter.png) top right no-repeat;
}
</style>
</head>

<body>
    <div class="header"></div>
    <div class="content">
        <div class="container"> </div>
    </div>
    <div class="footer"></div>
```

```
</body>
</html>
```

14. Jika dokumen HTML ditampilkan pada browser maka hasilnya:



15. Konten nanti akan ditulis didalam **kelas container**. Sebagai contoh, berikut dimasukkan beberapa teks ke kelas kontainer.

```
<div class="container">
  <h1>Tips</h1>
  <p>Menerapkan CSS pada situs web tidak bisa dilakukan dengan
sembarangan karena akan menimbulkan masalah perawatan situs web
dikemudian hari (susah untuk memperbaiki, memodifikasi, atau
meningkatkan tampilan situs web). Hal yang perlu Anda perhatikan
adalah mulailah mengatur tampilan situs web Anda dari yang umum ke
khusus. Maksudnya disini mulailah mengatur tampilan tag <body>
terlebih dahulu sebelum mengatur bagian header, footer, penampilan
paragraf, gambar, dan seterusnya. Sebagai panduan, pelajari tentang
susunan aturan CSS yang baik di lampiran modul praktikum ini
(lampiran 1).</p>
</div>
```

Hasilnya:



16. Selanjutnya buat markup untuk input pencarian. Tambahkan kode berikut pada tag div kelas "header":

```
<div class="header">
  <div class="left">
    <form name="search_form" action="/* link untuk proses */"
      method="post">
      <input type="text" name="search" id="search" class="search"
        value="Pencarian" />
      <input type="submit" name="submit" id="submit"
        class="submit" />
    </form>
  </div>
</div>
```

17. Pada bagian CSS tambahkan kode berikut:

```
.header .left{
  width: 230px;
  float:left;
  height: 30px;
  padding:68px 0 0 40px;
  color:#FFFFFFF;
}.
header .left .search{
  width: 250px;
  height: 14px;
  padding:3px 5px;
  color:#aaa;
}.
```

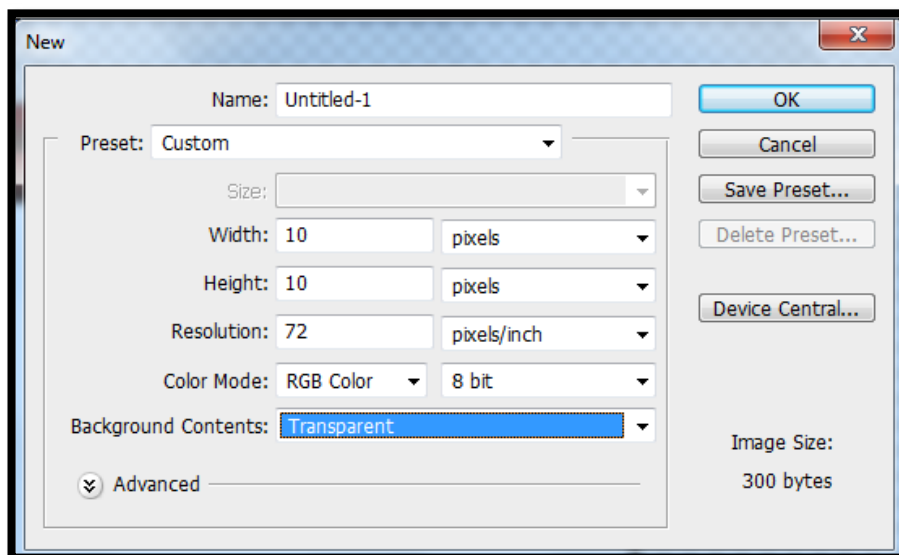
```

header .left .submit{
    display: none;
}

```

Aturan CSS diatas untuk memberi gaya CSS pada kelas “left”, “search”, dan “submit”. Jika ketika ditampilkan posisi kotak pencarian kurang tepat dengan background, sesuaikan nilai-nilai properti width, dan padding.

18. Buat sebuah file Photoshop dengan ukuran 10px x 10px, atur Background Contents menjadi Transparent, langsung simpan dengan nama **transparent.gif**. Apabila file gambar ini ditampilkan di browser maka gambar file ini tidak akan terlihat. Tapi karena tidak terlihat bukan berarti tidak berguna, ini akan dijadikan background untuk elemen tertentu seperti input sehingga tampilannya juga menjadi transparan tidak putih.



19. Tambahkan file ini sebagai background pada kelas “header .left .search”.

```

.header .left .search{
    background:url(images/transparent.gif);
    border:0;
    width: 250px;
    height: 14px;
    padding:3px 5px;
    color:#aaa;
}

```

Hasilnya kotak pencarian menjadi transparan:



Selanjutnya buat logo kreasi Anda sendiri dengan ukuran yang disesuaikan, beri nama logo.png, lalu tambahkan kedalam tampilan dengan menambahkan kode berikut setelah tag div kelas "left" pada bagian html:

```
<div class="left">
    <form name="search_form" action="/* link untuk proses */"
    method="post">
    <input type="text" name="search" id="search" class="search"
    value="Pencarian" />
    <input type="submit" name="submit" id="submit" class="submit" />
    </form>
</div>
<div class="right">
    <a href="#"></a>
</div>
```

20. Tambahkan kelas "right"

```
.header .right{
    width:225px;
    padding:30px 60px 0 0;
    float:right;
}.
clear{ clear:both; }
```

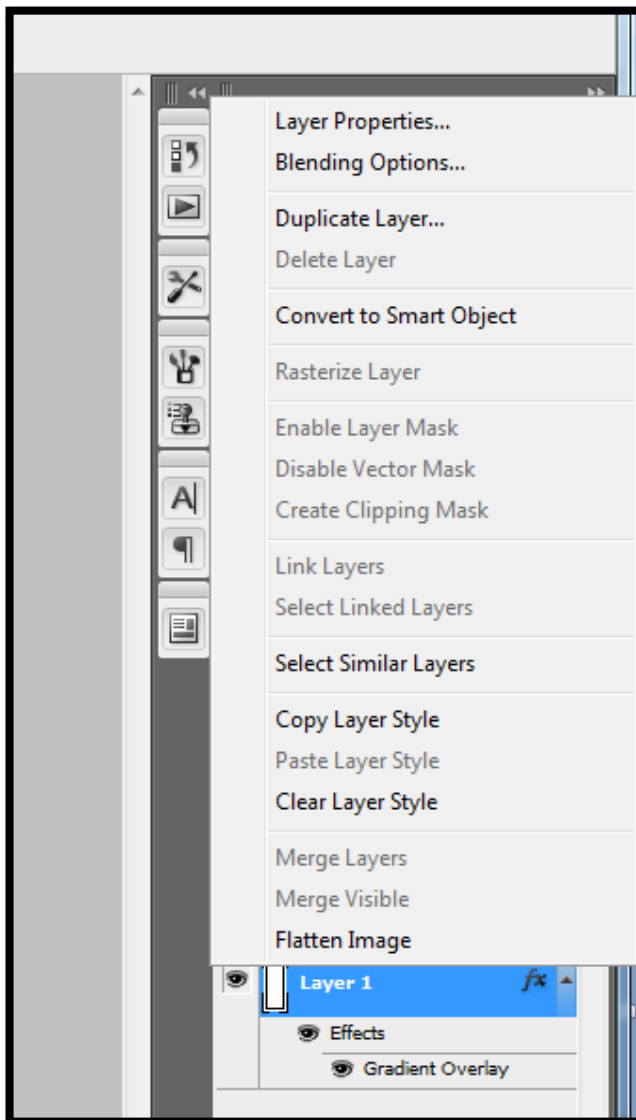
Perhatikan pada class="left" ada property float : left; dan pada class="right" ada property float : right. Properti ini digunakan untuk menggeser elemen sesuai dengan yang kita inginkan ke sebelah kiri/kanan. Apabila dibawah parent ada dua elemen dengan float berlawanan maka elemen tersebut akan membentuk kolom sejajar seperti table dengan 2 kolom asalkan presisi width nya tepat. Kalau presisi pengukuran width salah maka elemen di sebelah kanan akan jatuh ke bawah sehingga hanya terbentuk 2 kolom yang berantakan. Hal yang paling rumit adalah setelah selesai mengatur presisi di browser anda harus melakukan tes di beberapa browser yang lain, karena setiap jenis browser memiliki kompatibilitas yang berbeda-beda, bisa saja di firefox baik, tapi begitu dilihat di internet explorer ternyata berantakan.

Selain itu ada properti clear:both yang digunakan untuk membersihkan pengaruh float dari tag sebelumnya sehingga tag yang dibuat dibawah float tidak terpengaruh geser ke kanan atau ke kiri.

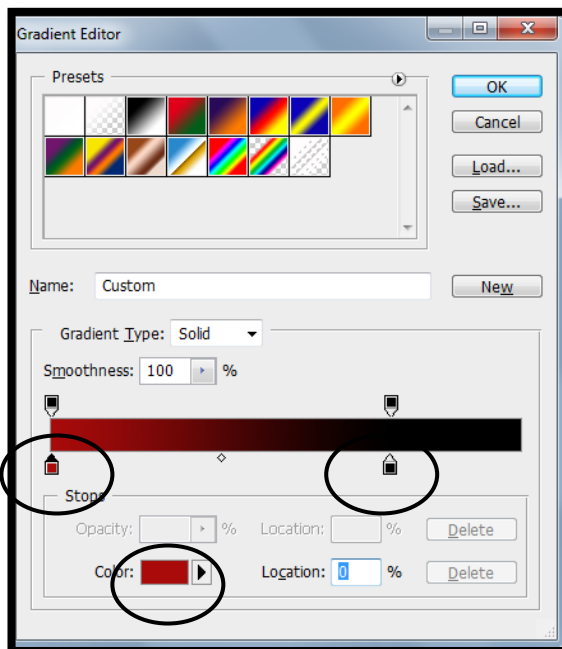
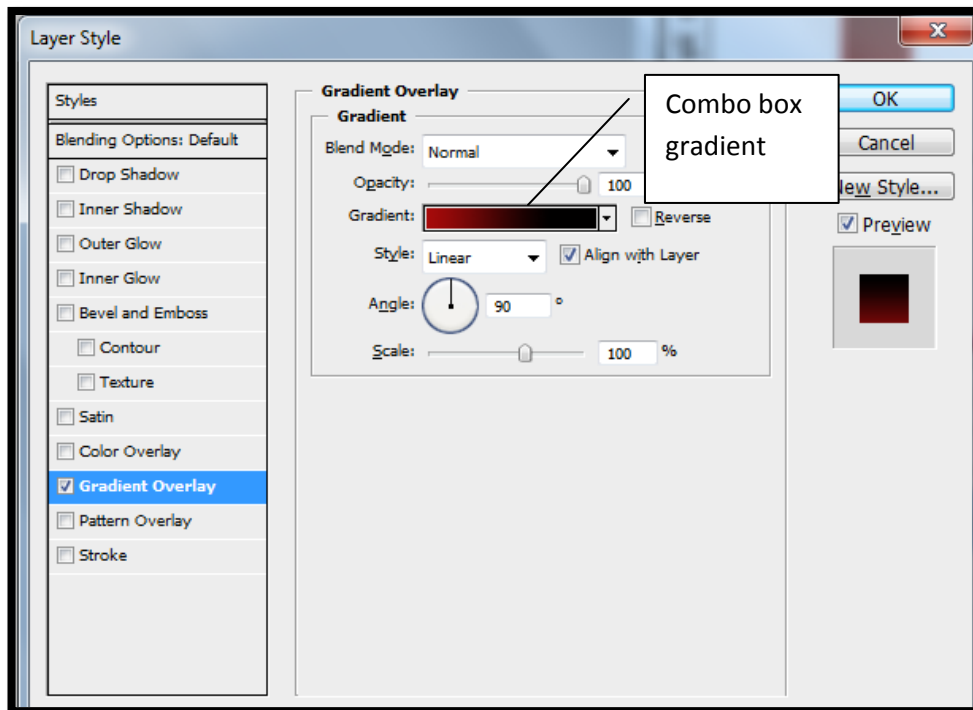
21. Selanjutnya adalah membuat navigasi menu, menggunakan tag . Sebelumnya silahkan buat background dengan ukuran 10 x 40 piksel bergradasi hitam ke merah tua sehingga tampak seperti dibawah ini:



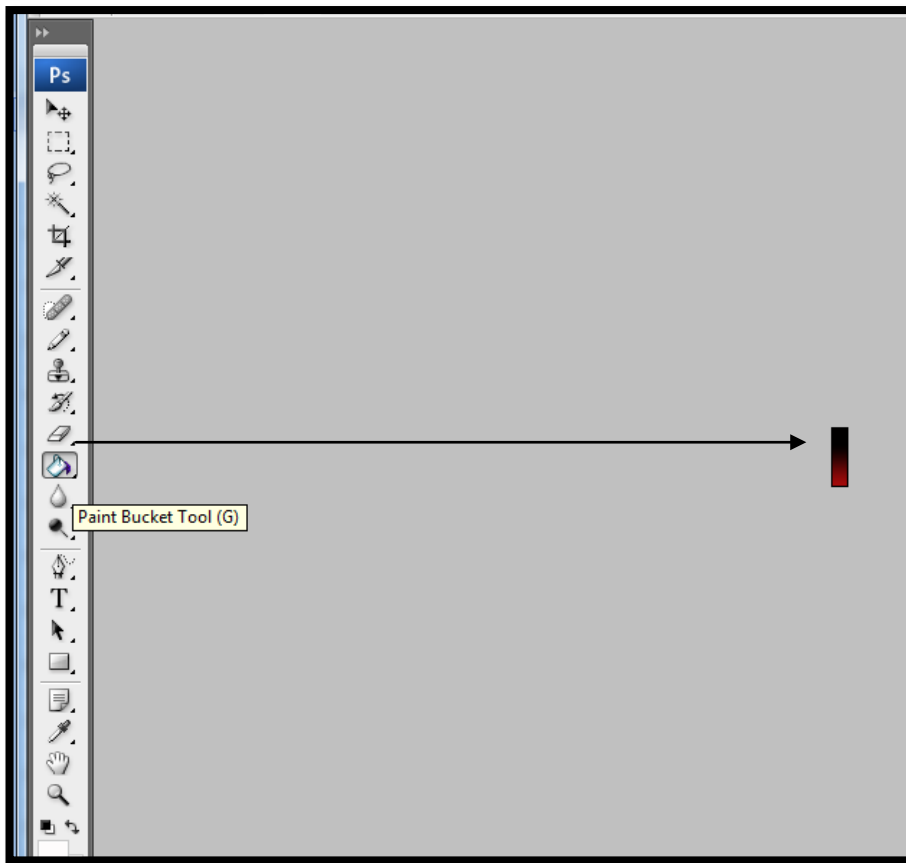
Caranya buat file photoshop baru dengan ukuran kanvas 10 x 40 piksel berwarna latar putih. Kemudian munculkan klik kanan pada layer pilih Blending Option.



Pilih dan centang **Gradient Overlay**. Lalu klik pada **combo box Gradient** untuk menampilkan **Gradient editor**. Tentukan warna gradasi hitam dan merah dengan klik pada **slider gradient** lalu klik pilihan warna dibawahnya.



Gunakan **Paint Bucket Tool** untuk mengecat bidang kanvas.



22. Tambahkan gaya markup CSS sebagai berikut pada bagian CSS tepat di bawah tag .clear

```
ul.navigation{
    list-style:none;
    margin:0 auto;
    padding:0;
}
ul.navigation li{
    display: block;
    float: left;
    text-align: center;
    width: 125px;
    height:40px;
    color: #FFFFFFF;
    font-size: 18px;
    font-weight:bold;
    line-height:30px;
    text-decoration:none;
    border-left:3px solid #000000;
}
ul.navigation li:hover, li.active{
background:url(images/bg_navigation_hover.jpg) repeat-x top;
}
ul.navigation li a{
    display: block;
    width: 125px;
    height:40px;
```

```

        color: #FFFFFF;
        font-size: 18px;
        font-weight: bold;
        line-height: 30px;
        text-decoration: none;
    }
    ul.navigation li a: hover, ul.navigation li a.active{
        color: #6fca0e;
    }
}

```

23. Tambahkan kode html untuk membuat menu yang diletakkan sesudah tag <div> kelas "right" seperti dibawah ini:

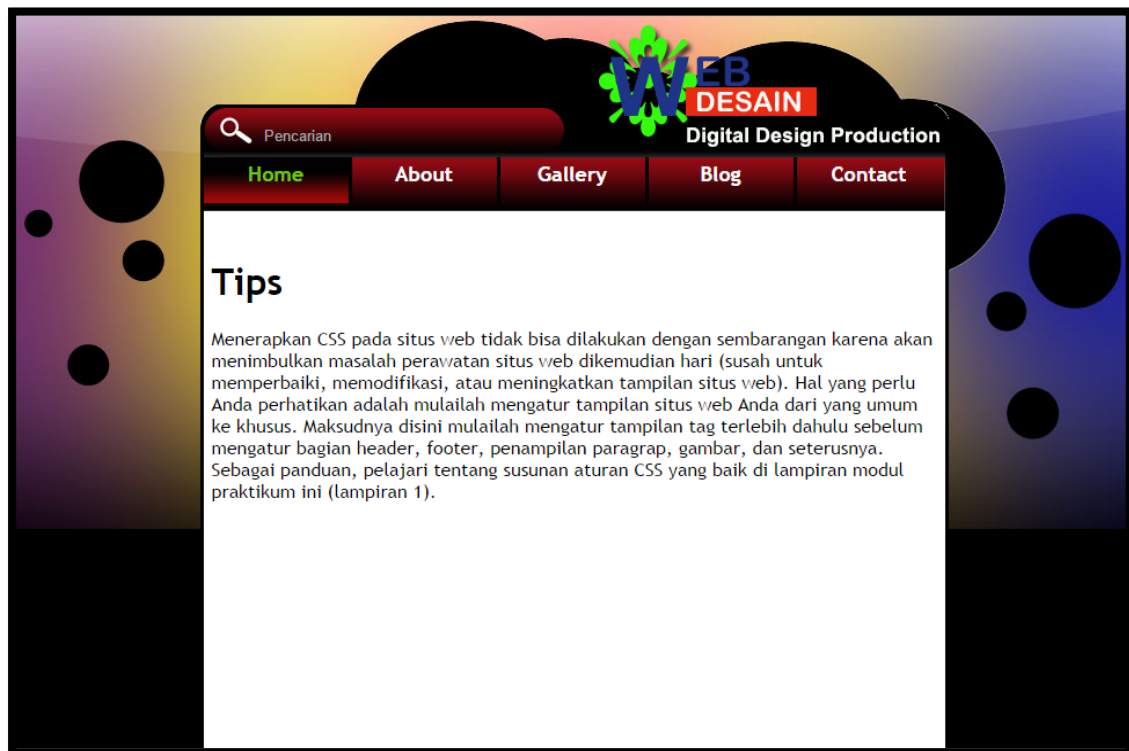
```

<div class="right">
<a href="#"></a>
</div>

<ul class="navigation">
    <li><a href="#">Home</a></li>
    <li><a href="#">About</a></li>
    <li><a href="#">Gallery</a></li>
    <li><a href="#">Blog</a></li>
    <li><a href="#">Contact</a></li>
</ul>

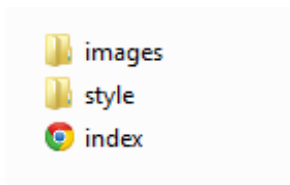
```

24. Setelah langkah terakhir Anda lakukan maka tampilan layout situs Anda seperti ini:



Jika tampilan dari layout situs Anda masih belum rapi alias berantakan, coba atur lagi nilai-nilai padding untuk kelas “.header .left” dan “.header .right”. Tambahkan sementara properti “border:1px solid” pada definisi kelas tersebut untuk memunculkan garis batasnya sehingga Anda akan tahu area kelas-kelas tersebut. Dengan demikian Anda akan jauh lebih mudah untuk mengetahui posisi yang kurang pas dan mana yang harus ukurannya

25. Langkah-langkah berikutnya adalah langkah-langkah terakhir yaitu membuat layout untuk isi halaman utama situs. Sebelumnya penerapan css yang dilakukan dengan cara inline, sekarang kita akan menggunakan cara eksternal, yaitu memisahkan kode-kode css pada file terpisah dari dokumen HTML. Buatlah file css dengan Dreamweaver dan simpan dalam direktori bernama “style” yang akan Anda buat. Dengan demikian susunan direktori situs:



Oleh karena kode-kode css berpindah, maka arah penunjukkan gambar background pada kode-kode css perlu diatur ulang. Misalnya:

```
background:#000000 url(images/background.jpg) no-repeat top;
```

diubah menjadi

```
background:#000000 url(../images/background.jpg) no-repeat top;
```

Layout isi halaman utama situs dibagi menjadi dua kolom. Kolom kiri untuk membuat kategori, menyimpan widget, dan sebagainya. Kolom kanan untuk menampilkan isi dari halaman itu sendiri. Lebar konten adalah 646 piksel, sudah dikurangi untuk border dan padding. Jadi kurang lebih tinggal 620px. Untuk kolom kiri dan kanan lebarnya memperhitungkan ukuran tersebut. Jadi untuk kolom kiri lebarnya 190px dan kolom kanan 430px, total 620px.

Sekarang buang heading dan paragraf yang sudah dibuat pada langkah sebelumnya dalam tag div kelas “container” lalu tambahkan markup berikut:

```
<div class="container">
  <div class="column">
    <div class="left-column"></div>
    <div class="right-column"></div>
  <div class="clear"></div>
```

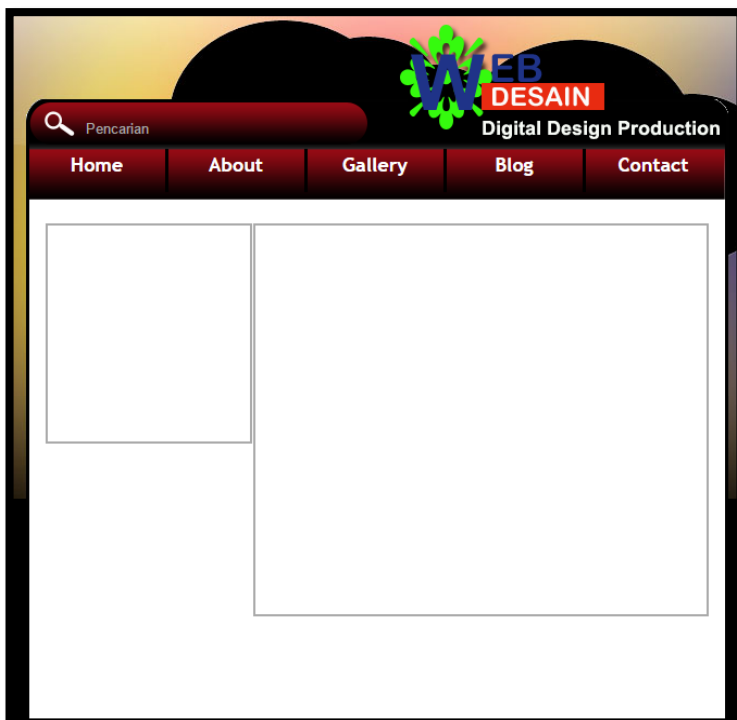
```
</div>  
</div>
```

Kelas “left-column” untuk kolom kiri dan “right-column” untuk kolom kanan.

26. Tambahkan style untuk membentuk 2 kolom tersebut:

```
.container .column{ }  
.container .column .left-column{  
    float:left;  
    width:186px;  
    border-top:2px solid #aaa;  
    border-bottom:2px solid #aaa;  
    border-left:2px solid #aaa;  
    border-right:2px solid #aaa;  
    margin-left:8px;  
    height:200px;  
}  
.container .column .right-column{  
    float:right;  
    width:415px;  
    border:2px solid #aaa;  
    margin-right:8px;  
    height:360px;  
}
```

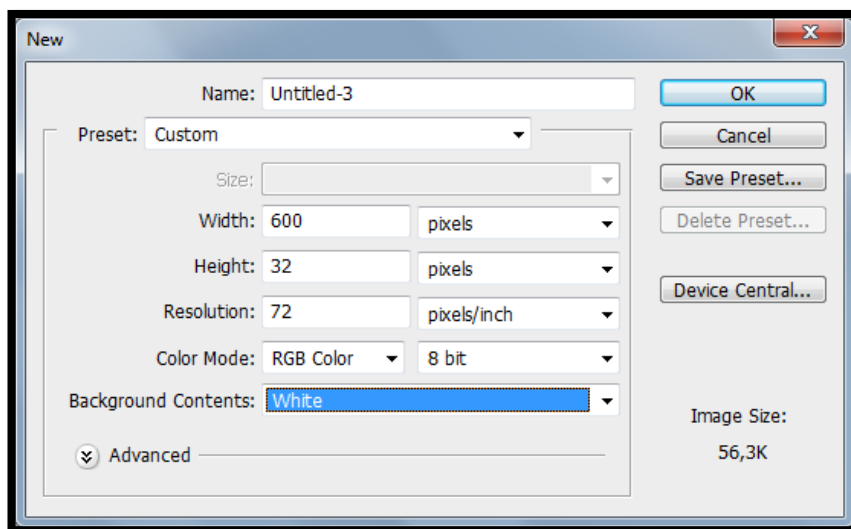
Hasilnya seperti gambar ini:

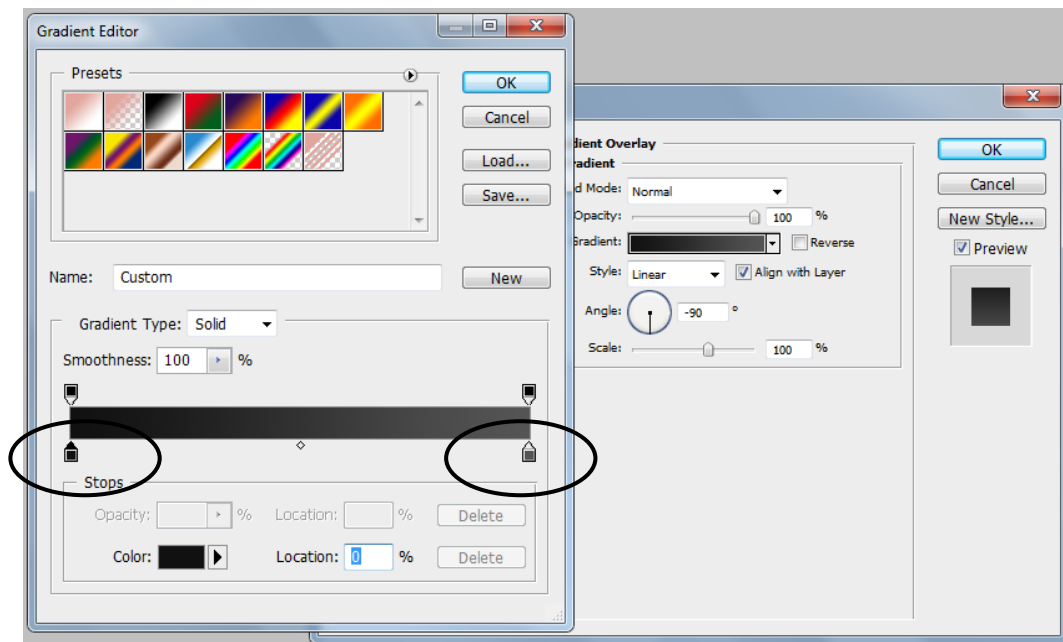


Pada CSS3 dan HTML5 salah satu tambahan propertinya adalah **-moz-border-radius** dan **-webkit-border-radius** yang berguna untuk membuat lengkungan di sudut kotak. Tambahkan kode CSS3 sehingga kode CSS sebelumnya menjadi seperti ini:

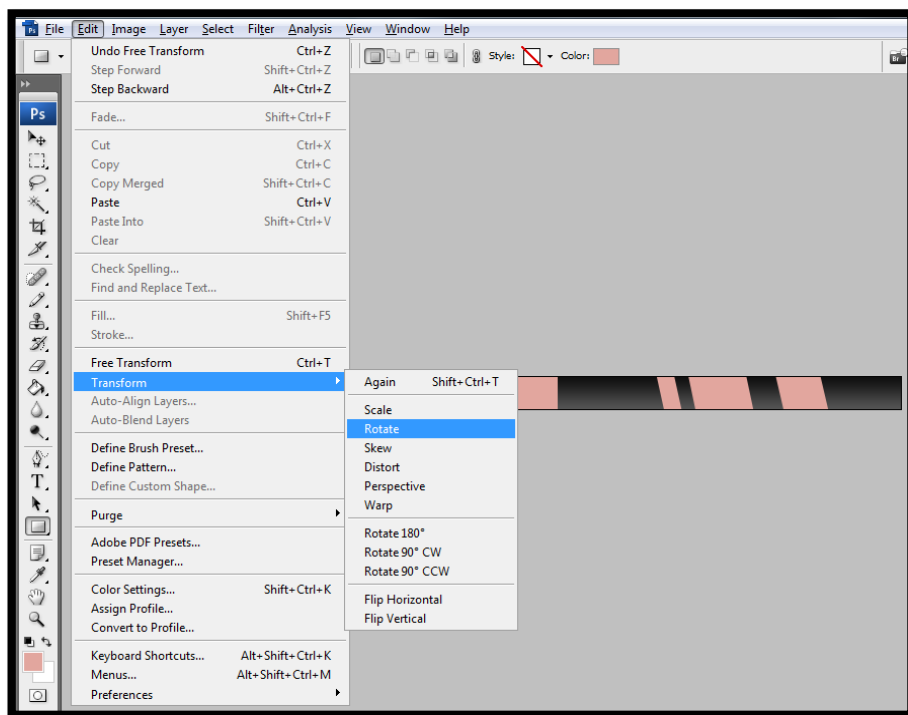
```
.container .column{ }
.container .column .left-column{
    float:left;
    width:186px;
    border-top:2px solid #aaa;
    border-bottom:2px solid #aaa;
    border-left:2px solid #aaa;
    border-right:2px solid #aaa;
    margin-left:8px;
    height:200px;
-moz-border-radius-topleft:10px;
-moz-border-radius-bottomright:10px;
-webkit-border-top-left-radius:10px;
-webkit-border-bottom-right-radius:10px;
}
.container .column .right-column{
    float:right;
    width:415px;
    border:2px solid #aaa;
    margin-right:8px;
    height:360px;
-moz-border-radius:10px;
-webkit-border-radius:10px;
}
```

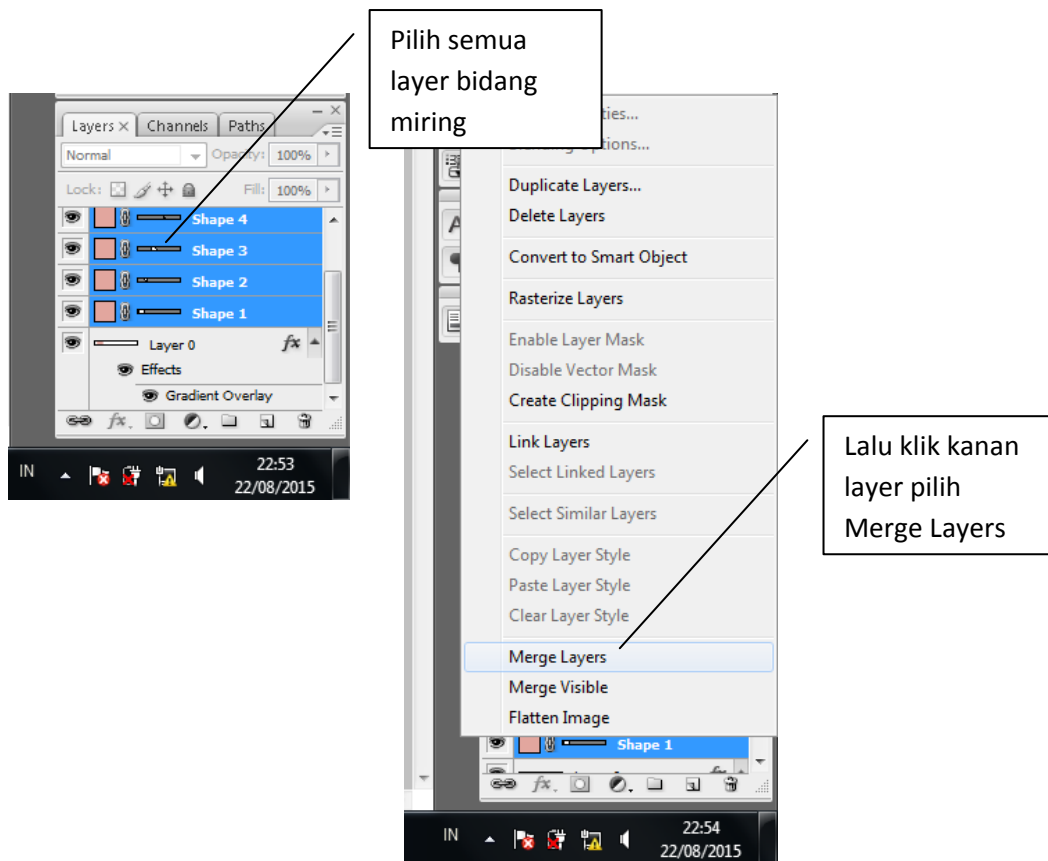
27. Buat background untuk judul kolom kanan dan kiri. Buat file photoshop berukuran 600pixel x 32pixel, beri blending option beri gradasi hitam (#111111) ke abu2 (#555555).





Buatlah bidang-bidang berbagai ukuran berwarna putih dengan **Rectangle Tool** lalu miringkan. Merge semua layer bidang miring yang dibuat lalu beri opacity 5% pada bidang yang telah di merge.





Hasilnya:



Buat satu lagi, kali ini dengan gradasi abu-abu (#aaaaaa) ke putih (#ffffff). Simpan yang pertama dengan **bar-right.png** dan yang ini dengan **bar-left.png**.



Tambahkan CSS background pada **.container .column .left-column** dan **.container .column .right-column**.

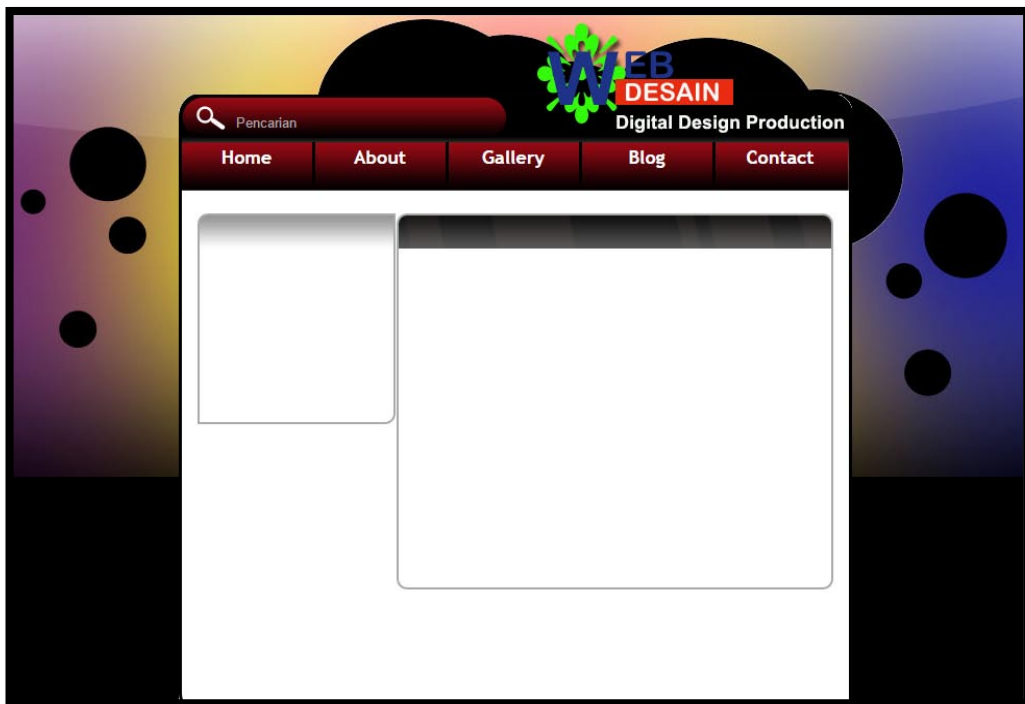
```
.container .column .left-column{
    float:left;
    width:186px;
    border:1px solid #555;
    margin-left:8px;
    height:200px;
    -moz-border-radius-topleft:10px;
    -moz-border-radius-bottomright:10px;
    -webkit-border-top-left-radius:10px;
    -webkit-border-bottom-right-radius:10px;
    background:url(../images/bar-left.png) top repeat-x;
}
.container .column .right-column{
```

```

float:right;
width:415px;
border:1px solid #111;
margin-right:8px;
height:360px;
-moz-border-radius:10px;
-webkit-border-radius:5px;
background:url(../images/bar-right.png) top no-repeat;
}

```

Hasilnya:



28. Setelah langkah-langkah sebelumnya, maka sebagai langkah terakhir adalah memformat semua elemen HTML yang akan tampil pada kolom kiri dan kanan. Untuk itu pertama hapus properti height untuk class left-column dan right-column. Kemudian tambahkan kode-kode CSS berikut ini:

```

.container .column .right-column .no-warp{
    width:410px;
    margin:0 auto;
    padding:5px 0 5px 0;
    background:url(../images/bg_warp.jpg) top repeat-x;
}
.container .column .left-column h1{
    width:180px;
    margin:0 auto;
    margin-bottom:10px;
    border-bottom:1px dotted #aaa;
    font-size:14px;
}

```

```

        font-weight:bold;
        color:#000;
        line-height:26px;
        height:26px;
        text-indent:10px;
    }
    .container .column .left-column ul{
        margin-bottom:20px;
        margin-left:20px;
        list-style:circle;
    }
    .container .column .left-column p{
        padding:0 5px;
        margin-bottom:10px;
    }
    .container .column .left-column ul li{
        margin-bottom:5px;
    }
    .container .column .right-column h1{
        font-size:14px;
        font-weight:bold;
        color:#FFF;
        line-height:32px;
        height:32px;
        text-align:center;
    }
    .container .column .right-column h2{
        font-size:14px;
        font-weight:bold;
        padding:2px 5px;
        margin-bottom:5px;
    }
    .container .column .right-column p{
        margin-bottom:10px;
        padding:0 5px;
    }
    .container .column .right-column p img{
        width:180px;
        float:left;
    }
    .container .column .right-column a{
        color:#F30;
        text-decoration:none;
    }
    .container .column .right-column a:hover{
        text-decoration:underline;
    }
}

```

Semua elemen dan class (p, h, dsb) yang ditulis di depan **.container .column .left-column** adalah bentuk format yang akan diterapkan pada semua elemen HTML yang ada di kolom kiri begitu juga semua elemen dan class yang ditulis di depan **.container .column .right-**

column adalah bentuk format yang akan diterapkan pada semua elemen HTML yang ada di kolom kanan.

Pada class no-warp ditambahkan background untuk memisahkan isi content yaitu **background:url(..images/bg_warp.jpg) top repeat-x;** . Gambar dibuat berukuran 40pixel x 100pixel dengan gradasi abu-abu ke putih.

29. Bagian bawah website biasanya berisi copy right dan link navigasi. Tambahkan kode css berikut:

```
.copy{
    width: 646px;
    height: 50px;
    margin: 0 auto;
    color:#FFF;
    text-align:center;
    font-size:14px;
}
```

Tambahkan kode HTML pada bagian footer sehingga menjadi seperti berikut:

```
<div class="footer">
    <a href="#">Home</a>
    <a href="#">About</a>
    <a href="#">Galerry</a>
    <a href="#">Blog</a>
    <a href="#">Contact</a>
</div>
<div class="copy">CopyRight &copy; by DigitalProduction</div>
```

30. Isi konten dengan menambahkan kode HTML berikut pada div class "content" sehingga kode HTML pada class "content" menjadi seperti ini:

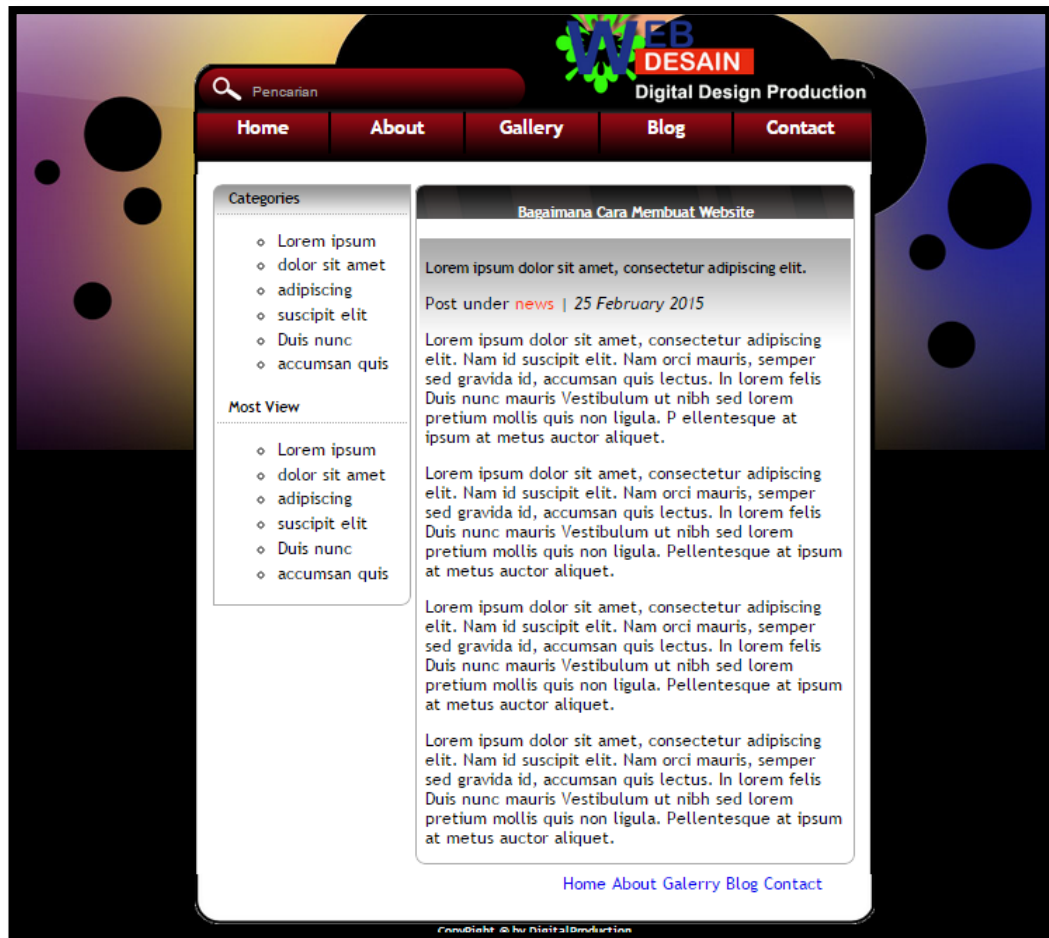
```
<div class="content">
    <div class="container">
        <div class="column">
            <div class="left-column">
                <h1>Categories</h1>
                <ul>
                    <li>Lorem ipsum</li>
                    <li>dolor sit amet</li>
                    <li>adipiscing</li>
                    <li>suscipit elit</li>
                    <li>Duis nunc</li>
                    <li>accumsan quis</li>
                </ul>
                <h1>Most View</h1>
            </div>
        </div>
    </div>
```

```

        <li>Lorem ipsum</li>
        <li>dolor sit amet</li>
        <li>adipiscing</li>
        <li>suscipit elit</li>
        <li>Duis nunc</li>
        <li>accumsan quis</li>
    </ul>
</div>
<div class="right-column">
    <h1>Bagaimana Cara Membuat Website</h1>
    <div class="no-warp">
        <h2>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</h2>
        <p>Post under <a href="#">news</a> | <em>25 February
2015</em></p>
        <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit.
        Nam id suscipit elit. Nam orci mauris, semper sed gravida id,
        accumsan quis lectus. In lorem felis Duis nunc mauris
        Vestibulum ut nibh sed lorem pretium mollis quis non ligula. P
        ellentesque at ipsum at metus auctor aliquet.
        </p>
        <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit.
        Nam id suscipit elit. Nam orci mauris, semper sed gravida id,
        accumsan quis lectus. In lorem felis Duis nunc mauris
        Vestibulum ut nibh sed lorem pretium mollis quis non ligula.
        Pellentesque at ipsum at metus auctor aliquet.
        </p>
        <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit.
        Nam id suscipit elit. Nam orci mauris, semper sed gravida id,
        accumsan quis lectus. In lorem felis Duis nunc mauris
        Vestibulum ut nibh sed lorem pretium mollis quis non ligula.
        Pellentesque at ipsum at metus auctor aliquet.
        </p>
        <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit.
        Nam id suscipit elit. Nam orci mauris, semper sed gravida id,
        accumsan quis lectus. In lorem felis Duis nunc mauris
        Vestibulum ut nibh sed lorem pretium mollis quis non ligula.
        Pellentesque at ipsum at metus auctor aliquet.
        </p>
    </div>
</div>
<div class="clear"></div>
</div>
</div>
</div>

```

Tampilkan pada browser, maka hasil akhirnya seperti ini:



Tugas

Tugas praktikum sebelumnya adalah Anda diminta untuk membuat desain layout Anda sendiri berdasarkan langkah-langkah praktikum yang diberikan. Tugas praktikum kali ini adalah melanjutkan tugas praktikum sebelumnya dengan menambahkan apa yang sudah dijelaskan dalam praktikum kali ini sehingga layout situs Anda sudah dilengkapi dengan header, kotak pencarian, dan menu navigasi.

Praktikum 7

Bahasa Javascript

Tujuan Pembelajaran

Mempraktekan penggunaan Javascript untuk membuat validasi form sederhana

Pokok Bahasan

- Validasi form
- Fungsi setIndexof()

Indikator Keberhasilan

Mahasiswa dapat membuat validasi dengan Javascript

Penjelasan

Bahasa script Javascript paling populer di Internet dan dapat bekerja pada semua browser utama seperti Internet Explorer, Firefox, Google Chrome, Safari, dan Opera.

Untuk mempelajari Javascript dibutuhkan penguasaan terhadap bahasa utama web yaitu HTML atau XHTML. Beberapa hal penting yang perlu diketahui tentang javascript:

- Javascript dirancang untuk memberikan unsur interaktif kepada halaman HTML
- Javascript merupakan bahasa script. Bahasa script adalah bahasa pemrograman yang ringan, dalam arti aturan pemrogramannya tidak serumit bahasa pemrograman murni
- Javascript biasanya digunakan dengan cara menuliskannya secara langsung pada halaman HTML
- Javascript merupakan bahasa interpreter, yang berarti script Javascript tidak perlu dikompilasi sebelum dieksekusi.
- Javascript bersifat gratis

Javascript dapat digunakan untuk melakukan hal-hal berikut:

- membuat teks dinamis dalam halaman HTML. Javascript dapat menulis teks atau isi suatu variabel pada halaman HTML.
- menangani event. Kejadian-kejadian seperti tindakan mengklik mouse, atau ketika halaman HTML telah selesai dimuat ke browser dapat dikenali oleh javascript dan dapat mengatur tindakan tertentu ketika event itu terjadi.
- membaca dan menulis elemen-elemen HTML, yang berarti dapat digunakan untuk mengubah konten suatu elemen HTML.
- memvalidasi data, misalnya data form sebelum dikirim ke server sehingga dapat meringankan server.
- mendeteksi browser pengunjung

membuat cookies yang akan menyimpan dan memanggil informasi kedalam komputer pengunjung.

Javascript digunakan dengan cara menuliskan langsung pada dokumen HTML seperti contoh berikut:

```
<html>
<body>
<script type="text/javascript">
document.write("Hello world!!");
</script>
</body>
</html>
```

Selain itu javascript juga dapat ditulis terpisah dalam suatu file seperti CSS dengan akhiran **.js**. Untuk itu dokumen HTML perlu memanggil file javascript dengan menambahkan kode berikut pada bagian head dokumen HTML:

```
<head>
<script type="text/javascript" src="nama_file_javascript.js"></script>
</head>
```

Praktikum

Dalam praktikum ini Anda akan membuat sebuah halaman Kontak. Biasanya halaman kontak ini berisi form yang digunakan untuk mengirimkan pesan kepada pemilik situs berikut data diri sipengirim pesan. Form kontak yang akan Anda buat terdiri dari isian:

- Nama
- Email
- Blog/ Situs
- Pesan

Masing-masing isian dilengkapi dengan validasi input sehingga data yang masuk adalah data yang diinginkan. Validasi akan dibuat menggunakan bahasa Javascript. Form masih belum dilengkapi dengan file pengolah data (dengan PHP), jadi data isian form belum dapat tersimpan.

Ikuti instruksi praktikum berikut ini:

1. Buatlah form kontak berisi isian nama, email, blog/situs, dan pesan. Aturlah gaya tampilan form dengan css eksternal. Kode-kode CSS yang dibuat disimpan kedalam file CSS yang telah dibuat pada praktikum sebelumnya. Beri nama dengan **kontak.html** dan simpan pada direktori yang sama dengan file index.html yang telah dibuat sebelumnya.
2. Buat file javascript untuk menyimpan kode-kode validasi. Beri nama dengan **validasi.js** dan simpan pada direktori baru dengan nama **script**. Validasi yang dibuat:
 - Memastikan isian nama dan email tidak boleh kosong
 - Memastikan alamat email ditulis dengan benar yaitu mengandung simbol "@" setelah nama pengguna dan mengandung tanda titik (".") seperti "gmail.com"

Untuk validasi pertama periksa atribut **value** dari elemen input nama atau email memiliki isi atau tidak kosong. Sedangkan validasi kedua menggunakan fungsi **indexOf** (huruf "O" kapital) yang digunakan untuk memeriksa keberadaan karakter "@" dan "." dalam alamat email. Buat validasi javascript tersebut dalam bentuk fungsi. Kode javascript yang diperlukan dan ditulis pada file *validasi.js* sebagai berikut:

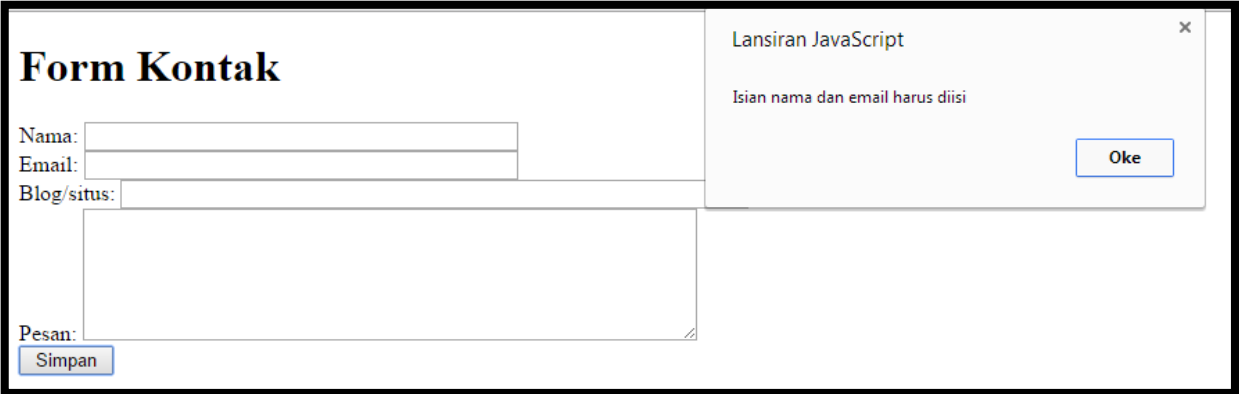
```
function validasikontak(dataform){
    if (dataform.nama.value==""){
        alert ("Data nama belum diisi");
        dataform.nama.focus();
        return (false);
    }
    if (dataform.email.value==""){
        alert ("Data email masih kosong");
        dataform.email.focus();
        return (false);
    }else{
        var vemail=dataform.email.value;
        posisiAt= vemail.indexOf("@");
        posisiTitik=vemail.lastIndexOf(".");
        if (posisiAt < 1 || (posisiTitik - posisiAt < 2)){
            alert("Alamat email belum benar")
            dataform.email.focus();
            return (false);
        }
    }
    return (true);
}
```

Kode validasi javascript diatas menggunakan struktur **if** untuk melakukan proses validasi. Nama fungsi *validasikontak* dengan satu parameter *dataform* berisi obyek form itu sendiri. Fungsi *indexOf()* dan *lastIndexOf()* akan mengembalikan nilai *false* jika isian salah dan *true* jika benar. Fungsi *validasikontak* ini akan mengembalikan nilai *false* jika isian masih salah dan *true* jika isian benar.

3. Panggil file javascript dari dokumen HTML dengan menuliskan kode berikut dalam tag<head> dokumen HTML:

```
<head>
<script type="text/javascript" src="js/validasi.js"></script>
</head>
```

Jika dokumen HTML dijalankan pada browser, maka ketika isian nama dan email kosong atau alamat email belum benar, browser akan menampilkan kotak peringatan seperti ini:



The image shows a web form titled "Form Kontak" with input fields for "Nama:", "Email:", "Blog/situs:", and "Pesan:". A "Simpan" button is at the bottom left. A JavaScript alert box titled "Lansiran JavaScript" is overlaid on the right, displaying the message "Isian nama dan email harus diisi" and an "Oke" button.

Klik tombol *Oke* untuk menutup kotak peringatan dan form akan disajikan lagi sehingga pengguna dapat memperbaiki isian nama dan email.

Tugas

Siapkan form kontak ini untuk digunakan dalam praktikum selanjutnya. Atur dan tingkatkan penampilan form menggunakan CSS. Tulis kode-kode CSS tersebut dalam file CSS yang sudah dibuat dalam praktikum sebelumnya.

Praktikum 8

DOM (Document Object Model)

Tujuan Pembelajaran

Mengenalkan konsep DOM kepada Mahasiswa agar dapat menggunakannya untuk meningkatkan unsur dinamis dan interaktif halaman web.

Pokok Bahasan

- Konsep DOM (Document Object Model)
- Metode getElementById
- Properti innerHTML

Indikator Keberhasilan

Mahasiswa dapat memahami konsep DOM dan dapat menerapkan konsep ini untuk memanipulasi elemen HTML

Penjelasan

Jika Anda ingin mengganti suatu gambar pada suatu halaman web secara otomatis saat dokumen HTML ditampilkan pada browser sehingga gambar akan berganti-ganti maka Anda memerlukan DOM. DOM atau *Document Object Model* adalah suatu antarmuka yang bebas platform dan bahasa tertentu yang membuat program dan script dapat mengakses dokumen HTML secara dinamis dan mengupdate struktur, konten, dan style dokumen.

Konsep DOM memperlakukan elemen-elemen HTML sebagai obyek. Metode dan properti dari obyek ini yang akan dimanipulasi dengan bahasa tertentu (biasanya Javascript) untuk membuat unsur dinamis dan interaktif pada suatu halaman web.

Untuk memahami konsep DOM berikut contoh yang akan mengubah konten elemen paragraf dengan id “demo” menggunakan properti “innerHTML”:

```
<html>
<head>
  <title>Contoh DOM</title>
</head>
<body>
  <p id="demo"></p>
  <script type="text/javascript">
    document.getElementById("demo").innerHTML = "Hello World!";
  </script>
</body>
</html>
```

Metode **getElementById** adalah metode yang biasa digunakan untuk mengakses elemen HTML berdasarkan atribut “id” dari elemen HTML. Properti **innerHTML** biasa digunakan untuk memperoleh atau mengubah isi suatu elemen HTML

Praktikum

Dalam praktikum kali ini Anda akan diajak untuk membuat suatu kemampuan interaktif yang diterapkan pada suatu gambar yang ditampilkan pada halaman web. Dengan kemampuan ini, sebuah ikon gambar dapat diklik untuk menampilkan gambar dalam ukuran besar sehingga akan tampak lebih jelas detailnya. Kemampuan ini sangat bermanfaat pada situs e-commerce dimana pengguna dapat melihat produk dengan lebih seksama sementara pemilik situs tidak harus menampilkan semua gambar dengan ukuran yang besar yang tentunya akan menghabiskan sumber daya situs.

Halaman yang akan dibuat akan menampilkan beberapa gambar dengan ukuran kecil. Ketika gambar-gambar tersebut diklik maka sebuah kotak melayang akan dibuat yang berisi gambar dengan ukuran besar. Untuk membuat halaman ini, file yang diperlukan adalah file html, css, dan javascript. File html untuk menampilkan gambar dalam halaman, file css untuk mengatur tampilan kotak melayang berisi gambar, sedangkan javascript untuk membuat fungsi-fungsi membuka dan menutup kotak melayang.

Konsep DOM yang digunakan dalam praktikum ini adalah mengambil isi suatu elemen dengan menggunakan metode *getElementById* dan memanipulasi isi suatu elemen HTML dengan properti *innerHTML*.

Untuk membuat praktikum ini Anda akan memerlukan beberapa buah gambar dengan ukuran cukup besar sekitar setengah layar browser. Siapkan beberapa gambar tersebut dalam folder **images**. Setelah itu ikuti instruksi praktikum berikut ini:

1. Buat sebuah dokumen HTML dengan kode sebagai berikut:

```
<!DOCTYPE html>
<html>
<head>
<title>Kotak Melayang untuk Gambar</title>
</head>

<body>
<h1>Demonstrasi Lightbox untuk menampilkan gambar</h1>
<p>Dalam demo ini akan menampilkan gambar dalam sebuah kotak yang dibuat
secara otomatis begitu gambar diklik</p>

<br>

<div id="shadowing"></div>
<div id="box">
  <div id="boxcontent" onclick="closebox()"></div>
  <div id="boxtitle"></div>
</div>

<p>Untuk menampilkan gambar kedalam kotak lightbox, klik thumbnail gambar
dibawah ini...</p>




<br>

<hr>

<p>(c) 2008 <a href="http://www.xul.fr/en/" target="_parent">xul.fr</a></p>
</body>
</html>
```

Dalam dokumen HTML tersebut ada kode berikut ini:

```
<div id="shadowing"></div>
<div id="box">
  <div id="boxcontent" onclick="closebox()"></div>
  <div id="boxtitle"></div>
</div>
```

Kode ini untuk membuat kotak melayang. Div dengan id “shadowing” untuk melapisi halaman web agar tidak dapat diklik ketika kotak melayang muncul. Div id “box” untuk membuat kotak melayang. Div id “boxcontent” untuk menempatkan gambar besar,

sedangkan div dengan id “boxtitle” untuk menampilkan nama gambar yang sedang ditampilkan.

Div “boxcontent” memiliki event handler onClick yang akan mengeksekusi fungsi javascript *closebox* jika area div ini diklik. Fungsi *closebox* untuk menutup kotak melayang.

Masing-masing tag memiliki event handler onClick yang akan menjalankan fungsi *openbox* yang akan membuat kotak melayang dan menampilkan gambar didalamnya.

2. Berikutnya buat file javascript **lightbox.js** berikut ini:

```
function openbox(url)
{
    var box = document.getElementById('box');
    document.getElementById('shadowing').style.display='block';

    var title = document.getElementById('boxtitle');
    title.innerHTML = url;

    var content = document.getElementById('boxcontent');
    content.style.padding="0";

    content.innerHTML = "";

    box.style.display='block';
}

function closebox()
{
    document.getElementById('box').style.display='none';
    document.getElementById('shadowing').style.display='none';
}
```

Variabel *box* berisi elemen div dengan id “box”, variabel *title* berisi elemen div “boxtitle” yang diisi dengan alamat dan nama dari gambar yang sedang ditampilkan. Variabel *content* berisi elemen div dengan id “boxcontent”. Div “boxcontent” akan diisi dengan gambar melalui kode *content.innerHTML=""*;

3. Berikutnya buat file CSS **lightbox.css** ini:

```
#shadowing
{
    display: none;
    position: absolute;
    top: 0%;
    left: 0%;
    width: 100%;
    height: 100%;
    background-color: #CCA;
    z-index:10;
    opacity:0.5;
```

```

        filter: alpha(opacity=50);
    }

    #box
    {
        display: none;
        position: absolute;
        top: 20%;
        left: 20%;
        width: 50%;
        height: 65%;
        padding: 0;
        margin: 0;
        background-color: white;
        z-index: 11;
    }

    #boxcontent
    {
        left: 0;
        right: 0;
        bottom: 0;
        margin: 0;
        padding: 8px;
        overflow: auto;
        width: 100%;
        height: 100%;
        text-align: center;
    }

    #boxtitle
    {
        left: 0%;
        width: 100%;
        height: 20px;
        padding: 0;
        margin: 0;
        color: black;
        text-align: center;
    }
}

```

Id *shadowing* mengatur gaya tampilan kotak yang melapisi halaman web agar tidak dapat diklik ketika kotak melayang sedang tampil. Id *box* mengatur gaya tampilan kotak melayang, id *boxcontent* mengatur area untuk menampilkan gambar, sedangkan id *boxtitle* mengatur gaya tampilan untuk menampilkan alamat dan nama gambar yang sedang tampil.

4. Setelah ketiga file tersebut selesai dibuat, silahkan jalankan file HTML lalu klik salah satu gambar yang ditampilkan. Apabila semua kode sudah benar, maka sebuah kotak melayang

akan muncul ditengah-tengah layar browser yang berisi gambar tersebut dalam ukuran besar.

5. Untuk lebih memahami kode-kode diatas, cobalah ubah-ubah nilai-nilai CSS dan eksperimen dengan file javascript-nya.

Tugas

Buatlah sebuah halaman yang akan menampilkan berbagai photo yang berhubungan dengan diri Anda. Buatlah agar gambar tersebut dapat diklik yang akan menampilkan gambar tersebut dengan ukuran besar. Simpanlah dengan nama gallery.html, dan gallery.js. Simpan file javascript-nya dalam direktori **js**. Sedangkan kode-kode CSS-nya tambahkan kedalam file css yang telah dibuat pada praktikum sebelumnya.

Praktikum 9

Bahasa PHP dan MySQL

Tujuan Pembelajaran

Mempraktekkan pembuatan program-program dasar pengolahan data dengan menggunakan bahasa PHP

Pokok Bahasan

- Program Menambah Data
- Program Mencetak Data
- Program Login
- Program-program Administrator

Indikator Keberhasilan

Mahasiswa memahami langkah-langkah pengolahan data menggunakan PHP dan dapat menerapkannya untuk membuat program input dan cetak data

Penjelasan

Php merupakan bahasa script yang dijalankan pada sisi server (SSS : Server Side Scripting). Database yang didukung PHP antara lain : MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC. PHP adalah software Open Source, bebas untuk diunduh dan digunakan.

File PHP dapat berisi teks, tag HTML, dan Script. File PHP dikembalikan ke browser dalam bentuk plain HTML. File PHP dapat berekstensi .php, .php3, atau .phtml.

Kelebihan PHP sebagai bahasa script adalah dapat dijalankan pada berbagai platform (windows, linux, dan sebagainya), kompatibel terhadap hampir semua server yang digunakan saat ini, bebas

diunduh dari situs resmi PHP www.php.net, mudah dipelajari dan berjalan dengan efisien pada sisi server.

Untuk menggunakan PHP harus menginstall terlebih dahulu web server Apache (atau IIS) pada komputer/server yang akan digunakan, serta menginstall PHP dan MySQL. Atau dapat menggunakan layanan hosting yang menyediakan dukungan terhadap PHP dan MySQL.

Proses mengolah data menggunakan PHP pada dasarnya menggunakan tiga langkah berikut:

1. Membuat sambungan dari PHP ke server database (MySQL)

Langkah pertama memerlukan fungsi *mysql_connect()* dengan tiga parameternya yaitu: nama host tempat server database mysql berada, nama user yang terdaftar pada server database MySQL, dan password dari nama user. Sintaksnya:

```
mysql_connect("nama_host","nama_username", "password_username");
```

Contoh:

```
mysql_connect("192.168.10.250","a121204567","");
```

2. Menentukan dan membuka database yang akan digunakan

Langkah kedua menggunakan fungsi *mysql_select_db()* dengan parameter tunggal yaitu nama database yang akan digunakan. Sintaks:

```
mysql_select_db("nama_db");
```

Contoh:

```
mysql_select_db("a121204567");
```

3. Menjalankan query

Langkah ketiga menggunakan fungsi *mysql_query()* dengan parameter berupa perintah sql yang akan dijalankan. Sintaksnya:

```
mysql_query("perintah_sql");
```

Contoh:

```
mysql_query("insert mahasiswa set nim='$nim', nama='$nama', ipk='$ipk'");
```

Jika untuk membuat program cetak, maka ditambah satu langkah lagi yaitu menguraikan data yang akan dicetak dengan menggunakan fungsi *mysql_fetch_row()* dengan parameter tunggal yaitu hasil query.

Contoh:

```

$sql="select * from mahasiswa";
$qry=mysql_query($sql)
While($hasil=mysql_fetch_row($qry)){
    echo "Nim :".$hasil[0]."<br />
        Nama: ".$hasil[1]."<br />
        IPK:".$hasil[2]."<br />";
}

```

Praktikum

Pada praktikum kali ini Anda akan dituntun untuk membuat program-program dasar menggunakan PHP. Salah satu program dasar tersebut adalah program input data. Untuk membuat program input data ini, silahkan gunakan form yang telah dibuat pada praktikum sebelumnya (kontak.html) yang berisi inputan data kontak untuk pengguna, yaitu NAMA, JENIS KELAMIN, EMAIL, ALAMAT, KOTA, PESAN. Form tersebut belum dilengkapi dengan file pengolah datanya. Oleh karena itu pada praktikum ini, form tersebut akan dilengkapi dengan program pengolah data. Selanjutnya ikuti instruksi-instruksi berikut ini:

1. Buat tabel dengan nama kontak dalam database Anda (database Anda biasanya sudah dibuatkan dengan nama sesuai username, a12xxxxxx. Jika belum, buat database dengan nama username Anda). Field-fieldnya adalah:

Field	Tipe Data	Panjang	Keterangan
id	integer	4	Not null, auto_increment
Nama	Varchar	30	Not null
jkel	Varchar	10	-
Email	Varchar	40	-
Alamat	Varchar	50	-
Kota	Varchar	20	-
Pesan	text	-	-

2. Tambahkah pada atribut action form nama file pengolah datanya yaitu **simpan_kontak.php**. File ini yang akan dibuat pada praktikum kali ini.
3. Buat file **simpan_kontak.php** dan simpan pada direktori yang sama dengan direktori dimana file kontak.html tersimpan.
4. Tulis kode berikut pada file **simpan_kontak.php**:

```

<?php
$host="nama_host";
$username="nama_username";

```

```

$password="password_username";
mysql_connect($host, $username, $password) or die("Koneksi gagal dibangun");
mysql_select_db("situspersonal") or die("Database tidak dapat dibuka");
//memindahkan nilai data form ke variabel sederhana agar mudah ditulis
$vnama=$_POST['nama'];
$vjkel=$_POST['jkel'];
$vemail=$_POST['email'];
$valamat=$_POST['alamat'];
$vkota=$_POST['kota'];
$vpesan=$_POST['pesan'];

$sql="insert kontak set nama='$vnama',
                                jkel='$vjkel',
                                email='$vemail',
                                alamat='$valamat',
                                kota='$vkota',
                                pesan='$vpesan'";
mysql_query($sql) or die("Proses simpan ke database gagal");
mysql_close();
header("location:kontak.html");
?>

```

Sebelumnya, ganti nilai *nama_host* dengan *dinus.net* atau *192.168.10.250*, *nama_username* dengan nama user Anda (*a12xxxxxxx*), dan *password_username* untuk username Anda biasanya dikosongkan. Pada program diatas, perintah sql disimpan dulu dalam sebuah variabel, kemudian variabel ini yang digunakan pada fungsi `mysql_query()`. Perintah terakhir, yaitu `header("location:kontak.html")`, digunakan untuk mengarahkan browser agar menampilkan form setelah proses penyimpanan ke database selesai

5. Jalankan form kontak lalu isi dengan data kemudian tekan tombol submitnya. Jika program sudah benar maka sebuah rekod akan ditambahkan kedalam database.
6. Setelah program input selesai, berikutnya adalah membuat program cetak untuk menampilkan data yang sudah tersimpan dalam database. Buat sebuah file php baru beri nama **cetak.php**. Simpan file ini dalam direktori baru yang perlu Anda buat terlebih dahulu yaitu **admin**. Program cetak disimpan dalam direktori admin karena program ini tergolong program administrator situs yang hanya boleh dibuka oleh pemilik situs.
7. Sebelum melanjutkan membuat program cetak, buat terlebih dahulu file php dan beri nama **koneksi.inc.php** serta simpan dalam direktori admin, yang berisi kode:

```

$host="isi_dengan_nama_host";
$username="isi_dengan_username";
$password="";
mysql_connect($host, $username, $password) or die("Koneksi gagal dibangun");
mysql_select_db("situspersonal") or die("Database tidak dapat dibuka");

```

Perintah-perintah yang ada didalam file koneksi.inc.php ini adalah perintah yang selalu digunakan setiap kali kita membuat program php. Oleh karena itu, biasanya perintah-perintah ini disimpan dalam file terpisah lalu dipanggil dari file php yang membutuhkannya. Hapus perintah-perintah yang sama dalam file simpan_kontak.php dan ganti dengan perintah:

```
include "admin/koneksi.inc.php";
```

Perintah ini akan memuat isi file koneksi.inc.php kedalam file simpan_kontak.php sehingga dapat digunakan olehnya.

8. Pada file cetak.php tulis kode berikut:

```
<?php
include "koneksi.inc.php";
$sql="select * from kontak order by nama;";
$qry=mysql_query($sql) or die("Proses cetak gagal");
echo "<table width='75%' cellpadding='2' cellspacing='0' border='1'>
    <tr>
        <th>No</th>
        <th>Nama</th>
        <th>Email</th>
        <th>Alamat</th>
        <th>Kota</th>
        <th>Pesan</th>
        <th></th>";
$no=1;
while($hasil=mysql_fetch_row($qry)){
    echo "<tr>
        <td>$no</td>
        <td>$hasil[0]</td>
        <td>$hasil[1]</td>
        <td>$hasil[2]</td>
        <td>$hasil[3]</td>
        <td>$hasil[4]</td>
        <td>$hasil[5]</td>
    </tr>";
    $no++;
}
echo "</table>";
?>
<a href="index.html">Kembali</a>
```

Tugas

Cari di Internet program login. Setelah diperoleh ganti namanya dengan index.php dan simpan dalam direktori admin. Buatlah tabel mysql yang diperlukan oleh program login ini dalam database Anda. Modifikasi kodenya agar Jika program login ini dijalankan kemudian diisi dengan username

dan password yang benar, maka akan menampilkan program cetak.php milik Anda. Jadi program cetak.php milik Anda hanya bisa dibuka melalui program login.

Praktikum 10

Aksesibilitas

Tujuan Pembelajaran

Mahasiswa dapat menggunakan Dreamweaver untuk memeriksa tingkat aksesibilitas suatu halaman web dan dapat menggunakan validator online untuk memeriksa validitas kode-kode HTML.

Pokok Bahasan

Aksesibilitas situs

Indikator Keberhasilan

Mahasiswa dapat melakukan pemeriksaan aksesibilitas situs dan validasi dokumen HTML secara online

Penjelasan

“The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect.”

Tim Berners-Lee, W3C Director and inventor of the World Wide Web

Kekuatan dari web adalah sifat universalitasnya, artinya web seharusnya dapat diakses oleh semua orang dengan mengabaikan segala keterbatasan yang dimiliki. Untuk dapat diakses oleh semua orang web harus mempertimbangkan setiap keterbatasan yang dimiliki oleh pengguna. Keterbatasan disini termasuk cacat fisik pengguna, sehingga tidak dapat menggunakan mouse. Keterbatasan lingkungan dimana pengguna mengakses web, misalnya, dalam ruangan yang bising

sehingga tidak dapat mendengarkan suara audio dari sebuah situs, ruangan yang bercahaya terlalu terang atau gelap sehingga mengurangi kemampuan untuk melihat layar monitor dengan jelas. Keterbatasan lain, berasal dari penurunan kemampuan yang normal dari manusia yang beranjak tua, pengurangan kemampuan melihat dan mendengar, dan sebagainya. Faktor teknologi, seperti komputer yang digunakan tergolong “kuno” sehingga “lelet” dan berlayar sempit. Juga mereka yang berada di daerah pedalaman dengan akses internet yang terbatas sehingga koneksi yang digunakan sangat lambat.

Pedoman Standar Aksesibilitas

Untuk dapat menjadikan situs Anda memiliki tingkat aksesibilitas yang tinggi, terdapat standar internasional yang dijelaskan secara lengkap dalam situs WAI (Web Accessibility Initiative) yang beralamat di URL www.w3.org/WAI. Misi dari WAI ini adalah untuk mengarahkan web menuju kemampuan penuhnya untuk dapat diakses yang memungkinkan orang-orang dengan keterbatasan dapat berpartisipasi secara sama dalam web.

Dari sisi bisnis, penelitian menunjukkan bahwa situs dengan aksesibilitas yang baik memiliki hasil pencarian yang lebih baik dalam mesin pencari, yang berarti mudah dicari dan ditemukan orang. Selain itu situs juga akan berkurang biaya perawatannya.

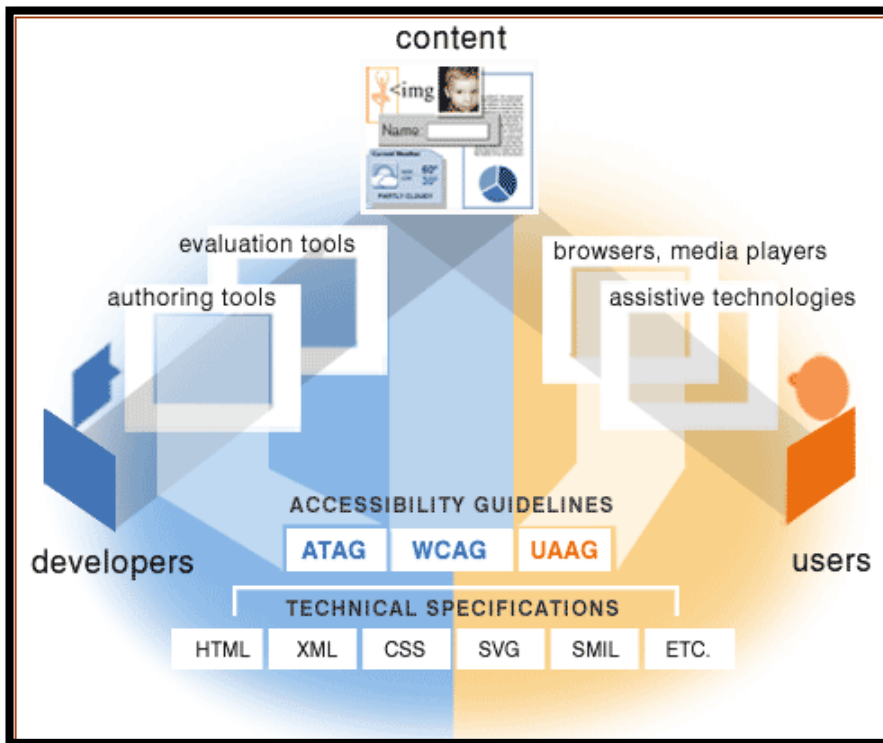
Seperti apa situs web yang beraksesibilitas tinggi? Salah satu kriteria umum untuk meningkatkan aksesibilitas situs adalah dengan menyediakan teks untuk setiap gambar dalam halaman web yang dapat berperan dengan baik sebagai pengganti gambar. Dengan demikian, pengguna yang menggunakan screen reader dalam mengakses web akan mengetahui gambar apa yang ada dalam halaman web meskipun dia tidak dapat melihat.

Komponen-komponen Aksesibilitas

Aksesibilitas web tergantung kepada kerjasama beberapa komponen. Komponen-komponen tersebut antara lain:

1. Konten. Informasi yang terkandung dalam suatu halaman web atau aplikasi web yang mencakup informasi alami seperti teks, gambar, dan suara. Kode atau markup yang mendefinisikan struktur halaman web, aspek presentasi halaman, dan sebagainya.
2. Browser web, media player, dan “user agent” lainnya.
3. Teknologi bantu, seperti screen readers, alternative keyboard, dan sebagainya
4. Pengetahuan dan pengalaman pengguna (*users*)
5. Pengembang (*developers*), termasuk desainer web, pemrogram web, penulis web,
6. Authoring tools, perangkat lunak yang digunakan untuk membuat situs

7. Evaluation tools, alat evaluasi aksesibilitas web seperti HTML validator, CSS validator, dan sebagainya.



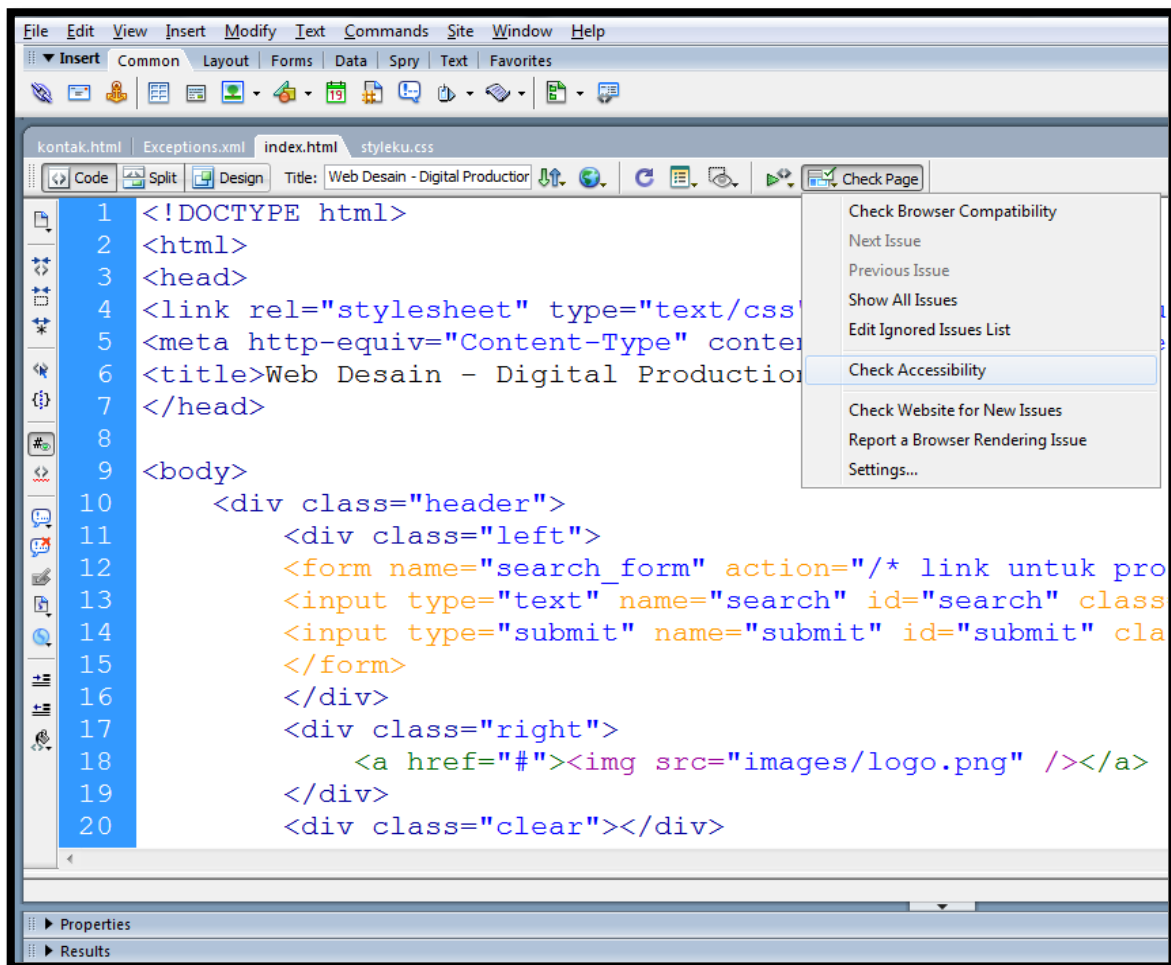
Dukungan dan kerjasama dari berbagai komponen tersebut sangat diperlukan untuk menghasilkan web yang aksesibel. Pihak pengembang perangkat lunak authoring, misalnya Adobe dengan Dreamweavernya, perlu memperhatikan dan mendukung aksesibilitas melalui pemberian fitur agar pengembang situs dapat membuat situs yang aksesibel dengan Dreamweaver.

Konsep aksesibilitas berkaitan erat dengan konsep-konsep lain seperti standar web, Arsitektur Informasi, Usabilitas, dan Findabilitas. Sebelum dapat diakses, situs harus ditemukan terlebih dahulu. Setelah dapat diakses, situs harus memudahkan pengguna untuk menggunakannya. Arsitektur Informasi berupaya agar setiap situs yang dihasilkan memiliki struktur informasi yang baik sehingga mudah diakses dan digunakan, selain itu mudah ditemukan oleh orang maupun mesin pencari. Standar web menyatukan berbagai perbedaan fitur dari berbagai browser sehingga meningkatkan aksesibilitas situs-situs.

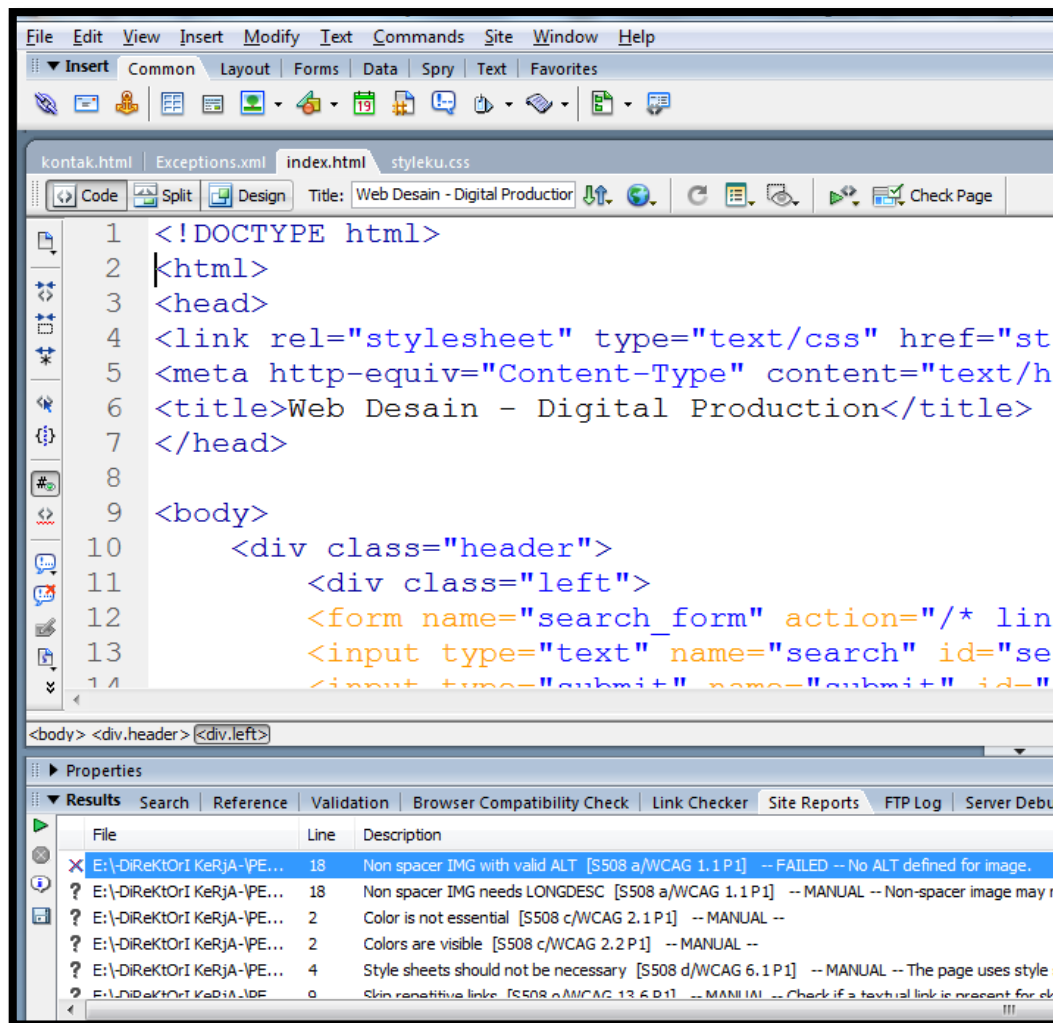
Praktikum

Dalam praktikum kali ini, Anda akan dituntun untuk membuat halaman-halaman web dari situs yang akan atau sedang dikembangkan memiliki tingkat aksesibilitas yang baik. Banyak alat yang dapat digunakan untuk itu, baik yang bersifat offline maupun online. Dalam hal ini, alat yang digunakan untuk melakukan hal ini adalah Dreamweaver. Ikuti instruksi berikut ini:

1. Buka dreamweaver Anda kemudian buka salah satu dokumen HTML yang telah Anda buat, misalnya disini adalah file index.html yang berisi halaman homepage yang didesain dengan Photoshop. Kemudian pilih tool **Check Page - Check Accessibility**.



2. Dreamwaver kemudian menghasilkan laporan pemeriksaan aksesibilitas yang terletak dibawah layar



Disitus tertulis **“Non space IMG with valid ALT (S508 a/WCAG 1.1 P1) --FAILED-- No ALT defined for image”** yang menginformasikan bahwa baris ke-18 dari dokumen HTML terdapat tag `<img>` yang tidak menggunakan atribut `alt` untuk mendefinisikan teks alternatif gambar. Ini adalah salah satu kriteria aksesibilitas sesuai ketentuan S508 a atau WCAG 1.1 P1. S508 atau WCAG adalah standar aksesibilitas yang digunakan secara internasional. Untuk informasi lebih lanjut silahkan kunjungi situs resmi masing-masing standar tersebut.

- Untuk memperoleh informasi berkaitan item laporan aksesibilitas diatas, kunjungi situs WAI dan lihat pada URL <http://www.w3.org/TR/UNDERSTANDING-WCAG20/text-equiv.html>. Disana dijelaskan standar kriteria aksesibilitas ini

www.w3.org/TR/UNDERSTANDING-WCAG20/text-equiv.html

W3C®

Understa

Contents Intro Previous: Understanding Techniques for WCAG Success Criteria Next: SC 1.1.1 [Non-text Content]

Text Alternatives

Understanding Guideline 1.1

Guideline 1.1: Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language.

Intent of Guideline 1.1

The purpose of this guideline is to ensure that all non-text content is also available in [text](#). "Text" refers to electronic text, not an image of text. E unique advantage that it is presentation neutral. That is, it can be rendered visually, auditorily, tactilely, or by any combination. As a result, inform electronic text can be presented in whatever form best meets the needs of the user. It can also be easily enlarged, spoken aloud so that it is eas reading disabilities to understand, or rendered in whatever tactile form best meets the needs of a user.

Note: While changing the content into symbols includes changing it into graphic symbols for people with developmental disorders and speech co difficulties, it is not limited to this use of symbols.

Advisory Techniques for Guideline 1.1 (not success criteria specific)

Specific techniques for meeting each Success Criterion for this guideline are listed in the understanding sections for each Success Criterion (list techniques, however, for addressing this guideline that do not fall under any of the success criteria, they are listed here. These techniques are n for meeting any success criteria, but can make certain types of Web content more accessible to more people.

- Providing sign language videos for audio-only files (future link)

4. Perbaiki masalah aksesibilitas tersebut berdasarkan informasi yang terdapat disana. Kemudian jalankan lagi cek aksesibilitas.

File Edit View Insert Modify Text Commands Site Window Help

Common Layout Forms Data Spy Text Favorites

kontak.html Exceptions.xml index.html styleku.css

Code Split Design Title: Web Desain - Digital Production

```

10 <div class="header">
11 <div class="left">
12 <form name="search_form" action="/* link untuk proses */" method="post">
13 <input type="text" name="search" id="search" class="search" value="Pencar
14 <input type="submit" name="submit" id="submit" class="submit" />
15 </form>
16 </div>
17 <div class="right">
18 <a href="#"></a>
19 </div>
20 <div class="clear"></div>
21 <ul class="navigation">
22 <li><a href="#">Home</a></li>
23 <li><a href="#">About</a></li>

```

<body> <div.header> <div.right> <a>

Properties

Results Search Reference Validation Browser Compatibility Check Link Checker Site Reports FTP Log Server Debug

File	Line	Description
E:\DIREKTORI KeRJA\PE...	18	Non spacer IMG with equivalent ALT [S508 a/WCAG 1.1 P1] -- MANUAL --
E:\DIREKTORI KeRJA\PE...	18	Non spacer IMG needs LONGDESC [S508 a/WCAG 1.1 P1] -- MANUAL -- Non-spacer image may need a LONGDESC attribute.
E:\DIREKTORI KeRJA\PE...	2	Color is not essential [S508 c/WCAG 2.1 P1] -- MANUAL --
E:\DIREKTORI KeRJA\PE...	2	Colors are visible [S508 c/WCAG 2.2 P1] -- MANUAL --
E:\DIREKTORI KeRJA\PE...	4	Style sheets should not be necessary [S508 d/WCAG 6.1 P1] -- MANUAL -- The page uses style sheets to present its content. Some browsers might be unable to understand style sheets, or
E:\DIREKTORI KeRJA\PE...	6	Skip repetitive links [S508 e/WCAG 13.6 P1] -- MANUAL -- Check if a textual link is present for skipping navigation links that are repeated in all the pages.

5. Pada laporan, baris ke-18 tersebut dilaporkan tidak FAILED lagi alias lolos kriteria tersebut.
6. Lakukan perbaikan aksesibilitas untuk baris-baris berikutnya.

Tugas

1. Kunjungi situs validator.w3.org. Kali ini Anda akan memeriksa dokumen HTML Anda secara online. Yang akan dicek dari dokumen HTML Anda adalah validitas markup (kode-kode HTML) dalam dokumen HTML Anda. Buat laporan aktifitas Anda yang berisi:
 - Listing kode dokumen HTML yang akan Anda cek validitasnya
 - Screenshoot laporan hasil pemeriksaan validitas dengan validator ini.
 - Penjelasan perbaikan yang telah dilakukan berdasarkan laporan hasil pemeriksaan
 - Screenshoot laporan hasil pemeriksaan setelah dilakukan perbaikan
2. Dikumpulkan pada pertemuan praktikum berikutnya.

Findabilitas

Tujuan Pembelajaran

Mempraktekkan aspek-aspek findabilitas sehingga situs mudah dicari

Pokok Bahasan

Findabilitas, tingkat kemudahan dicari dan ditemukan

Indikator Keberhasilan

Mahasiswa dapat membuat situsnya berpotensi mudah dicari

Penjelasan

Isitilah findabilitas digunakan untuk menyatakan kemudahan ditemukan informasi yang terkandung dalam sebuah situs web. Baik itu melalui eksternal situs (menggunakan mesin pencari dan sejenisnya) dan oleh pengguna yang berada didalam situs itu sendiri. Banyak situs yang berelevansi dengan kunci pencarian tidak muncul dalam daftar teratas hasil pencarian mesin pencari dikarenakan perancang dan pengembang situs tidak memperhatikan algoritma perankingan yang sedang bekerja saat ini. Arti penting findabilitas dapat ditemukan dalam hukum pertama e-commerce yang menyatakan bahwa “Jika pengguna tidak dapat menemukan produk, maka pengguna tidak dapat membeli produk.” Konsep findabilitas meliputi aspek Arsitektur Informasi, Desain Antarmuka Pengguna, Aksesibilitas, dan Optimasi Mesin Pencari (SEO).

Findability mirip tetapi berbeda dengan discoverability, yang didefinisikan sebagai kemampuan sesuatu, terutama sebuah konten atau informasi, dapat ditemukan. Findabilitas berbeda dari konsep

pencarian web. Findabilitas berkaitan dengan menemukan sesuatu di ruang yang dikenal sementara pencarian web mengacu pada 'mencari' sesuatu di ruang yang tidak diketahui.

Mark Baker, penulis buku "Every Page is Page One", menyebutkan bahwa findability "adalah masalah konten, bukan masalah pencarian". Sering terjadi suatu konten berhasil ditemukan, tetapi pengguna sering menemukannya jauh di dalam isi suatu situs. Findabilitas sempurna tidak mungkin tercapai, tapi harus diusahakan untuk mengurangi upaya pengguna untuk menemukan informasi yang mereka butuhkan.

Findabilitas dapat dibagi menjadi dua jenis, findabilitas eksternal dan on-site findabilitas. Findabilitas eksternal berhubungan dengan internet marketing dan search engine optimization (SEO). Beberapa faktor yang mempengaruhi findabilitas eksternal:

1. Pengindeksan oleh mesin pencari. Sebagai langkah pertama, halaman web harus bisa ditemukan oleh indexing crawler (peramban mesin pencari) agar dapat ditampilkan dalam hasil pencarian. Oleh karenanya hindari faktor-faktor yang dapat menyebabkan halaman web diabaikan oleh peramban mesin pencari. Salah satu cara yang dapat dilakukan adalah mempelajari algoritma pencarian mesin pencari agar situs dapat disesuaikan dengan algoritma tersebut, misalnya dengan menggunakan elemen heading (h1 s/d h6) secara tepat sesuai levelnya.
2. Deskripsi halaman di Hasil Pencarian. Setelah halaman web dapat diindeks oleh mesin pencari, langkah berikutnya adalah berusaha untuk menarik pengguna mengklik link ke halaman web. Pada saat itu, pengguna tidak dapat melihat seluruh halaman web, mereka hanya bisa melihat kutipan dari isi halaman web dan metadata. Oleh karena itu, penting untuk menampilkan informasi yang bermakna dalam ruang terbatas, yang biasanya hanya bisa beberapa kalimat, dalam hasil pencarian.
3. Kecocokan kata kunci. Secara semantik, atau makna kata, istilah yang digunakan oleh pengguna dan pembuat konten seringkali berbeda. Menjembatani kesenjangan ini akan sangat membantu membuat konten yang lebih findabel. Misalnya, dengan melakukan penelitian kata kunci untuk mencari kata kunci yang tepat bagi suatu konten.

On-site findabilitas berhubungan dengan kemampuan pengguna potensial untuk menemukan apa yang mereka cari pada suatu situs. Lebih dari 90 persen pengguna menggunakan alat pencarian internal situs dibandingkan browsing. Dari jumlah tersebut, hanya 50 persen yang berhasil menemukan yang mereka cari. Beberapa faktor yang mempengaruhi findabilitas ini:

1. Alat pencarian dalam situs. Jika pencari tidak menemukan apa yang mereka cari, mereka cenderung untuk meninggalkan situs. Pencari yang menemukan konten yang mereka cari berpotensi dua kali lipat untuk melakukan transaksi.
2. Produk dan link terkait. Pengalaman pengguna dapat ditingkatkan dengan mencoba mengetahui kebutuhan mereka dan menyediakan saran untuk produk dan informasi yang serupa.
3. Menyesuaikan situs dengan kebutuhan dan pilihan pengguna. Desain situs, konten yang kreatif, dan rekomendasi merupakan faktor utama yang mempengaruhi pengalaman pengguna.
4. Pengalaman lintas perangkat. Dengan munculnya perangkat komputasi selain komputer desktop, kita perlu memperhatikan kelancaran transisi antara perangkat untuk meningkatkan kepuasan pelanggan.

Strategi

Untuk membuat situs menjadi findabel, berikut beberapa strategi yang dapat dilakukan:

Aspek markup

- Gunakan markup yang valid yang tidak mengandung kesalahan yang dapat membuat halaman web anda sulit untuk diindeks oleh mesin pencari. Gunakan alat validasi W3C di <http://validator.w3.org>.
- Tulis markup yang bermakna secara semantik. Ini berarti gunakanlah tag sesuai dengan fungsi tag. Gunakan ekstraktor semantic W3C untuk memeriksa pekerjaan Anda (<http://www.w3.org/2003/12/semantic-extractor.html>)
- Gunakan teks tag title yang cerdas yang dimulai dengan nama halaman diikuti nama situs dan frase singkat yang kaya kata kunci yang mendeskripsikan situs.
- Gunakan tag heading untuk mendefinisikan bagian-bagian kunci halaman web Anda. Berhati-hati dalam menyertakan kata kunci yang relevan. Tempatkan tag heading dalam halaman dengan urutan sesuai tingkat pentingnya dimulai dengan heading 1 untuk nama situs atau organisasi.
- Gunakan tag “meta description” untuk menyediakan deskripsi tentang situs Anda untuk ditampilkan dalam halaman hasil mesin pencari.
- Usahakan sisi presentasi (CSS) dan perilaku (Javascript) berjenis eksternal sehingga mesin pencari tidak perlu mengunduhnya ketika sedang mengindeks halaman-halaman web Anda.
- Sertakan atribut “alt” dan “title” dalam tag image. Jika perlu tambahkan link “longdesc” untuk teks penjelasan terperinci tentang image.

- Gunakan tag dan untuk menandai kata kunci di area tertentu dari halaman web Anda untuk menekankan pada mesin pencari pentingnya kata tersebut.
- Jangan menggunakan tabel untuk mengatur layout. Gunakan tabel sesuai fungsinya, menampilkan data dalam bentuk tabular. Sertakan teks yang deskriptif dan kaya kata kunci menggunakan atribut “summary” dari tag <table> dan tag “caption”.
- Gunakan CSS sprites untuk menjaga request HTTP tetap minimal. Dengan demikian proses loading halaman web menjadi lebih cepat.
- Gunakan tag <abbr> dan <acronym> bersama atribut “title”-nya untuk menyediakan full teks dari semua akronim dan singkatan.
- Gunakan rel=“nofollow” dalam tag anchor yang berhubungan dengan situs-situs yang reputasinya dipertanyakan atau tidak diketahui.

Aspek Server

- Buat URL yang ramah terhadap mesin pencari, yang tidak mengandung string query, merefleksikan struktur navigasi Anda, dan mudah ditebak.
- Pilih nama domain yang berisi kata kunci yang ditargetkan.
- Sertakan kata kunci dalam nama file dan folder di seluruh bagian situs Anda.
- Buat halaman 404 yang dirancang sendiri untuk mengarahkan pengguna yang tersesat.
- Cegah file dan folder tertentu diindeks jika perlu dengan menggunakan file robot.txt.
- Buat file sitemap.xml dan beritahukan pada semua mesin pencari utama. Sertakan petunjuk ke file sitemap.xml Anda dalam robot.txt.

Aspek Konten

- Sertakan konten pada situs anda yang melayani kepentingan dan kebutuhan pengguna/pengunjung situs Anda.
- Sertakan berbagai jenis konten yang bervariasi dalam situs anda
- Buat halaman homepage yang dapat secara efektif menyalurkan trafik ke area yang kaya konten dari situs anda.
- Lakukan kajian tentang kata kunci yang paling menguntungkan terhadap target pengguna.
- Tempat kata kunci di lokasi yang strategis diseluruh bagian situs.
- Pastikan Javascript tidak digunakan untuk membuat navigasi situs.
- Hindari penggunaan frame.
- Lainnya
- Gunakan validator link untuk mencari link yang putus yang perlu diperbaiki.
- Pastikan blog Anda di setup untuk mengumumkan jika ada konten yang baru

Praktikum

Berdasarkan strategi diatas, lakukan perubahan yang dapat Anda lakukan terhadap file-file situs yang sudah Anda buat pada praktikum-praktikum sebelumnya untuk meningkatkan aspek findabilitas situs Anda.

Tugas

Catat segala perubahan yang Anda lakukan dan masukkan kedalam laporan yang dikumpulkan pada praktikum berikutnya

Hosting Situs

Tujuan Pembelajaran

Mempraktekkan proses hosting file-file situs ke Internet

Pokok Bahasan

- Pertimbangan memilih layanan hosting
- Mengupload file ke layanan hosting

Indikator Keberhasilan

Mahasiswa dapat mengupload file-file situsnya ke salah satu layanan hosting

Penjelasan

Setelah selesai mendesain dan membuat website secara offline, langkah selanjutnya adalah mengupload website tersebut ke salah satu layanan hosting. Upload adalah proses memindahkan file-file dari komputer Anda ke server hosting sehingga nantinya bisa diakses oleh banyak orang di seluruh dunia.

Untuk teknik upload website ada beberapa cara, berikut ini 2 cara yang umum dalam melakukan upload file-file website ke server hosting:

1. Upload menggunakan File Manager (atau Legacy File Manager).

Keuntungan mengupload file menggunakan file manager adalah Anda tidak perlu menginstal apapun di server, cukup menggunakan browser dan mengakses CPanel saja. Sedangkan kelemahannya adalah folder tidak bisa ikut diupload, sehingga harus membuat folder secara

manual. Selain itu proses upload dilakukan satu-persatu file. File manager hanya bisa mengupload 15 file sekaligus dan harus berada di folder yang sama.

2. Upload menggunakan software FTP

Upload file menggunakan software FTP cukup mudah dan menyenangkan. Selain bisa mengupload banyak file sekaligus (termasuk folder-foldernya), ketika koneksi terputus dan proses upload terhenti, Anda bisa melanjutkannya sewaktu-waktu. Selain itu upload menggunakan FTP lebih bisa diandalkan ketika harus mengupload file-file dalam ukuran besar.

Software-software FTP yang banyak digunakan antara lain adalah FileZilla dan CuteFTP, direkomendasikan menggunakan FileZilla karena gratis dan mudah digunakan.

Ada beberapa faktor yang bisa kita jadikan pertimbangan ketika memilih hosting, antara lain:

1. Kebutuhan hosting.

Seperti apa kebutuhan hosting kita, akan sangat menentukan besaran space hosting yang kita ambil, lokasi server, dan support hosting yang diperlukan. Jika website Anda adalah website baru, maka kapasitas hosting yang diperlukan tidak sebesar website lama / existing website.

2. Jenis website yang akan dibangun.

Sebagai gambaran, jika Anda hendak membuat toko online, tentunya akan sedikit berbeda ketika Anda membuat website profil pribadi, karena toko online membutuhkan space yang lebih besar untuk menampung gambar-gambar produk yang banyak.

3. Lokasi server.

Siapa target market Anda? Apakah pengunjung lokal atau luar negeri? Mungkin Anda perlu tahu, lokasi server yang ditawarkan oleh para penyedia hosting amat beragam, antara lain:

- **Hosting IIX** = lokasi server di Indonesia
- **Hosting USA** = lokasi server di USA
- **Hosting SG** = lokasi server di Singapore
- **Hosting EU** = lokasi server di Eropa

Idealnya, semakin dekat lokasi server dengan pengunjung situs, maka kecepatan proses loading situs akan lebih cepat bagi pihak pengunjung. Meskipun begitu, kecepatan server

tidak hanya dipengaruhi oleh lokasi tetapi juga faktor lain seperti spesifikasi hardware server. Oleh karena itu perhatikan juga spesifikasi server yang ditawarkan oleh layanan web hosting.

- Jangan terjebak dengan istilah unlimited.

Jika Anda sering melihat berbagai paket hosting, tentunya akan sering menemui istilah “Unlimited Bandwidth” atau “Unlimited Space”. Jangan terkecoh oleh istilah ini. Karena pada dasarnya semua resource memiliki batasan. Termasuk kapasitas bandwidth dan space hosting. Jadi yang dimaksud unlimited di sini adalah Anda berhak menggunakan fasilitas space atau bandwidth secara unlimited hingga batas limit sesuai spesifikasi hardware server dan sesuai kebijakan masing-masing penyedia hosting.

- Utamakan ekstensi domain “.com”.

Ketika hendak memilih nama domain, usahakan cari yang ekstensi domain .com terlebih dahulu, baru kemudian .net, .org, .web dan lain sebagainya. Karena secara psikologis, orang akan cenderung mengetikkan ekstensi domain .com disaat menuliskan alamat suatu situs. Karena memang ekstensi tersebutlah yang paling populer.

- Keandalan dan uptime

Jika kita *meng-host* pada server yang memiliki *uptime* di bawah 50%, maka kita akan banyak kehilangan pengunjung, bisnis, dan pendapatan! Web hosting yang kita pilih harus memiliki jaminan *uptime* yang baik. Memang hampir semua hosting tidak ada yang memiliki jaminan *uptime* 100%, tapi setidaknya cari yang memiliki *uptime* minimal 99%. Periksa juga redundansi *host*, tanyakan bagaimana cara mereka mengatasi kegagalan fungsi jika terjadi *error*/kerusakan. Misalnya, generator cadangan dan *Uninterruptible Power Supply*(UPS).

- Dukungan pelanggan

Layanan dan dukungan pelanggan juga harus benar-benar kita perhatikan ketika memilih web hosting. Ciri web hosting yang baik adalah memiliki tim dukungan (teknis) handal yang siap membantu kita kapanpun jika dibutuhkan selama 24/7, termasuk hari libur. Karena server dan jaringan harus selalu dipantau setiap saat. Dukungan pelanggan bisa melalui email, telepon, dan *live chat*.

- Backup data

Pilihlah web hosting yang juga menyediakan layanan *backup* data untuk file kita. Karena jika terjadi hal yang tidak terduga, website bisa di *restored* dengan mudah. Cari tahu seberapa

sering server bisa melakukan *backup*. Ingat, meskipun web hosting yang kita gunakan memiliki fitur backup sekalipun, kita tetap harus melakukan *backup* sendiri ke hardisk untuk berjaga-jaga.

- **Fleksibilitas**

Mengingat bisnis akan terus tumbuh dan berkembang, maka kebutuhan akan sumber daya juga semakin meningkat. Penting memilih web hosting yang memiliki fleksibilitas tinggi dan memungkinkan kita untuk dengan mudah melakukan *upgrade* sumber daya di kemudian hari. Ini termasuk penambahan ruang penyimpanan, RAM, processor, jumlah akun email, dan yang lainnya.

- **Kapasitas penyimpanan dan transfer data**

Kapasitas ini penting untuk dipertimbangkan, pastikan kapasitas ruang penyimpanan cukup untuk menampung file-file website kita. Sesuaikan kapasitas penyimpanan dan transfer data (*bandwidth/traffic*) data yang ditawarkan dengan kebutuhan website kita saat ini.

- **Fitur yang ditawarkan**

Sistem operasi pada server yang kita gunakan menentukan jenis script apa saja yang bisa kita jalankan. Server Linux memang jauh lebih murah dan stabil, tapi misal kita membutuhkan host untuk script ASP maka tidak ada pilihan lain selain menggunakan hosting Windows. Terlepas dari apapun sistem operasi yang digunakan, pastikan server memiliki beberapa fitur dasar berikut ini:

- FTP, SSH
- PHP, Perl, CGI
- Subdomain
- Statistik dan logs
- FrontPage extensions
- Email

- **Control panel yang mudah digunakan**

Pilihlah hosting yang memiliki control panel yang mudah digunakan untuk mengelola akun web dan email kita. Jika memungkinkan mintalah demo untuk melihat *control panel hosting* tersebut, dengan begitu kita bisa mencoba dan merasakan bagaimana mengelola server website kita.

Praktikum

Pelajari pertimbangan dalam memilih layanan hosting yang baik dan sesuai dengan kebutuhan Anda

Tugas

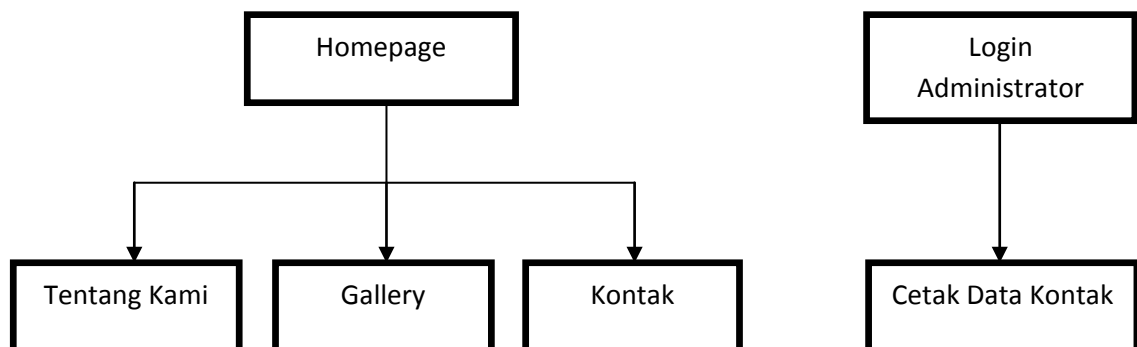
1. Upload situs personal yang halaman demi halamannya Anda buat di setiap praktikum. Jika Anda mengikuti kegiatan praktikum dengan benar, maka saat ini Anda sudah memiliki sebuah situs web personal yang terdiri dari halaman-halaman:

- Homepage
- Tentang kami
- Gallery
- Kontak

Selain itu terdapat 2 halaman administrator yang tersimpan dalam direktori admin, yaitu:

- Login administrator
- Cetak Data Kontak

Berikut Peta Situs dari halaman-halaman yang telah Anda buat:



2. Pada pertemuan berikutnya sampaikan alamat URL situs Anda untuk diperiksa dan diambil nilainya sebagai nilai Tugas Akhir

Lampiran 1: Organisasi CSS

Ketika Anda bekerja dengan file CSS yang memiliki ribuan deklarasi, maka Anda akan mudah untuk menjadi kewalahan hanya untuk mencari aturan CSS yang diinginkan ketika suatu saat perlu melakukan perbaikan atau peningkatan tampilan. Cara terbaik untuk menghindari hal ini adalah dengan mengorganisasikan aturan-aturan CSS Anda sehingga benar-benar terorganisir. Berikut contoh pengorganisasian yang logis dan banyak dilakukan oleh para pemrogram web yang dimulai dengan pengaturan struktur dasar layout tampilan:

```
1. /*****
2.
3. ----- UMUM -----
4.
5. *****/
6.
7. html {}
8.
9. body {}
10.
11. a {}
12.
13.
14.
15. /*****
16.
17. ----- STRUKTUR UTAMA -----
18.
19. *****/
20.
21. #container {}
22.
23. #header {}
24.
25. #content {}
26.
27. #kolomutama {}
28.
29. #kolomsamping {}
30.
31. #footer {}
32.
33. #copyright {}
34.
```

```

35.
36.
37. /*****
38.
39. ----- HEADER -----
40.
41. *****/
42.
43. /* -- ***** Main Navigation
44.
45. -----*/
46.
47. #mainnav {}
48.
49.
50.
51. /* --***** Secondary Navigation
52.
53. -----*/
54.
55. #subnav {}
56.
57. /*****
58.
59. ----- KOLOM UTAMA -----
60.
61. *****/
62.
63. #kolomutama {}
64.
65.
66.
67. /* -- ***** Home Page
68.
69. -----*/
70.
71. #bd-home {}
72.
73.
74.
75. /* --***** About Page
76.
77. -----*/
78.
79. #bd-about {}

```

```

80.
81. /* --***** Photography Page
82.
83. -----*/
84.
85. #bd-photography {}
86.
87. /* --***** Contact Page
88.
89. -----*/
90.
91. #bd-contact {}
92.
93.
94.
95. /*****
96.
97. ----- SIDE COLUMN -----
98.
99. *****/
100.
101. #sidecol {}
102.
103.
104.
105. /*****
106.
107. ----- FOOTER & COPYRIGHT-----
108.
109. *****/
110.
111.
112.
113. /* --***** Footer
114.
115. -----*/
116.
117. #footer {}
118.
119. /* --***** Copyright
120.
121. -----*/
122.
123. #copyright {}
124.

```

125. /** -----Third Level ----- **

126.

127.

128.

129. /* -- fourth level-- */