

Pendahuluan

CSS yang memiliki kepanjangan *Cascade Style Sheet* ini digunakan para web designer untuk mengatur style elemen yang ada dalam halaman web mereka, mulai dari memformat text, sampai pada memformat layout. Tujuan dari penggunaan CSS ini adalah supaya diperoleh suatu kekonsistenan style pada elemen tertentu.

Perkembangan CSS sendiri diawali pada tahun 1996, dimana W3C (World Wide Web Consortium), sebuah konsorsium untuk standarisasi web, menyusun draft proposal untuk membuat CSS ini dan akhirnya dapat berjalan. Selanjutnya pada pertengahan tahun 1998, W3C mengembangkan CSS2 yang diperbarui untuk kepentingan media lain (tidak hanya untuk PC web browser). Akhirnya mulai pada tahun 2000, telah dikembangkan CSS3 oleh W3C yang sampai saat ini masih terus diperbarui lagi.

Namun pada saat ini, hanya CSS2 yang masih didukung oleh kebanyakan web browser, tapi tidak demikian halnya untuk CSS3. Masih sedikit web browser yang mendukung CSS3. Oleh karena itu pada tutorial ini hanya akan dibahas mengenai CSS2 saja.

Keuntungan Penggunaan CSS

Dengan menggunakan CSS, Anda (sebagai webmaster) akan lebih mudah untuk mengatur style elemen dalam halaman web Anda. Sebagai contoh, misalnya untuk mengatur style elemen heading, Anda menginginkan jenis font nya adalah Arial, 20 pixel dan berwarna merah. Dengan CSS, Anda cukup menuliskan properti dari elemen heading tersebut sekali saja, dan Anda akan memperoleh hasil yang diinginkan. Bandingkan apabila tidak menggunakan CSS, Anda akan mengatur style pada setiap elemen heading yang ada. Tentu saja hal ini akan sangat merepotkan. Perhatikan contoh berikut ini

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">
h1 {
    color: red;
    font-family: "arial";
}
</style>

</head>

<body>
```

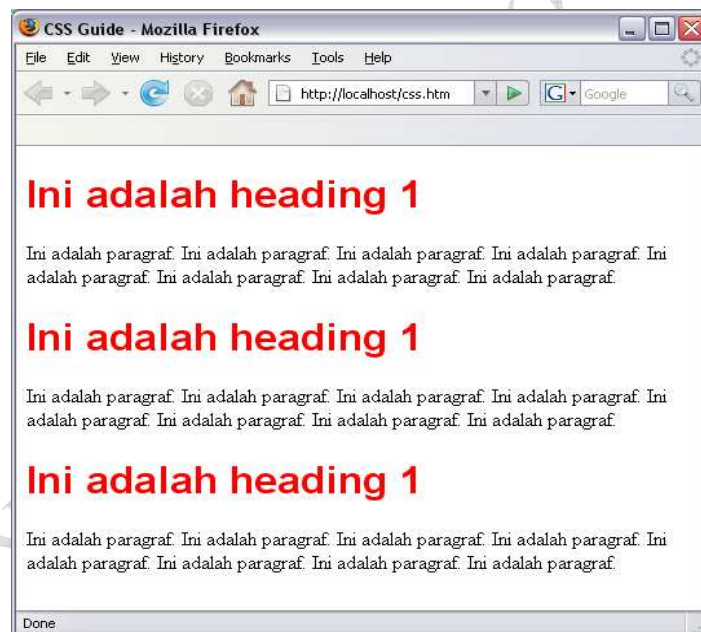
```
<h1>Ini adalah heading 1</h1>
<p>Ini adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini
adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. </p>

<h1>Ini adalah heading 1</h1>
<p>Ini adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini
adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. </p>

<h1>Ini adalah heading 1</h1>
<p>Ini adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini
adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. </p>

</body>
</html>
```

Code di atas akan menghasilkan tampilan sebagai berikut.



Perhatikan code sebelumnya! Untuk mengatur style elemen heading 1 (h1) cukup menuliskannya pada CSS sekali saja. Bandingkan apabila Anda tidak menggunakan CSS, maka codenya akan berbentuk seperti berikut untuk mendapatkan hasil yang sama.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
<title>CSS Guide</title>

</head>

<body>

<h1><font face="arial" color="red">Ini adalah heading 1</font></h1>
<p>Ini adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini
adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. </p>

<h1><font face="arial" color="red">Ini adalah heading 1</font></h1>
<p>Ini adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini
adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. </p>

<h1><font face="arial" color="red">Ini adalah heading 1</font></h1>
<p>Ini adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini
adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. </p>

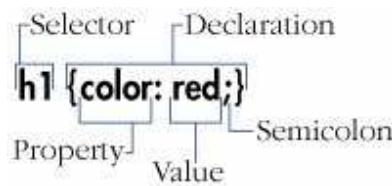
</body>
</html>
```

Anda akan menuliskan style pada setiap elemen heading 1 yang dibuat. Tentu hal ini akan merepotkan. Apalagi suatu saat Anda bermaksud untuk mengubah style nya, maka Anda akan mengubahnya satu-persatu. Sangat-sangat merepotkan ☹.

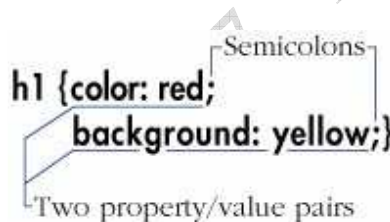
Selain itu, dengan CSS akan diperoleh suatu kekonsistenan style. Anda tentu dapat merasakan keuntungan ini, dibandingkan dengan tanpa adanya CSS yang kemungkinan besar ketidakkonsistenan akan terjadi. Hal ini dikarenakan style hanya dituliskan sekali, tidak seperti dengan tanpa CSS yang ditulis style berulang kali. Meskipun Anda masih bisa mengandalkan copy and paste... masih saja ada peluang terjadinya ketidakkonsistenan.

Struktur Style

Style pada CSS memiliki struktur sebagai berikut



Selector digunakan untuk menentukan pada elemen apa style tersebut diterapkan. Selector dapat berupa nama id elemen atau nama class. Bagian deklarasi menerangkan style yang akan dibuat. Bagian ini terdiri dari properti dan value. Properti dapat diisi dengan jenis warna, ukuran, perataan margin dll, sedangkan value diisi dengan nilai dari propertinya, misalnya red untuk warna dll. Setiap akhir penulisan properti dan value harus diakhiri dengan tanda titik koma (semicolon). Tanda ini juga digunakan sebagai pemisah antar satu properti dengan properti lain.



Apabila Anda lupa memberikan tanda titik koma, maka style tersebut tidak akan dijalankan oleh browser.

Mungkin Anda bertanya, bagaimana jika kita akan membuat style untuk beberapa elemen yang berbeda namun stylenya sama? Misalkan kita akan membuat style untuk elemen h1, h2, h3 namun memiliki style yang sama. Berikut ini adalah sintaksnya:

```
h1, h2, h3 { properti1 : value1;
             properti2 : value2;
             properti3 : value3;
             .
             .
             .
}
```

Lebih Lanjut Tentang Selector

Dalam CSS, terdapat cascade yang merupakan metode untuk mengatur style supaya tidak terjadi konflik. Untuk lebih jelasnya perhatikan contoh berikut ini.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">
p {
    color: red;
}

em {
    color: green;
}
</style>

</head>

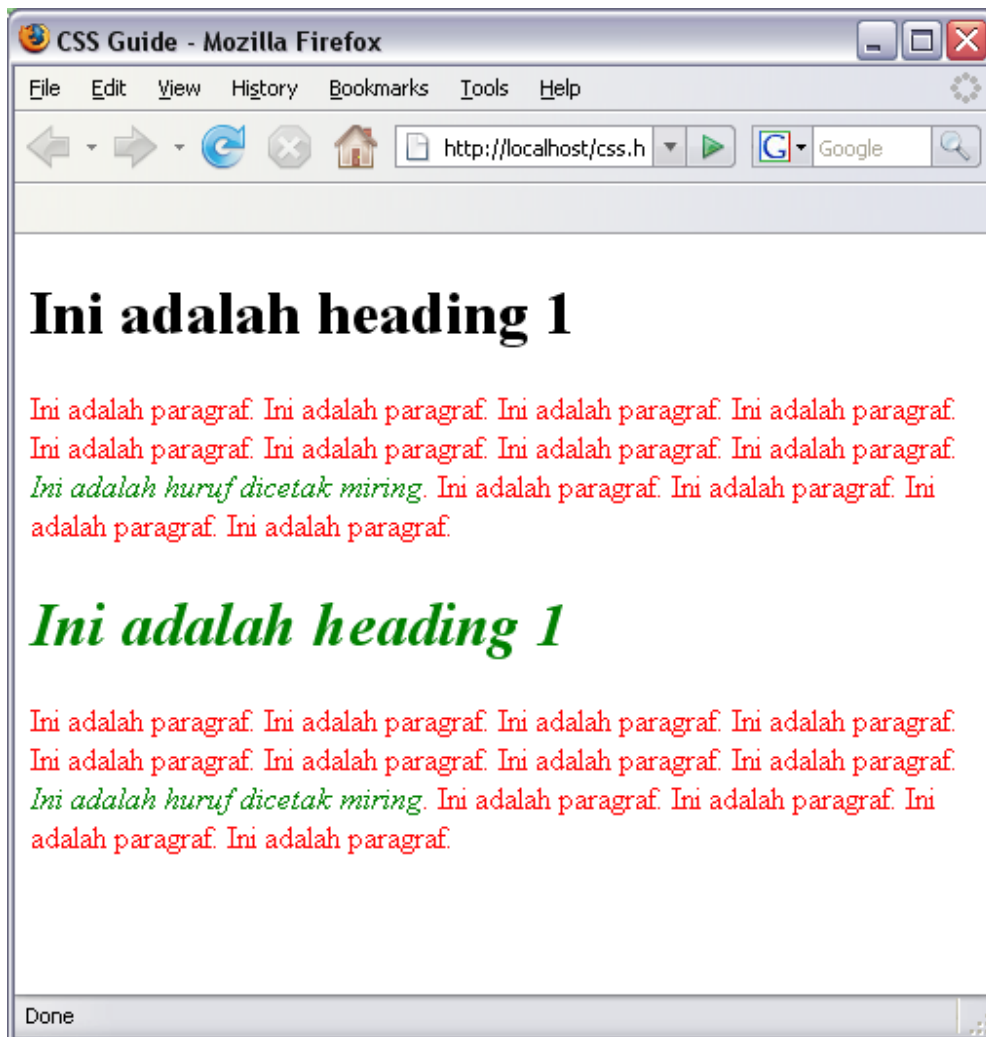
<body>

<h1>Ini adalah heading 1</h1>
<p>Ini adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini
adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. <em>Ini adalah huruf dicetak
miring</em>. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. </p>

<h1><em>Ini adalah heading 1</em></h1>
<p>Ini adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini
adalah paragraf. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. <em>Ini adalah huruf dicetak
miring</em>. Ini adalah paragraf. Ini adalah paragraf. Ini adalah
paragraf. Ini adalah paragraf. </p>

</body>
</html>
```

Pada code di atas terdapat style untuk mengatur tampilan elemen paragraf (<p>) dan untuk emphasize atau (cetak miring seperti halnya italic). Perhatikan apa yang terjadi dengan tampilannya di browser



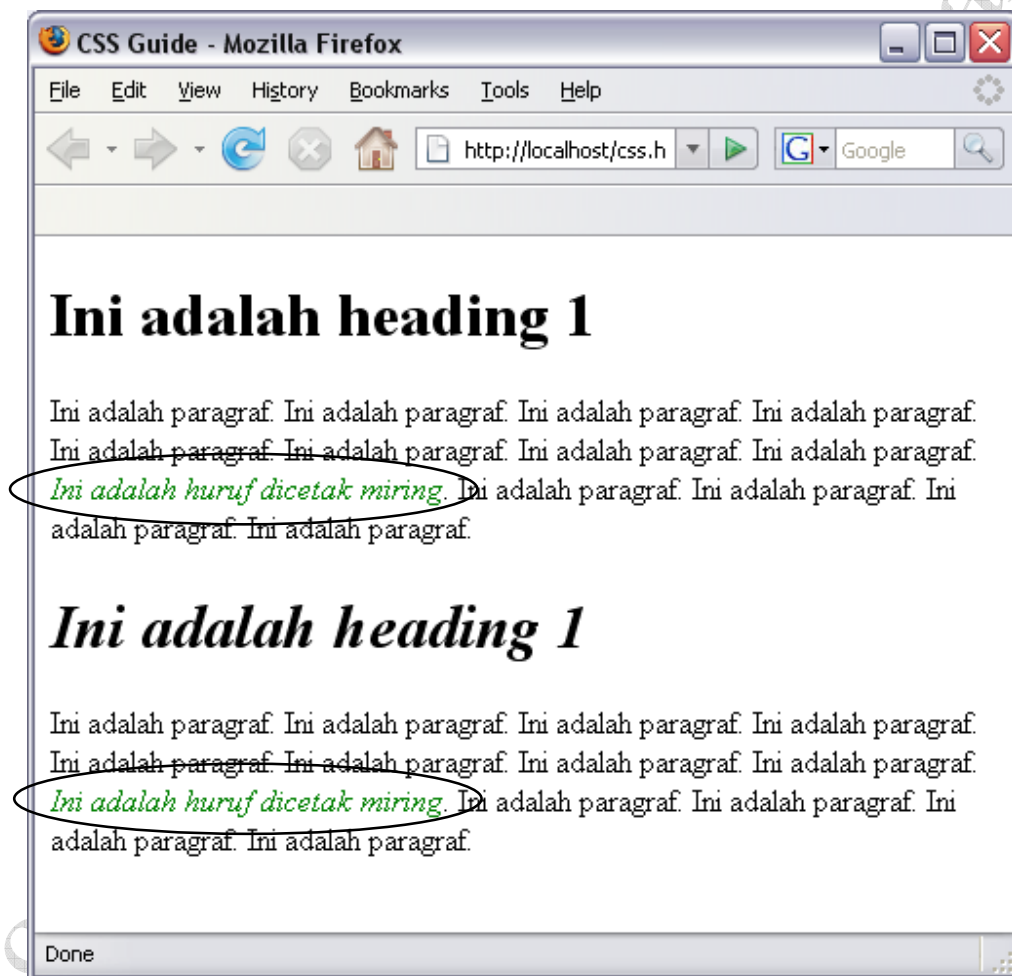
Dengan Anda mendeklarasikan style warna hijau untuk elemen ``, maka style tersebut akan berlaku untuk setiap elemen tersebut (termasuk pada bagian heading). Tentu saja hal ini membuat tampilan halaman web tidak bagus, karena style heading menjadi tidak konsisten (terdapat heading yang berwarna hijau). Lantas.. bagaimana cara mengatur style supaya warna hijau hanya berlaku untuk `emphasize` yang terletak di dalam paragraf saja? Untuk itu Anda harus mengubah deklarasi stylenya menjadi seperti berikut

```
<style type="text/css">
p {
    color: red;
}
```

```
p em {  
    color: green;  
}  
</style>
```

Maksud dari code yang bercetak merah tersebut adalah bahwa style warna hijau diterapkan pada elemen emphasize yang berada di dalam elemen paragraf <p>.

Hasil tampilan setelah menggunakan style yang telah dimodifikasi tersebut adalah:



Selector Dengan Class dan Nama ID

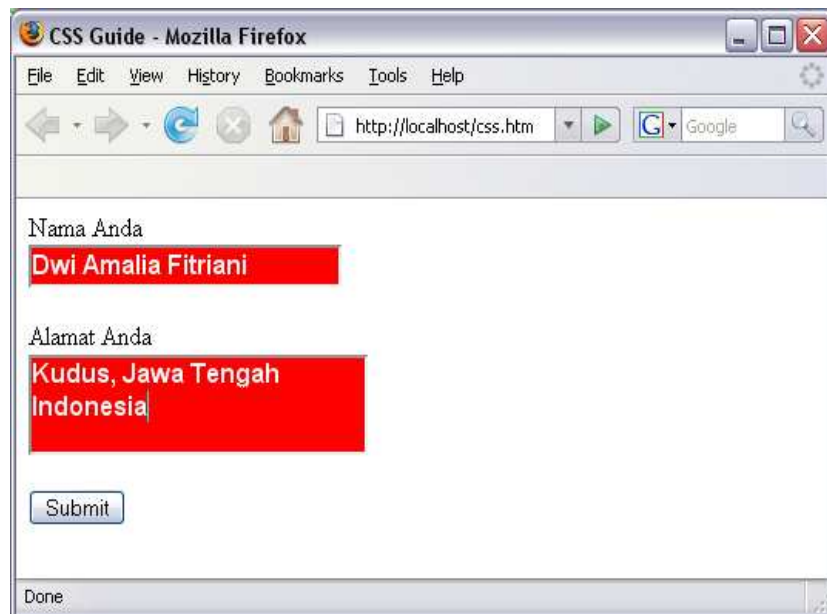
Class dapat digunakan untuk mengelompokkan beberapa elemen supaya memiliki style sama. Secara umum sintaks style untuk class tertentu adalah sbb:

```
.namaKelas {  
    properti1 : value1;  
    properti2 : value2;  
    .  
    .  
}
```

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<title>CSS Guide</title>  
  
<style type="text/css">  
.inputText {  
    color: #FFFFFF;  
    font-family: arial;  
    background: #FF0000; /* membuat background merah */  
    font-weight: bold; /* membuat text bold */  
}  
  
</style>  
</head>  
  
<body>  
  
<form name="form1">  
Nama Anda <br />  
<input type="text" name="alamat" class="inputText" /><br /><br />  
Alamat Anda <br />  
<textarea class="inputText"></textarea><br /><br />  
<input type="submit" name="submit" value="Submit" />  
</form>  
</body>  
</html>
```

Hasilnya adalah sebagai berikut:



Style juga dapat digunakan untuk mengatur tampilan elemen dengan id tertentu. Berikut ini adalah contohnya:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

p {
    color: #FFFFFF;           /* membuat warna text putih */
    font-family: arial;
    background: #FF0000;     /* membuat background merah */
    font-weight: bold;
}

p.group {
    color: #FF0000;          /* membuat warna text putih */
    font-family: arial;
    background: #FFFFFF;     /* membuat background putih */
}

p#one {
    color: blue;
    font-family: "times new roman";
}

</style>
```

```
</head>

<body>

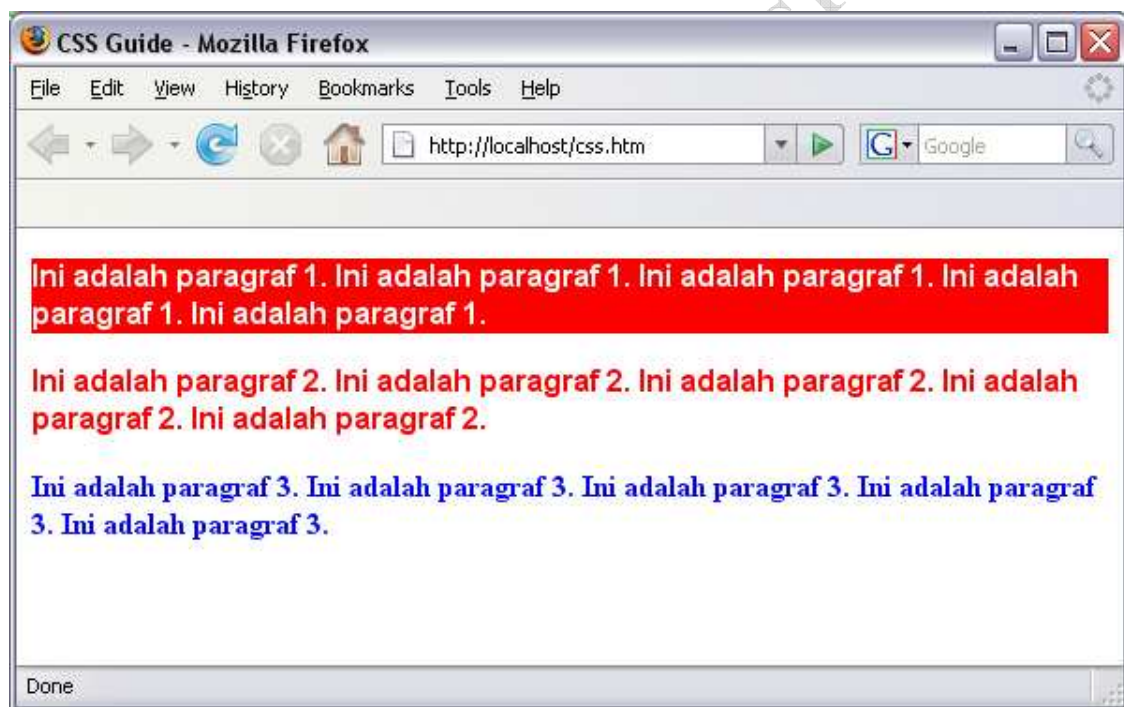
<p>Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. </p>

<p class="group">Ini adalah paragraf 2. Ini adalah paragraf 2. Ini
adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. </p>

<p class="group" id="one">Ini adalah paragraf 3. Ini adalah paragraf 3.
Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3.
</p>

</body>
</html>
```

Perhatikan hasil tampilan dari code di atas berikut ini



Pada tampilan di atas, tampak bahwa perintah

```
p {
    .
    .
    .
}
```

digunakan untuk mengatur style pada elemen `<p>`. Sedangkan perintah

```
p.group {  
    .  
    .  
    .  
}
```

digunakan untuk mengatur style hanya pada `<p>` yang memiliki class bernama group dan perintah

```
p#one {  
    .  
    .  
    .  
}
```

berfungsi untuk mengatur style pada `<p>` yang memiliki id bernama one. Meskipun pada contoh ini, terdapat elemen `<p class="group" id="one">`, namun style pada elemen ini mengikuti `p#one`, bukannya `p.group` karena nama id lebih diprioritaskan daripada class.

Lantas bagaimana dengan tampilan dengan style berikut ini

```
<style type="text/css">
```

```
p    {  
    color: black;  
}  
  
p.group {  
    color: red;  
}  
  
.group {  
    color: blue;  
}  
  
#group {  
    color: green;  
}
```

```
</style>
```

yang diterapkan pada tag berikut?

```
<p>Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.  
Ini adalah paragraf 1. Ini adalah paragraf 1. </p>
```

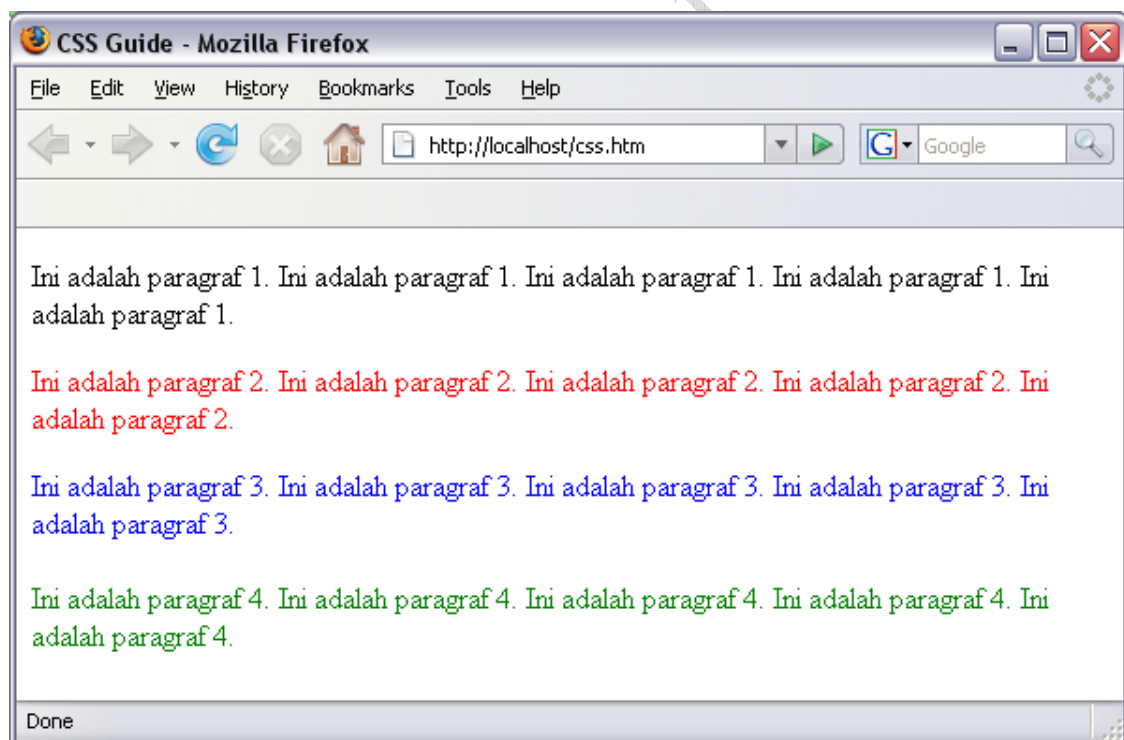
```
<p class="group">Ini adalah paragraf 2. Ini adalah paragraf 2. Ini  
adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. </p>
```

```
<div class="group">Ini adalah paragraf 3. Ini adalah paragraf 3. Ini  
adalah paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3. </div>
```

```
<div id="group">Ini adalah paragraf 4. Ini adalah paragraf 4. Ini  
adalah paragraf 4. Ini adalah paragraf 4. Ini adalah paragraf 4. </div>
```

Apabila Anda dapat menjawab pertanyaan di atas, maka Anda sudah menguasai dasar-dasar CSS awal ini khususnya pada masalah selector.

Menurut analisis saya, untuk paragraf pertama akan memiliki warna hitam, paragraf ke dua berwarna merah, paragraf ke tiga akan berwarna biru dan paragraf ke empat akan berwarna hijau. Mari kita lihat hasil tampilannya:



Berikut ini penjelasannya:

Style berikut akan diterapkan pada elemen <p>

```
p {
    color: black;
}
```

Sedangkan style berikut hanya akan diterapkan pada elemen <p> yang memiliki class bernama group.

```
p.group {
    color: red;
}
```

Selanjutnya untuk style berikut ini akan diterapkan pada elemen yang memiliki class bernama group.

```
.group {
    color: blue;
}
```

Perhatikan tampilan gambar di atas, bahwa untuk <p class="group"> tidak akan terpengaruh karena elemen tersebut sudah memiliki style sendiri yaitu p.group. Apabila style p.group tidak ada, maka elemen <p class="group"> akan terpengaruh style .group (text akan menjadi merah)

Suatu style juga dapat ditulis sebagai berikut

```
#group {
    color: green;
}
```

Style tersebut hanya akan mengatur tampilan pada elemen yang memiliki id bernama group.

Lebih Lanjut Tentang Value

Pada bab ini akan dijelaskan lebih lanjut tentang aturan penulisan value dari properti yang diberikan.

Terdapat beberapa jenis value yang dapat berikan pada properti style, antara lain berupa predefined value, bilangan, prosentase, url, dan juga warna.

Predefined Value

Predefined value merupakan nilai yang sudah terdefiniskan oleh (X)HTML. Nilai ini dapat dipilih dari beberapa alternatif pilihan yang ada. Contoh penggunaan predefined value ini adalah pada style berikut

```
h1 {  
    font-style : italic;  
}
```

Style di atas digunakan untuk membuat text miring pada elemen heading 1. Dalam hal ini value 'italic' merupakan predefined value. Untuk properti font-style terdapat pilihan value yang telah didefinisikan oleh (X)HTML yaitu italic, normal, dan oblique. Perhatikan bahwa predefined value ditulis tanpa diapit tanda petik.

Bilangan

Suatu value dapat pula berupa bilangan tertentu (tanpa satuan apapun). Berikut ini contohnya:

```
p {  
    line-height : 2;  
}
```

Maksud style di atas adalah mengatur jarak antar baris paragraf menjadi 2 kali dari ukuran fontnya.

Panjang, Besar dan Prosentase

Untuk menyatakan panjang, besar dan prosentase, value harus diberikan satuan seperti px (pixel), in (inch), cm (centimeter), mm (milimeter), pt (points) dan pc (picas). Berikut ini beberapa contohnya

```
h1 {
  font-size : 20px; /* membuat ukuran font 20 pixel */
}

hr {
  width : 20%;

  /* membuat garis horizontal dengan panjang 20% terhadap lebar
  jendela browser */
}
```

URL

Sebuah value juga dapat berupa URL. Berikut ini contohnya

```
body
{
  background : url(img/image.jpg);
}
```

Style di atas untuk memberi background image pada body yang diambil dari folder img dengan nama file image.jpg.

Warna

Untuk memberi value berupa warna pada properti, ada beberapa cara yang dapat dilakukan yaitu dengan menuliskan secara eksplisit warna yang diinginkan, misal (red, green, yellow, dll) atau dengan menuliskannya dalam kode hexadesimal. Berikut ini contohnya

```
p
{
  color : red;
}

p
{
  color : #FF0000; /* identik dengan warna merah (hexadesimal)
}
```

Membuat CSS Eksternal

Pada contoh-contoh sebelumnya, jenis CSS yang dibuat adalah CSS internal karena ditulis jadi satu dengan code (X)HTML nya. Perhatikan kembali contoh berikut ini

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">
p {
    color: #FFFFFF;
    font-family: arial;
    background: #FF0000;
    font-weight: bold;
}

p.group {
    color: #FF0000; /* membuat warna text putih */
    font-family: arial;
    background: #FFFFFF; /* membuat background putih */
}

p#one {
    color: blue;
    font-family: "times new roman";
}
</style>
</head>

<body>

<p>Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. </p>

<p class="group">Ini adalah paragraf 2. Ini adalah paragraf 2. Ini
adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. </p>

<p class="group" id="one">Ini adalah paragraf 3. Ini adalah paragraf 3.
Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3.
</p>

</body>
</html>
```

Anda dapat menaruh CSS yang ada dalam code di atas ke dalam file tertentu, dengan harapan file CSS itu dapat digunakan oleh halaman web yang lain. File CSS inilah yang disebut CSS eksternal yang memiliki file extension .css

Sebagai contoh, misalnya style di atas kita simpan ke dalam file style.css

style.css

```
p {
    color: #FFFFFF;
    font-family: arial;
    background: #FF0000;
    font-weight: bold;
}

p.group {
    color: #FF0000; /* membuat warna text putih */
    font-family: arial;
    background: #FFFFFF; /* membuat background putih */
}

p#one {
    color: blue;
    font-family: "times new roman";
}
```

Lantas... bagaimana cara mengintegrasikan style.css tersebut ke halaman web yang kita buat? Caranya adalah dengan memberikan perintah

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

pada bagian header halaman web.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>
<link rel="stylesheet" type="text/css" href="style.css" />
</head>

<body>
<p>Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. </p>

<p class="group">Ini adalah paragraf 2. Ini adalah paragraf 2. Ini
adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. </p>

<p class="group" id="one">Ini adalah paragraf 3. Ini adalah paragraf 3.
Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3.
</p>

</body>
</html>
```

Dengan konsep ini, Anda dapat menggunakan CSS eksternal yang berasal dari situs web lain untuk mengatur style halaman web Anda, dengan cara menambahkan URL secara lengkap pada atribut href nya.

```
<link rel="stylesheet" type="text/css" href="http://.../style.css" />
```

copyright 2007 . rosihan ari

Style Untuk Font Family

Font family digunakan untuk menentukan jenis huruf/font pada suatu tampilan teks. Berikut ini adalah nama properti dan penulisan valuenya:

```
font-family: name;
```

dengan name adalah jenis font yang Anda inginkan.

Contoh:

```
p {  
  font-family: arial;  
}
```

Anda juga dapat memberikan jenis font lebih dari satu yang dipisahkan dengan koma.

Contoh:

```
p {  
  font-family : "arial black", arial, "Times New Roman";  
}
```

Maksud dari style di atas adalah bahwa jenis font arial black akan diprioritaskan untuk digunakan. Namun apabila font tersebut tidak disupport oleh browser maka yang digunakan arial, begitu seterusnya.

Sebagai catatan, untuk jenis font yang terdiri dari lebih satu kata harus diapit dengan tanda petik ganda (double quotes).

Membuat Teks Italic

Membuat teks italic dapat pula dilakukan melalui style. Berikut ini cara penulisannya

font-style : italic;

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

p {
    font-family: arial;
    font-style: italic;
}

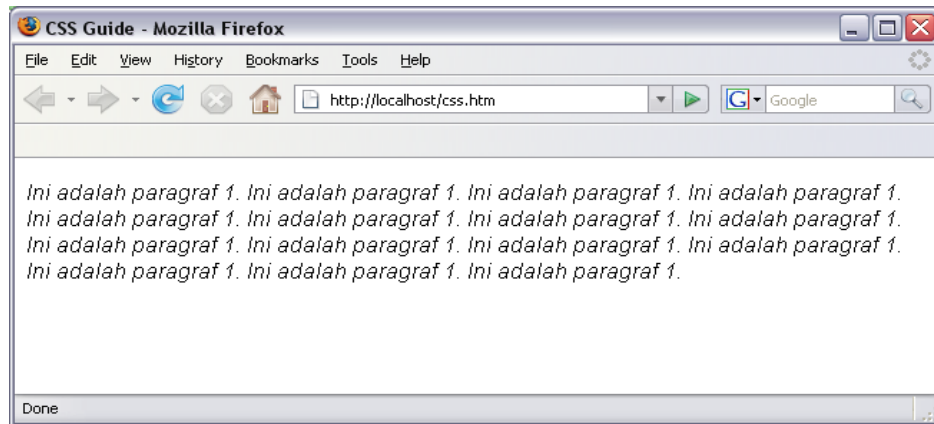
</style>
</head>

<body>

<p>Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
</p>

</body>
</html>
```

Style di atas akan menghasilkan teks paragraf yang berjenis font arial dan italic.



Membuat Teks Tebal (Bold)

Untuk membuat teks tebal atau bold, properti yang digunakan adalah

```
font-weight : bold;
```

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

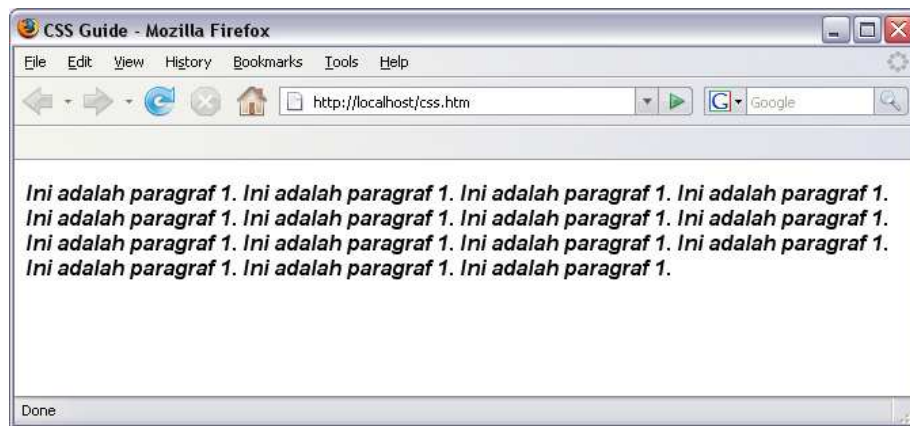
p {
    font-family: arial;
    font-style: italic;
    font-weight: bold;
}

</style>
</head>

<body>

<p>Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
</p>
</body>
</html>
```

Tampilannya diperoleh berikut ini:



Mengatur Ukuran Font

Pengaturan ukuran font dapat dilakukan dengan style yang menggunakan properti

```
font-size : size;
```

dengan size diisi angka yang menyatakan ukuran disertai satuannya.

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">
p.styled {
    font-size: 20px; /* membuat font berukuran 15 pixel */
}

</style>
</head>

<body>

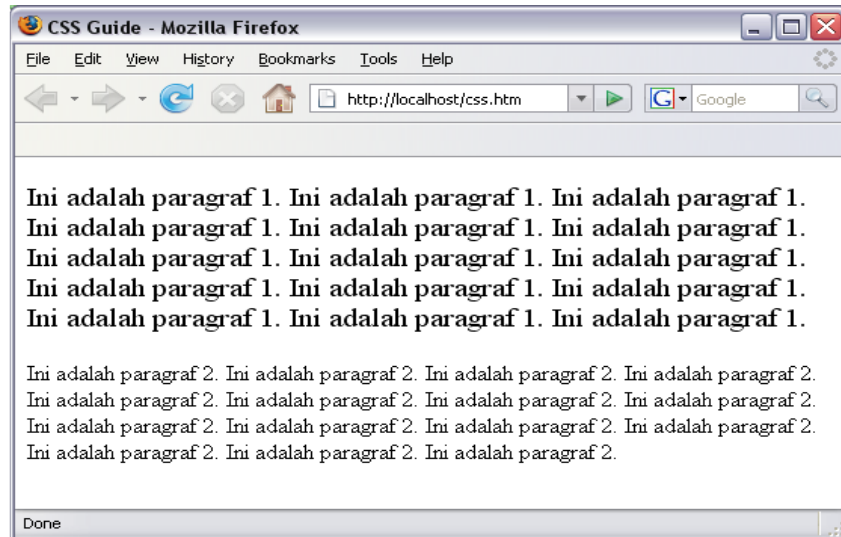
<p class="styled">Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. </p>

<p>Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
</p>

</body>
</html>
```

Pada contoh di atas, paragraf pertama berukuran 20 pixel sedangkan paragraf kedua berukuran normal (default). Sebagai catatan, teks berukuran 16px secara default.

Hasilnya tampak berikut ini



Mengatur Jarak Baris Dalam Paragraf

Anda dapat mengatur jarak baris teks dalam suatu paragraf dengan menggunakan CSS. Berikut ini adalah nama properti nya

line-height : size;

dengan size diisi angka yang menyatakan ukuran disertai satuannya.

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">
p.styled {
    line-height: 30px;
}

</style>
</head>

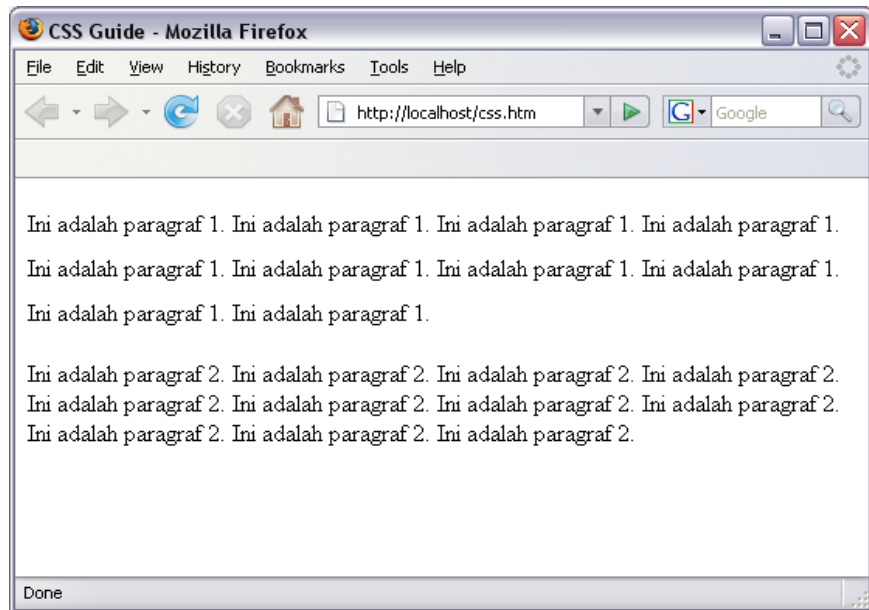
<body>

<p class="styled">Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. </p>

<p>Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. </p>

</body>
</html>
```

Paragraf pertama diberikan style untuk mengatur jarak antar baris sebesar 30 pixel, sedangkan paragraf kedua memiliki jarak baris default. Perhatikan perbedaannya pada tampilan berikut ini



Mengubah Warna Text

Untuk mengubah warna font, propertinya adalah:

color : warna;

dengan value warna diisikan dengan warna yang diinginkan, dapat berupa kode hexadesimalnya, maupun nama eksplisitnya.

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

p.styled1 {
    font-family: "Palatino Linotype";
    font-size: 20px;
    color: #FF0000;
}

p.styled2 {
    color: green;
}

</style>
</head>

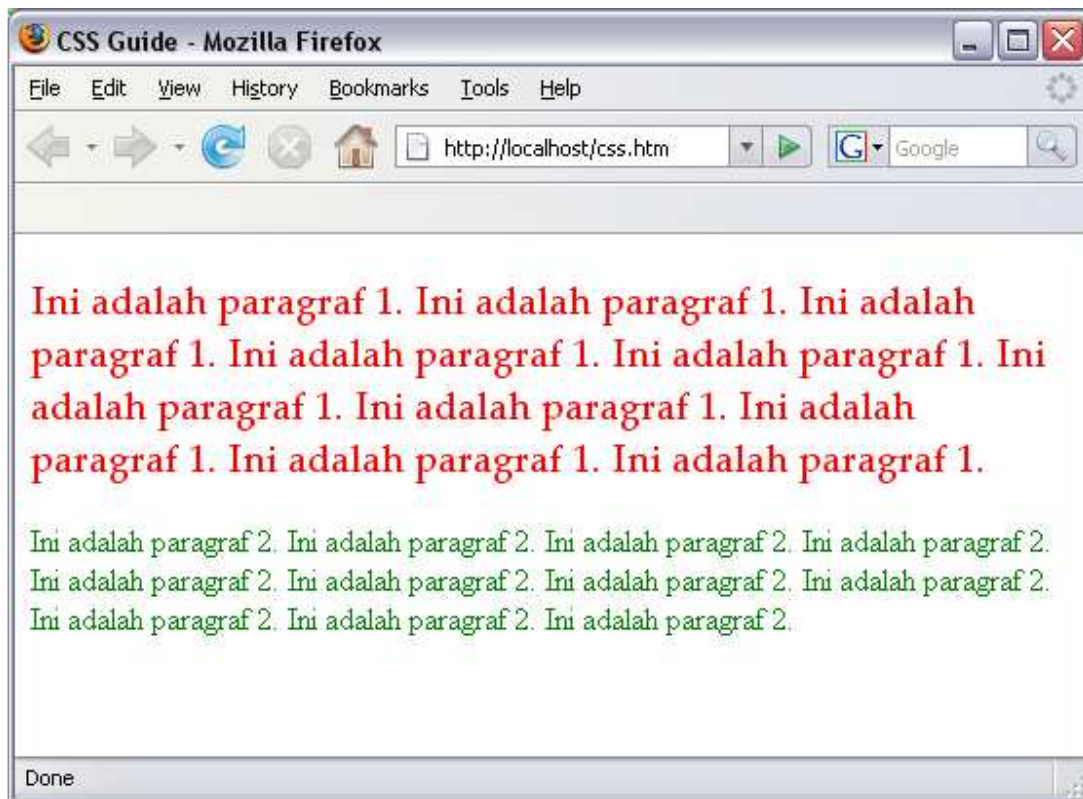
<body>

<p class="styled1">Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. </p>

<p class="styled2">Ini adalah paragraf 2. Ini adalah paragraf 2. Ini
adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. Ini
adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. Ini
adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. </p>

</body>
</html>
```

Paragraf pertama pada contoh di atas akan diberikan style text berwarna merah (#FF0000) sedangkan paragraf kedua berwarna hijau (green).



Mengubah Warna Background Text

Apabila Anda ingin mengubah warna background text, maka berikut ini adalah propertinya:

background : warna;

cara penulisan value dari warna sama dengan color, yaitu dapat berupa nama eksplisit warna yang diinginkan maupun kode hexadesimalnya.

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

body {
    background : green;
}

p.styled {
    font-family: "Palatino Linotype";
    font-size: 20px;
    color: #FF0000;
    background: #FFFFFF; /* warna putih */
}

p {
    color: orange;
    font-weight: bold;
    background: #000000; /* warna hitam */
}

</style>
</head>

<body>

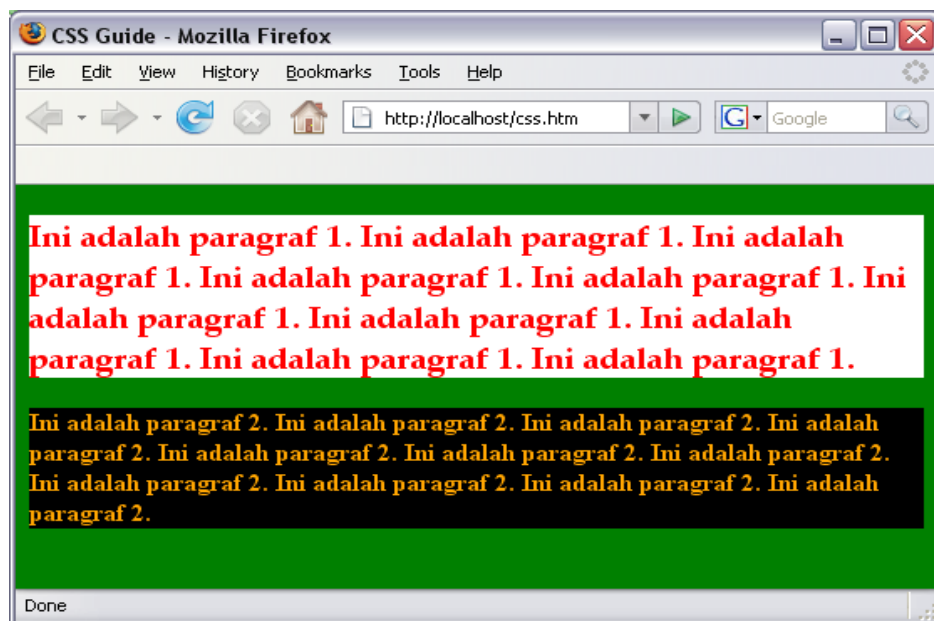
<p class="styled">Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini
adalah paragraf 1. Ini adalah paragraf 1. </p>

<p>Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. </p>
```

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

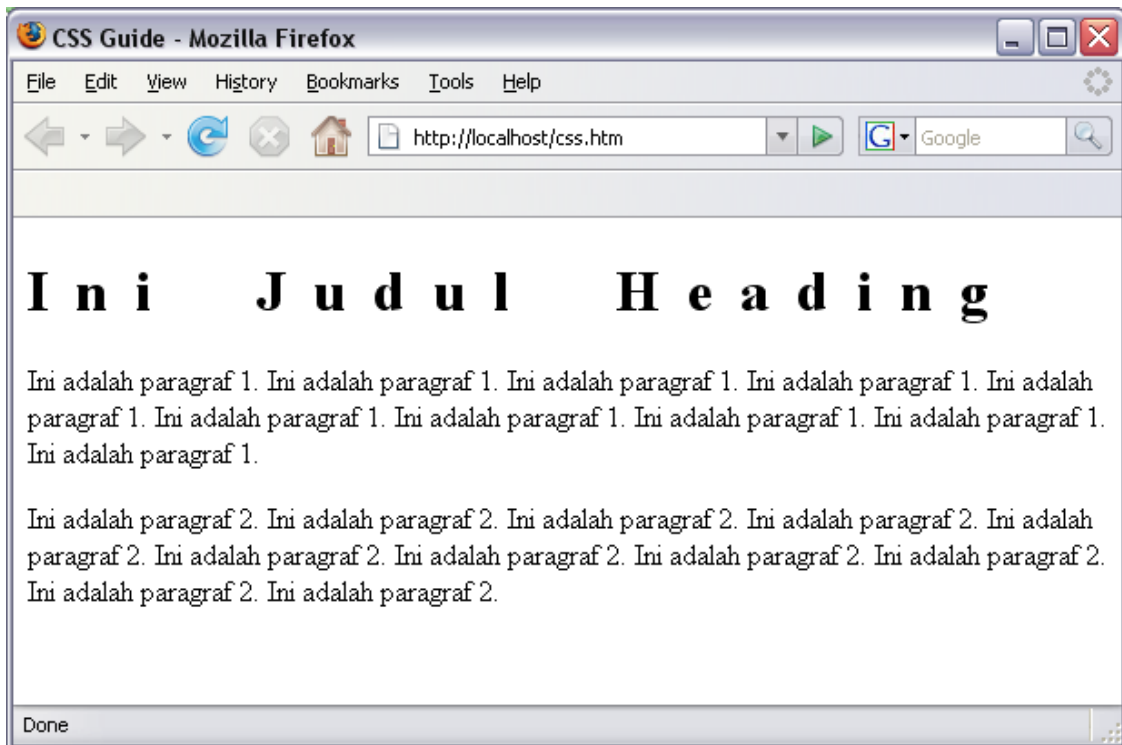
Pada contoh di atas, bagian body diberi warna background (seluruh halaman), namun perkecualian untuk bagian paragraf. Paragraf pertama diberi warna background putih sedangkan paragraf kedua diberi warna hitam.

Berikut ini tampilan di browser:



Mengatur Jarak Spasi Antar Text

Perhatikan contoh tampilan berikut ini:



Pada judul heading terlihat bahwa antar kata dan huruf terdapat jarak tertentu. Jarak tersebut diatur dengan menggunakan properti sbb:

```
word-spacing : jarak;  
letter-spacing : jarak;
```

Properti pertama di atas digunakan untuk mengatur jarak antar kata, sedangkan yang kedua untuk mengatur jarak antar huruf.

Sebagai contoh, berikut ini code yang menghasilkan tampilan di atas.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<title>CSS Guide</title>  
  
<style type="text/css">
```



```
h1 {  
    word-spacing: 20px;  
    letter-spacing: 15px;  
}  
  
</style>  
</head>  
  
<body>  
  
<h1>Ini Judul Heading</h1>  
  
<p>Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.  
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.  
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.  
Ini adalah paragraf 1. </p>  
  
<p>Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.  
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.  
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.  
Ini adalah paragraf 2. Ini adalah paragraf 2. </p>  
  
</body>  
</html>
```

Membuat Text Indent

Pada bagian awal paragraf biasanya terdapat text indent atau teks yang agak menjorok ke dalam paragraf dalam jarak tertentu. Berikut ini adalah properti yang digunakan untuk mengaturnya:

text-indent : jarak;

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

p {
    text-indent : 40px;
}

</style>
</head>

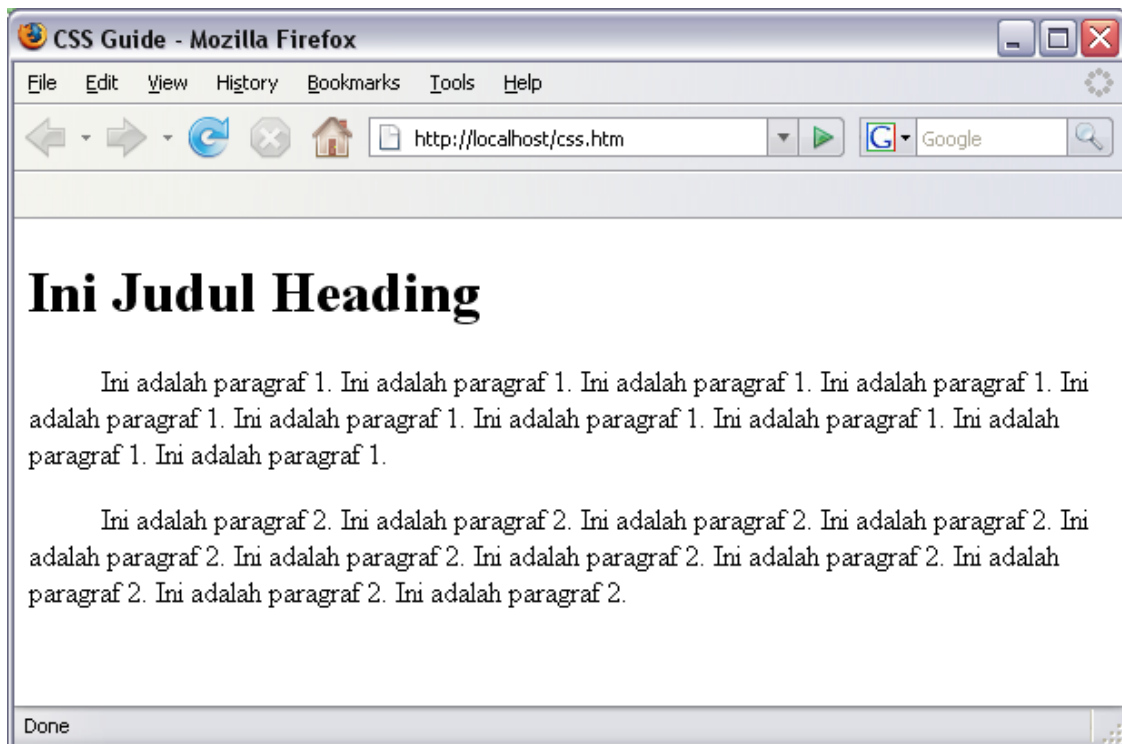
<body>

<h1>Ini Judul Heading</h1>

<p>Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1.
Ini adalah paragraf 1. </p>

<p>Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2.
Ini adalah paragraf 2. Ini adalah paragraf 2. </p>

</body>
</html>
```



Dalam contoh di atas, awal paragraf dibuat text indent berjarak 40 pixel.

Perataan Teks (Text Alignment)

Anda dapat mengatur perataan teks dengan menggunakan properti

`text-align` : jenis;

dengan value jenis dapat digantikan dengan left, center, right dan justify.

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

#p1 {
    text-align : center;
}

#p2 {
    text-align : justify;
}

#p3 {
    text-align : right;
}

</style>
</head>

<body>

<h1>Ini Judul Heading</h1>

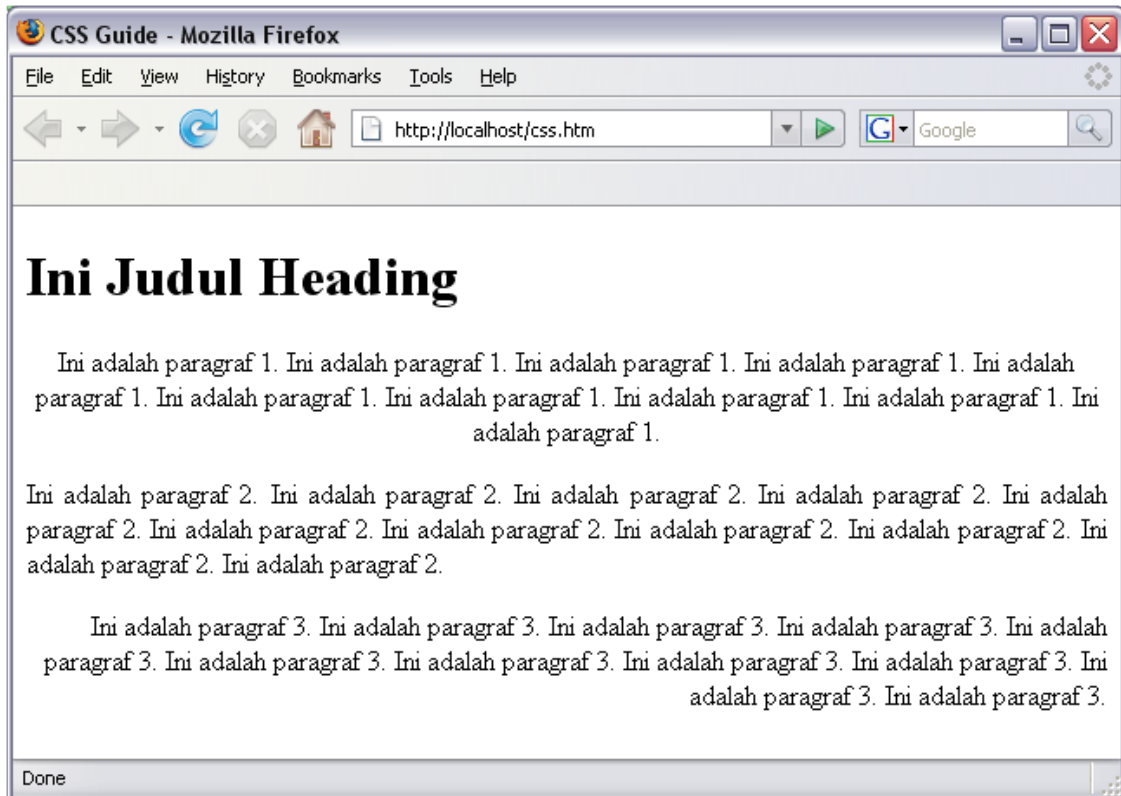
<p id="p1">Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah
paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah
paragraf 1. Ini adalah paragraf 1. Ini adalah paragraf 1. Ini adalah
paragraf 1. </p>

<p id="p2">Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah
paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah
paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. Ini adalah
paragraf 2. Ini adalah paragraf 2. Ini adalah paragraf 2. </p>

<p id="p3">Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah
paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah
```

paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3. Ini adalah paragraf 3. </p>

</body>
</html>



Jenis perataan yang digunakan pada contoh di atas adalah rata tengah untuk paragraf pertama, rata kanan kiri (justify) untuk paragraf kedua dan rata kanan untuk paragraf ketiga.

Mengubah Text Case

Dengan CSS, Anda dapat pula mengubah suatu teks menjadi huruf kecil semua (lowercase), besar semua (uppercase), atau capitalize (hanya huruf pertama dari kata yang dibuat besar). Hal ini dilakukan dengan menggunakan properti

text-transform : option;

dengan option dapat diganti dengan lowercase, uppercase, atau capitalize.

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

#heading1 {
    text-transform : uppercase;
}

#heading2 {
    text-transform : lowercase;
}

#heading3 {
    text-transform : capitalize;
}

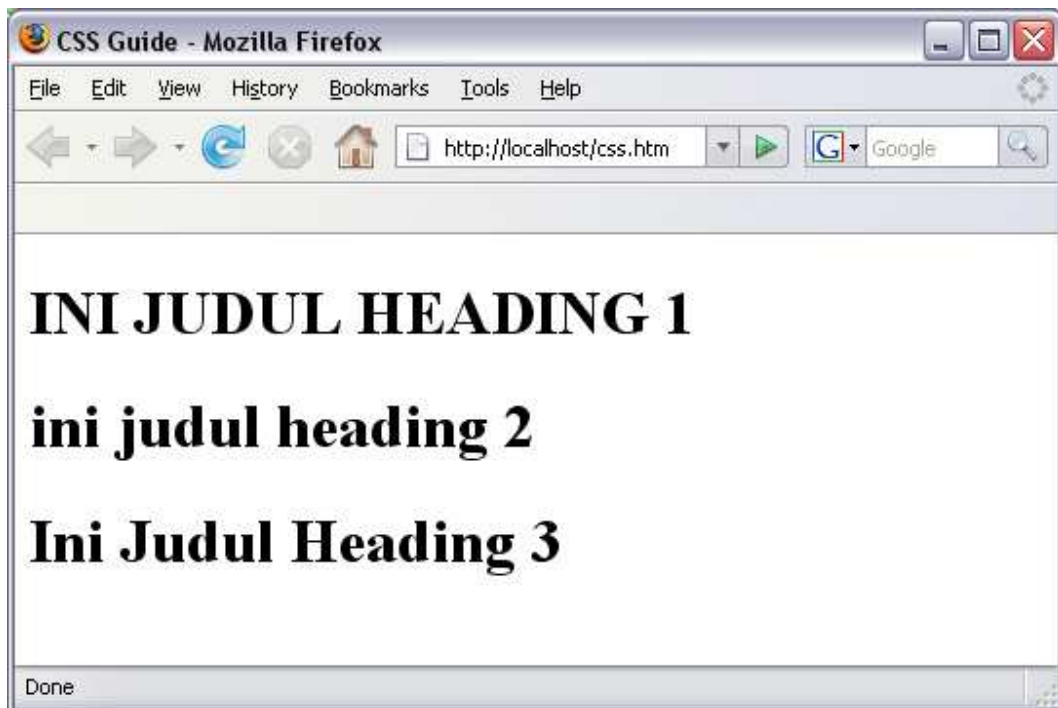
</style>
</head>

<body>

<h1 id="heading1">Ini Judul Heading 1</h1>
<h1 id="heading2">Ini Judul Heading 2</h1>
<h1 id="heading3">Ini Judul Heading 3</h1>

</body>
</html>
```

Hasil tampilan code di atas dapat dilihat pada gambar berikut.



Membuat Drop Cap

Ingin mempercantik tampilan paragraf dengan menambahkan drop cap pada huruf pertamanya? Berikut ini contoh penulisan style untuk membuat hal tersebut.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

p.styleku:first-letter {
    float: left;
    font-family: "Lucida Grande";
    font-size: 80px;
}

p {
    text-align : justify;
}

</style>
</head>

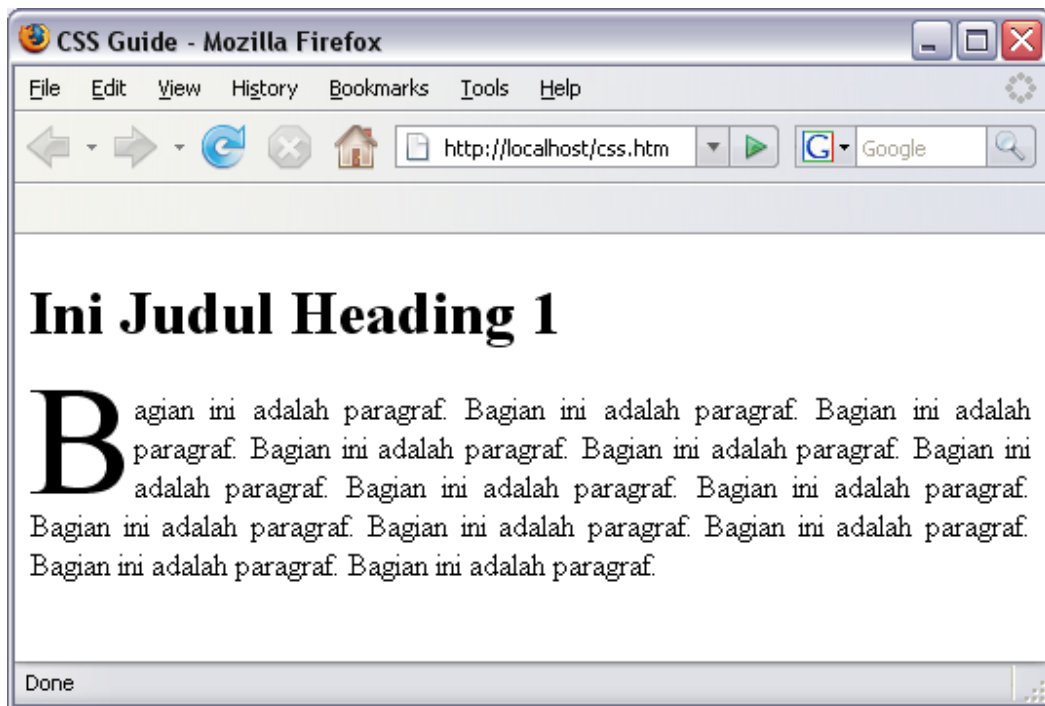
<body>

<h1 id="heading1">Ini Judul Heading 1</h1>

<p class="styleku">Bagian ini adalah paragraf. Bagian ini adalah
paragraf. Bagian ini adalah paragraf. Bagian ini adalah paragraf.
Bagian ini adalah paragraf. Bagian ini adalah paragraf. Bagian ini
adalah paragraf. Bagian ini adalah paragraf. Bagian ini adalah paragraf.
Bagian ini adalah paragraf. Bagian ini adalah paragraf. Bagian ini adalah paragraf.
Bagian ini adalah paragraf. Bagian ini adalah paragraf. </p>

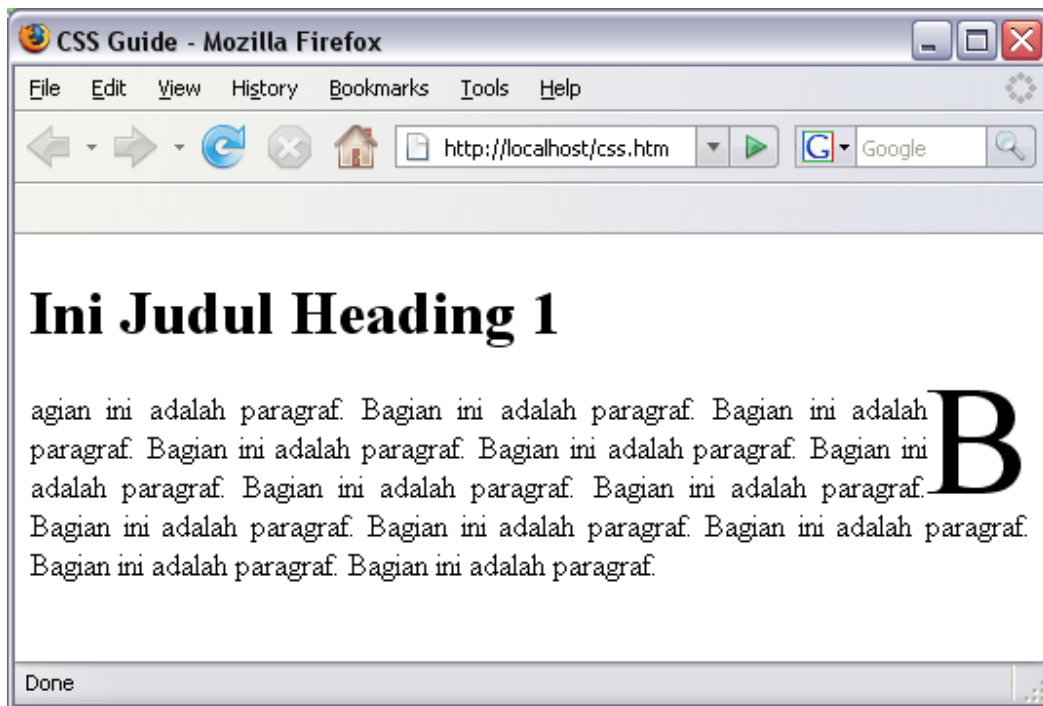
</body>
</html>
```

Adapun tampilan dari code di atas adalah sebagai berikut.



```
p.styleku:first-letter {  
    float: left;  
    font-family: "Lucida Grande";  
    font-size: 80px;  
}
```

Code style di atas hanya akan berlaku untuk huruf pertama dari paragraf yang memiliki class styleku, dalam hal ini adalah huruf B. Sedangkan properti float digunakan untuk mengatur posisi huruf B tersebut, apakah di sebelah kiri atau kanan, sehingga valuenya adalah left. Apabila value yang dipilih adalah right, maka berikut ini tampilannya:



sungguh jelek sekali bukan ☺

Membuat Text Decoration

Anda dapat membuat text decoration seperti underline, overline, line-through, maupun blink dengan properti

text-decoration : option;

dengan option dapat diganti dengan underliner, overline, line-through atau blink. Untuk melihat perbedaannya, perhatikan contoh berikut.

Contoh:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>CSS Guide</title>

<style type="text/css">

#heading1 {
    text-decoration : blink;    /* membuat teks berkelap kelip */
}

#heading2 {
    text-decoration : line-through;
    /* membuat garis ditengah text/ dicoret */
}

#heading3 {
    text-decoration : underline; /* membuat garis bawah */
}

#heading4 {
    text-decoration : overline; /* membuat garis di atas text */
}

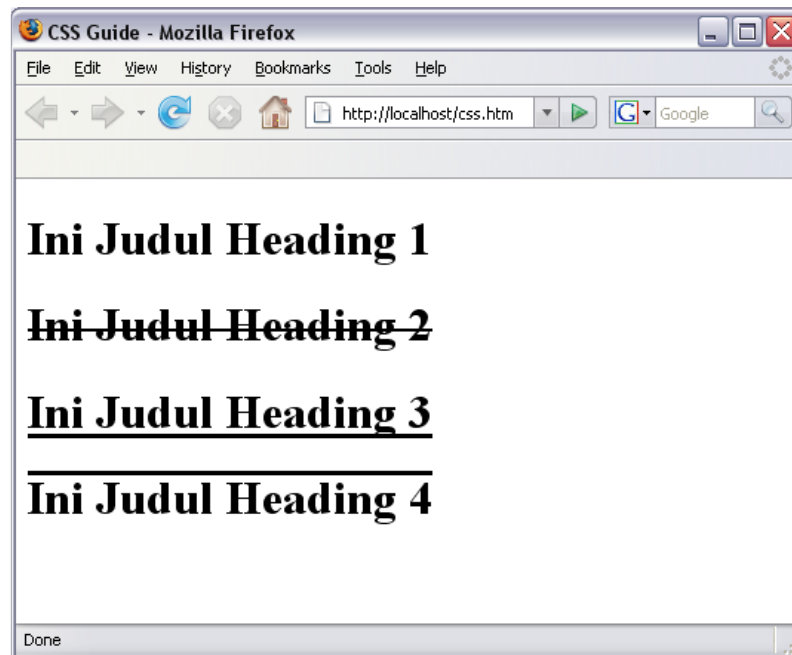
</style>
</head>

<body>

<h1 id="heading1">Ini Judul Heading 1</h1>
<h1 id="heading2">Ini Judul Heading 2</h1>
<h1 id="heading3">Ini Judul Heading 3</h1>
<h1 id="heading4">Ini Judul Heading 4</h1>

</body>
</html>
```

Tampilan dari code di atas adalah sebagai berikut, akan tetapi untuk efek blink pada heading 1 tidak kelihatan ketika di screen shot.

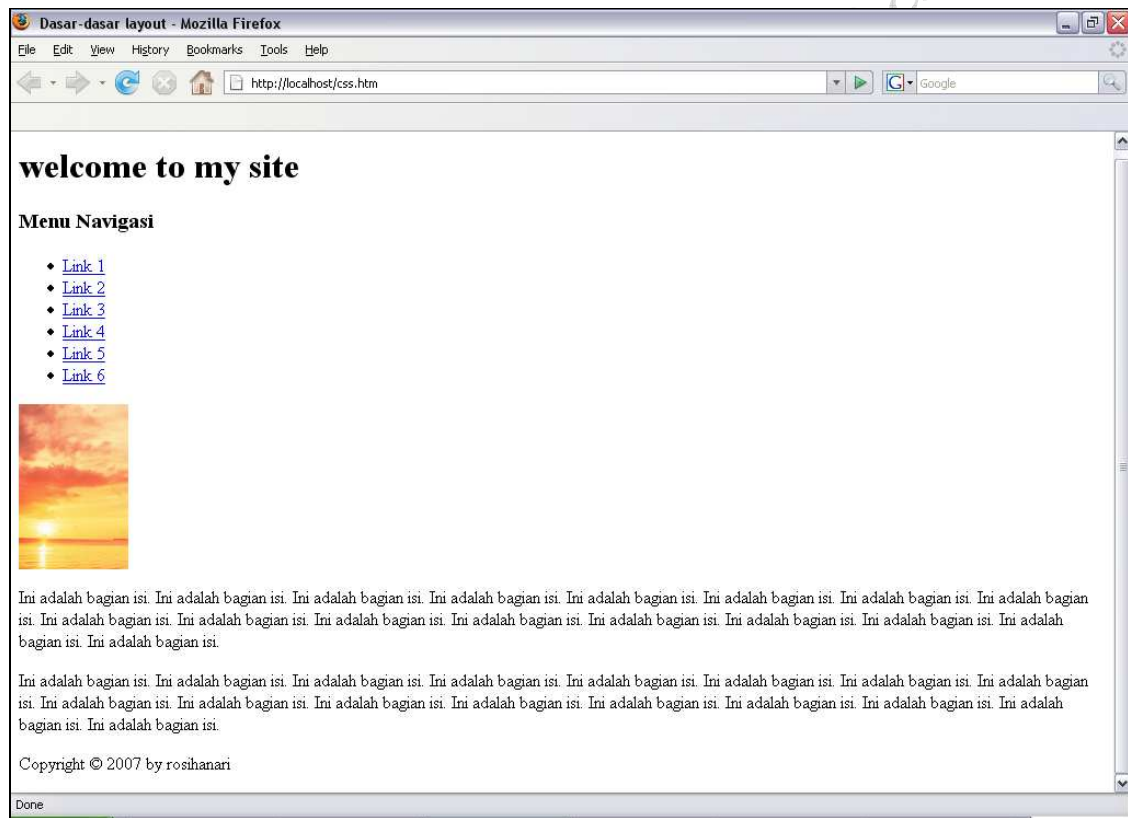



```
        Ini adalah bagian isi. Ini adalah bagian isi. Ini adalah
        bagian isi. Ini adalah bagian isi. Ini adalah bagian isi.
        Ini adalah bagian isi. Ini adalah bagian isi.
    </p>
</div>

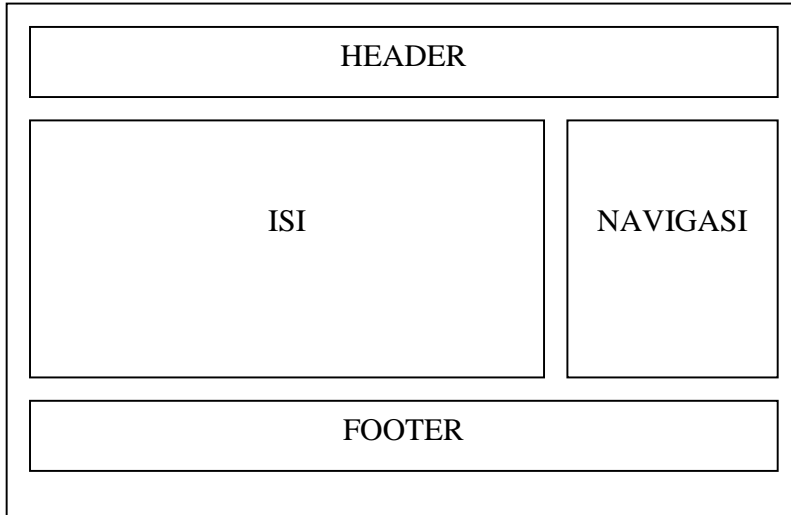
<div id="footer">
    <p>Copyright &copy; 2007 by rosihanari</p>
</div>

</body>
</html>
```

Tampilan code di atas tampak pada gambar berikut ini.



Pada contoh ini, kita akan membuat layout sbb:



Perhatikan desain di atas, terdapat 4 bagian sel yang akan kita buat. Sebaiknya untuk nama id pada masing-masing division diberi nama sesuai desain supaya mudah dipahami.

Untuk yang pertama, kita akan membuat sel untuk bagian header. Berikut ini style untuk bagian header yang saya buat.

```
#header {  
    padding:10px;           /* jarak teks dari tepi kotak */  
    border:1px solid #000000; /* membuat garis tepi 1 pixel */  
    background-color:#CCCCC; /* membuat warna latar */  
}
```

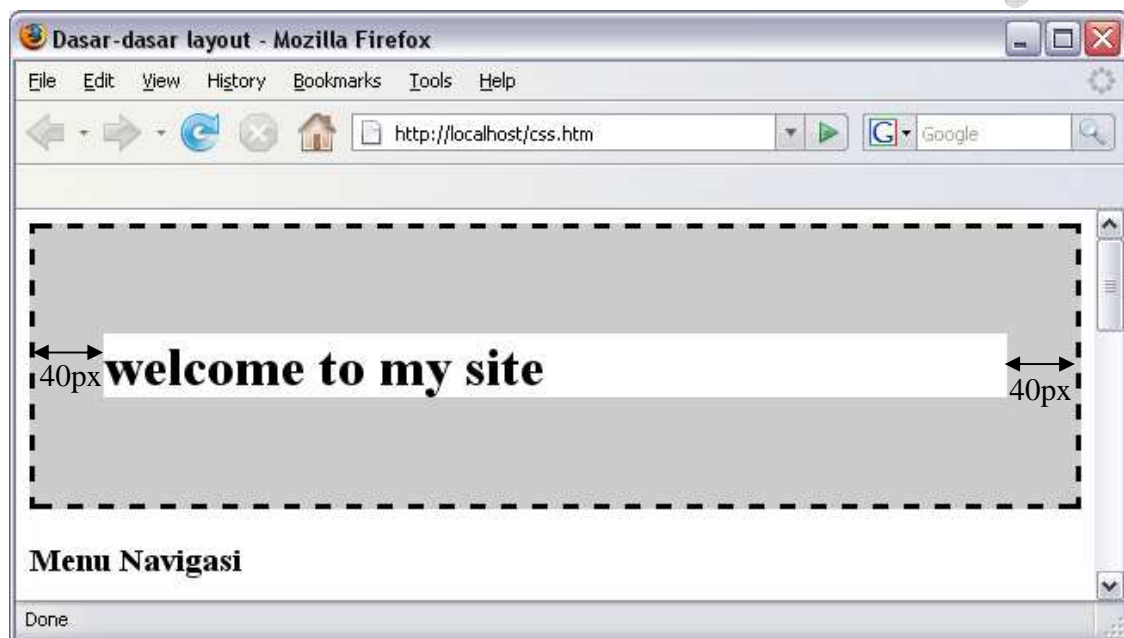
Dengan style di atas, tampilan header sudah agak lebih bagus (lihat tampilan berikut)



Properti padding menyatakan jarak antara tepi sel dengan elemen yang ada di dalamnya. Berikut contoh style lain dengan padding nya 40 pixel dan juga tipe garis bordernya putus-putus (dashed).

```
#header {  
    padding:40px;  
    border:3px dashed #000000; /* garis tepi putus-putus */  
    background-color:#CCCCCC;  
}
```

Hasilnya adalah:



Kita kembalikan ke style yang header awal saja ya, karena contoh di atas hanya untuk menjelaskan penggunaan padding dan dashed. Style header yang kita gunakan adalah

```
#header {  
    padding:10px;  
    border:1px solid #000000;  
    background-color:#CCCCCC;  
}
```

Sedangkan style untuk footernya dapat ditulis sebagai berikut:

```
#footer {  
    padding:10px;  
    border:1px solid #000000;  
    background-color:#CCCCCC;  
}
```


Hasil tampilannya adalah :



Selanjutnya kita buat style untuk bagian isi.

```
#isi {  
    padding:10px;  
    border:1px solid #000;  
}
```



Untuk bagian navigasi, karena bagian tersebut ingin diletakkan di sebelah kanan, maka berikan properti `float: right`. Perhatikan style berikut

```
#navigasi {  
    float:right;  
    padding:10px;  
    border:1px solid #000;  
    background: yellow;  
}
```

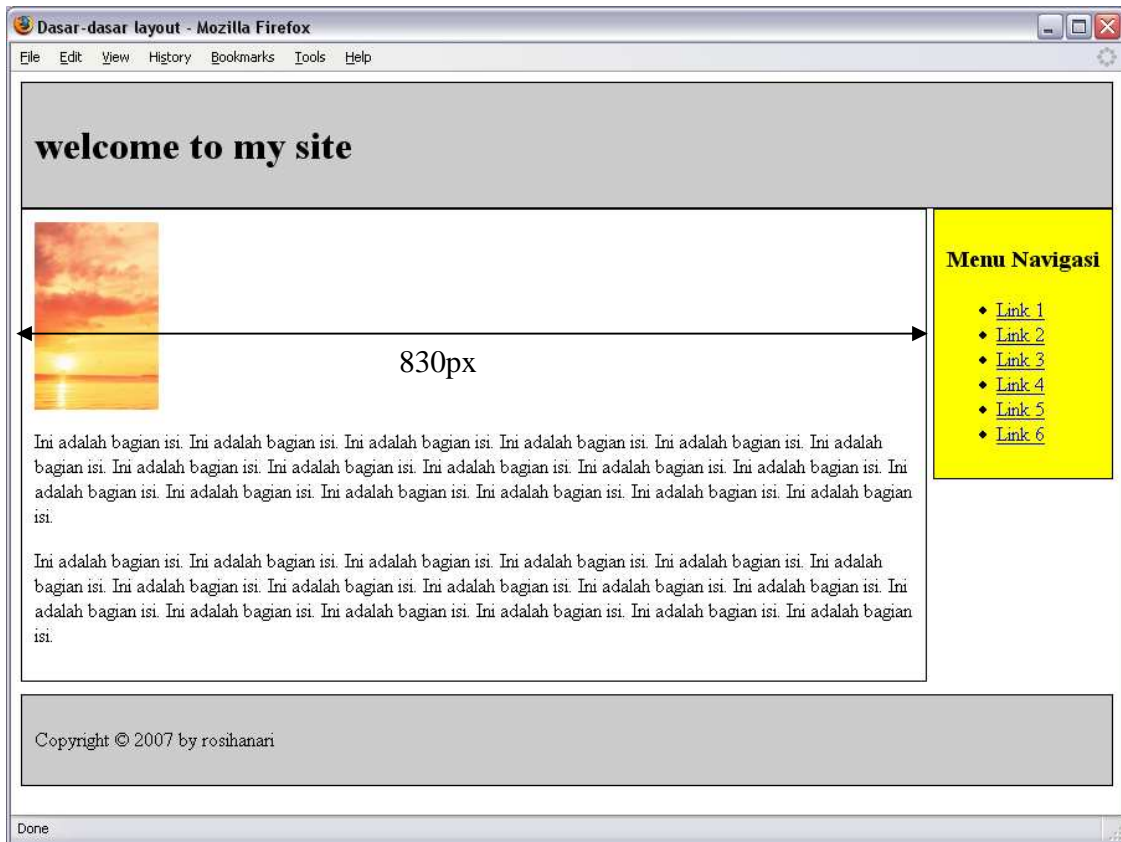
Efek penggunaan properti `float: right` dapat dilihat pada tampilan berikut ini.



Meskipun sudah diletakkan di sebelah kanan, ternyata bagian isi masih belum seperti yang diinginkan sesuai desain layout. Oleh karena itu kita atur lebar sel untuk bagian isi dengan properti width dan juga kita beri jarak sedikit antara sel isi dan footer dengan properti margin-bottom. Perhatikan style berikut:

```
#isi {
    width: 830px;                /* lebar elemen */
    margin-bottom:10px;
    padding:10px;
    border:1px solid #000;
}
```

Efek modifikasi style pada bagian isi dapat dilihat pada gambar berikut ini.

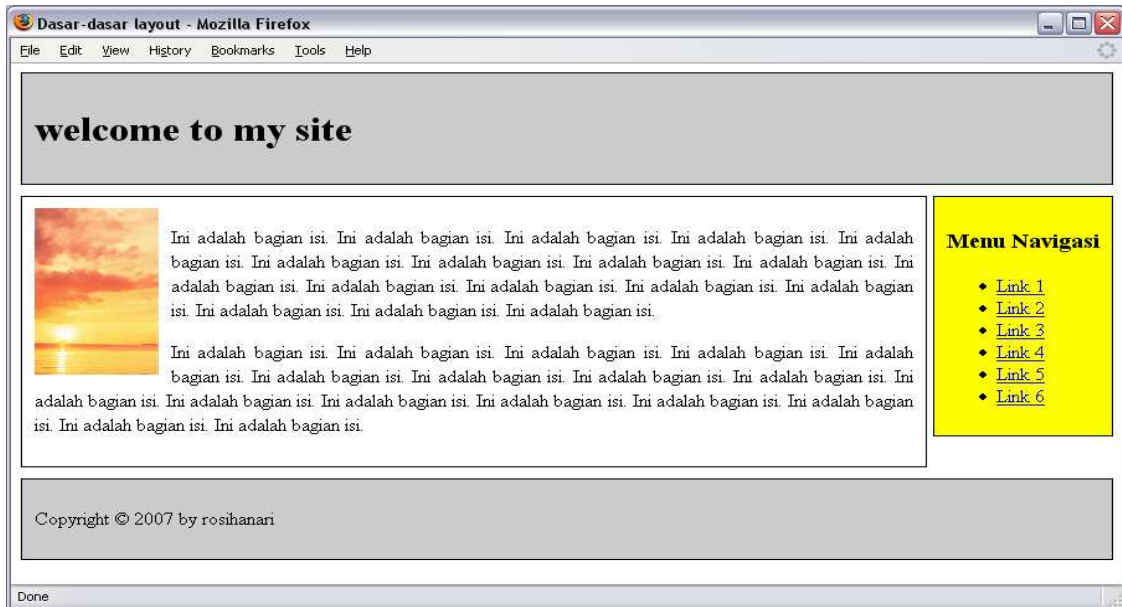


Pada gambar di atas tampak, bahwa bagian isi sudah memiliki sel sendiri selebar 830 pixel dan ada sedikit jarak (10 pixel) dari footer.

Ada baiknya kita tambahkan sedikit jarak antar bagian navigasi dengan header, begitu pula dengan bagian isi. Untuk keperluan ini, gunakan properti `margin-top`. Supaya seimbang dengan yang lain, kita beri jarak 10 pixel.

```
#navigasi {  
    float:right;  
    padding:10px;  
    margin-top: 10px;  
    border:1px solid #000;  
    background: yellow;  
}  
  
#isi {  
    width: 810px;  
    margin-top: 10px;  
    margin-bottom:10px;  
    padding:10px;  
    border:1px solid #000;  
}
```


Properti line-height digunakan untuk mengatur jarak antar baris teks paragraf. Perhatikan efek penggunaan style tersebut dalam gambar berikut



Lho, kok gambarnya akan menyembul ke atas terhadap teks paragrafnya? OK kita modifikasi lagi style gambarnya. Kita ingin supaya paragrafnya dinaikkan sedikit supaya sejajar dengan image.

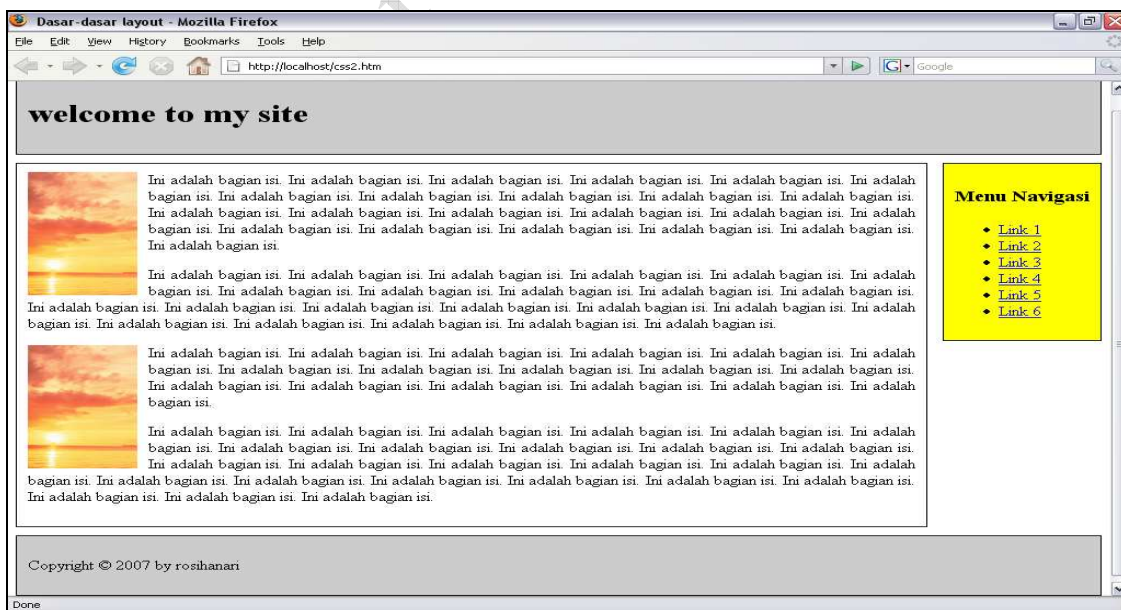
```
#isi p {  
    margin-top: 0px;  
    text-align: justify;  
    line-height: 20px;  
}
```

Hasilnya dapat dilihat pada tampilan berikut.



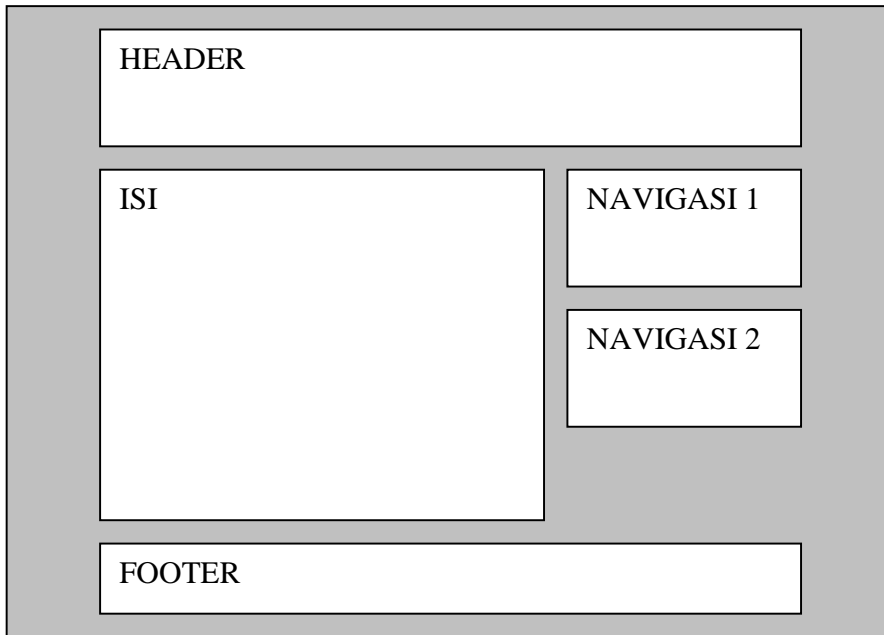
OK... that's better. ☺

Tampilan gambar berikut ini menunjukkan manfaat penggunaan class pada bagian paragraf dan image apabila terdapat beberapa image dan paragraf dalam bagian isi. Gambar menunjukkan bahwa setiap image dan paragraf akan memiliki style yang sama.



Membuat Layout dengan CSS (Sample 2)

Masih tentang bagaimana membuat layout dengan CSS, kita lanjutkan untuk membuat layout yang lain. Berikut ini adalah desain layout yang akan kita buat



Bagian background (warna abu-abu) akan kita beri image.

Langkah pertama, seperti biasa kita harus membuat bagian-bagian elemen dahulu menggunakan division (div). Berikut ini adalah code dasar untuk setiap elemennya.

```
<body>
<div id="content">
  <div id="header">
    <h1>welcome to my site</h1>
  </div>

  <div id="menu">
    <div class="navigasi">
      <h3>Menu Navigasi A</h3>
      <ul>
        <li><a href="link1.htm">Link 1</a></li>
        <li><a href="link2.htm">Link 2</a></li>
        <li><a href="link3.htm">Link 3</a></li>
        <li><a href="link4.htm">Link 4</a></li>
      </ul>
    </div>
  </div>
</div>
```

```
</div>

<div class="navigasi">
<h3>Menu Navigasi B</h3>
<ul>
  <li><a href="link1.htm">Link 1</a></li>
  <li><a href="link2.htm">Link 2</a></li>
  <li><a href="link3.htm">Link 3</a></li>
  <li><a href="link4.htm">Link 4</a></li>
</ul>
</div>

</div>

<div id="isi">

<p>Ini adalah bagian isi. Ini adalah bagian isi. Ini adalah
  bagian isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini
  adalah bagian isi. Ini adalah bagian isi. Ini adalah bagian
  isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini adalah
  bagian isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini
  adalah bagian isi. Ini adalah bagian isi. Ini adalah bagian
  isi. Ini adalah bagian isi.
</p>

<p>Ini adalah bagian isi. Ini adalah bagian isi. Ini adalah
  bagian isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini
  adalah bagian isi. Ini adalah bagian isi. Ini adalah bagian
  isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini adalah
  bagian isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini
  adalah bagian isi. Ini adalah bagian isi. Ini adalah bagian
  isi. Ini adalah bagian isi.
</p>

<p>Ini adalah bagian isi. Ini adalah bagian isi. Ini adalah
  bagian isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini
  adalah bagian isi. Ini adalah bagian isi. Ini adalah bagian
  isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini adalah
  bagian isi. Ini adalah bagian isi. Ini adalah bagian isi. Ini
  adalah bagian isi. Ini adalah bagian isi. Ini adalah bagian
  isi. Ini adalah bagian isi.
</p>
</div>

<div id="footer">
  <p>Copyright &copy; 2007 by rosihanari</p>
</div>
</div>
</body>
```

Seperti yang kita lihat pada code di atas, bahwa semua sel (header, isi, footer dan navigasi) merupakan bagian dari bagian content. Mengapa dimasukkan ke dalam bagian

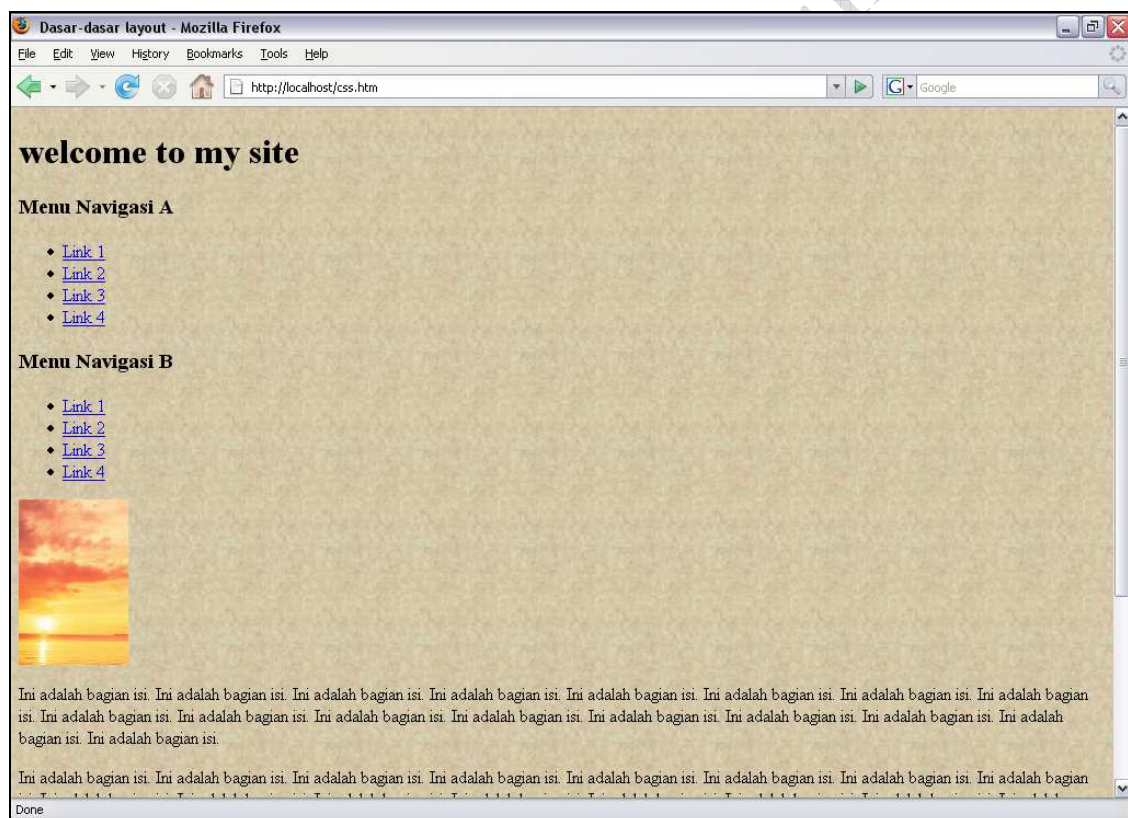
content? Hal ini dikarenakan kita akan membuat style untuk bagian tersebut, misalnya meletakkan di tengah halaman dan juga mengatur lebar bagian content nya.

OK... let's get to the point...

Kita akan membuat style untuk body nya terlebih dahulu. Karena kita akan menambahkan efek background image pada seluruh halaman, maka berikut ini adalah stylenya

```
body {
    background: url(background.jpg);
}
```

dan hasilnya adalah:



Selanjutnya kita akan membuat style untuk bagian content dengan mengatur lebarnya adalah 750 pixel dan berada di tengah halaman.

Adapun efek tampilan dari style di atas adalah berikut ini:



Apabila Anda merasa space antara text heading dengan tepi blok bawah dan atas terlalu lebar, dapat menggunakan line-height untuk mengaturnya. Semakin kecil nilai line-height maka akan semakin kecil spacenya.

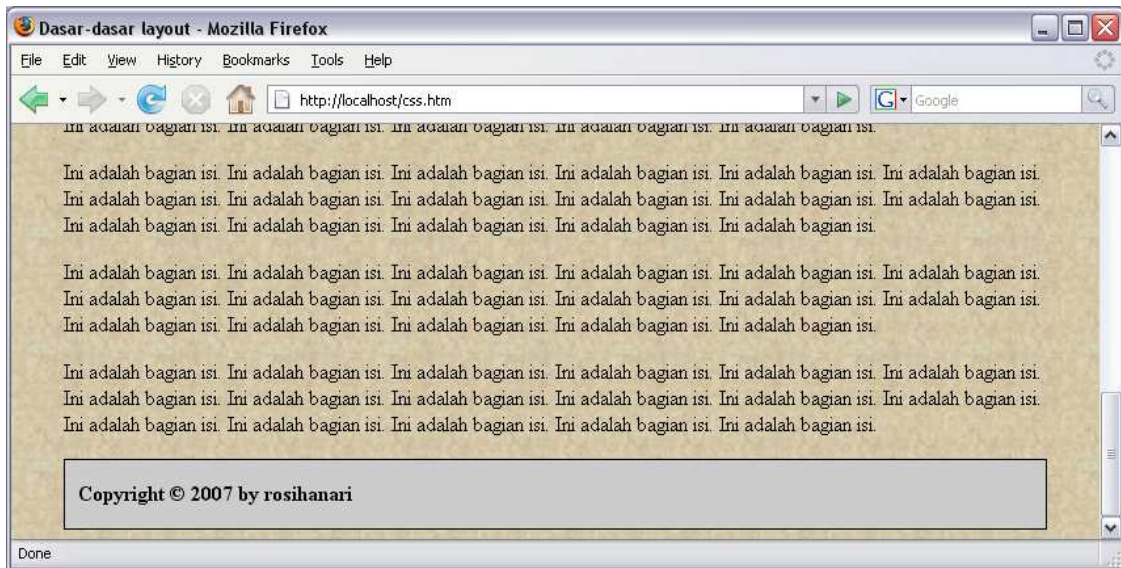
```
h1 {  
    color: #FFFFFF;  
    font-size: 40px;  
    line-height: 5px;  
}
```

Berikut ini perubahannya:



Selanjutnya kita akan membuat style untuk bagian footer beserta teks pada footernya.

```
#footer {  
    padding: 10px;  
    border: 1px solid #000000;  
    background-color: #CCCCCC;  
}  
  
#footer p {  
    text-align: justify;  
    font-weight: bold;  
}
```

Berikutnya, untuk bagian navigasi, bisa kita buat stylenya seperti di bawah ini.

```
#menu {  
    float:right;  
}
```

Sedangkan untuk setiap class navigasi, kita buat style sebagai berikut:

```
.navigasi {  
    border:1px dashed #000;  
    margin-top: 10px;  
    padding: 10px;  
    background: yellow;  
}
```

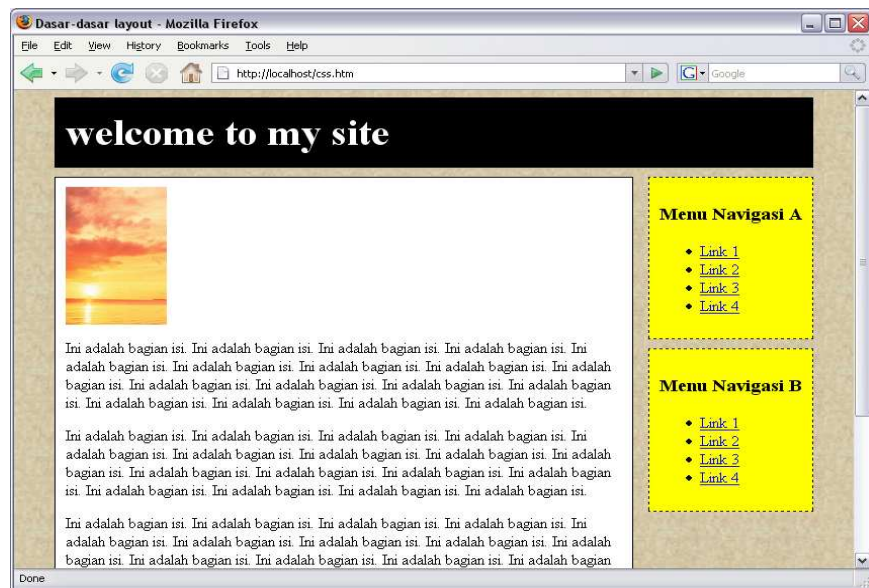
Perhatikan efek perubahan dari penggunaan style di atas pada gambar berikut ini.



Sekarang saatnya membuat style untuk bagian isi.

```
#isi {  
    width:550px;  
    margin-top: 10px;  
    margin-bottom:10px;  
    padding:10px;  
    border:1px solid #000;  
    background: white;  
}
```

Lihat hasil style pada bagian isi pada gambar berikut ini.



Hmmm... sudah mulai kelihatan cantik ☺

Berikutnya style untuk image

```
.gambar {  
    float: left;  
    margin-top: 0px;  
    margin-right: 10px;  
    margin-bottom: 20px;  
    margin-left: 0px;  
    border: 1px solid #000000;  
}
```




Gambar di atas adalah hasil menggambarkan posisi image setelah diberi efek style.

Sedangkan berikut ini adalah style untuk mengatur paragraf dalam bagian isi.

```
#isi p {
    text-align: justify;
    line-height: 18px;
    margin-top: 0px;
}
```

OK.... Selesai sudah kita membuat layout dengan CSS. Sebuah layout tanpa menggunakan tabel berhasil kita buat. Perhatikan hasil akhir layout kita pada gambar berikut.



Anda dapat berkreasi sendiri untuk membuat layout yang lain, misalnya layout yang memiliki kolom 3 buah atau yang lain. Jangan lupa untuk tetap menggunakan CSS ☺

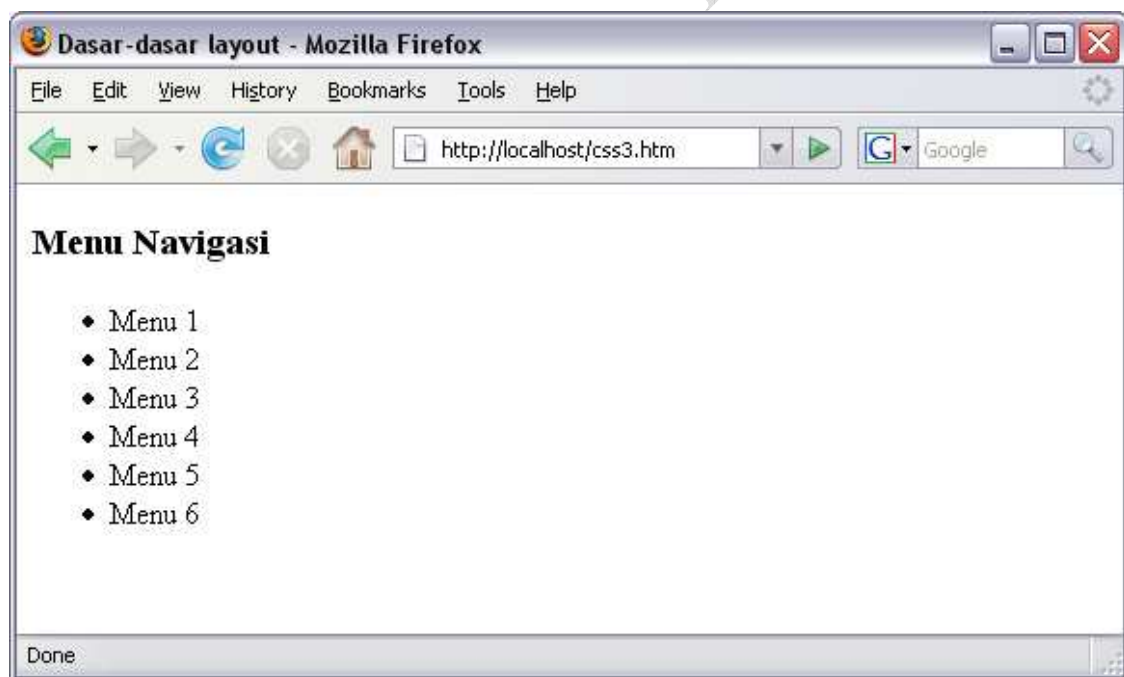
Mengatur Style List

Dengan CSS, Anda dapat pula mengatur style suatu list seperti mengubah marker maupun indentasinya, background dll.

Sebagai contoh, perhatikan code list berikut ini:

```
<h3>Menu Navigasi</h3>
<ul>
  <li>Menu 1</li>
  <li>Menu 2</li>
  <li>Menu 3</li>
  <li>Menu 4</li>
  <li>Menu 5</li>
  <li>Menu 6</li>
</ul>
</div>
```

List di atas bertipe unordered list dan secara default akan memiliki tampilan sebagai berikut:



Anda dapat mengubah style marker setiap elemen list dengan menggunakan properti berikut ini:

```
list-style-type: marker
```

dengan marker dapat diganti dengan disc, circle, square, decimal (1, 2, 3, ...), upper-alpha (A, B, C, ...), lower-alpha (a, b, c, ...), upper-roman (I, II, III, IV, ...), atau lower-roman (i, ii, iii, iv, ...)

Sebagai contoh, kita akan mengubah markernya dengan circle.

```
ul {  
    list-style-type: circle;  
}
```

dan hasilnya adalah sebagai berikut



Sedangkan untuk marker bertipe square, style adalah

```
ul {  
    list-style-type: square;  
}
```

dengan hasil yang diperoleh adalah sebagai berikut



Anda mungkin bosan dengan model marker yang itu-itu saja. Marker dapat diganti dengan image yang kita inginkan. Sebaiknya gunakan image berformat gif dan berukuran kecil.

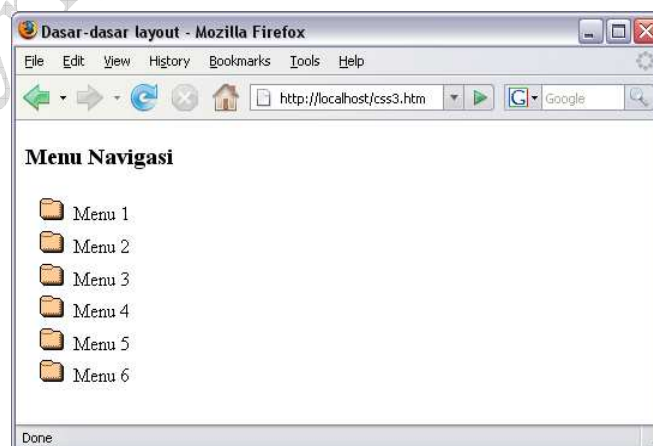
Berikut ini propertinya:

```
list-style-image: url(image.gif)
```

Sebagai contoh, berikut ini style yang akan menggunakan image bernama folder.gif yang ada di folder img.

```
ul {  
    list-style-image: url(img/folder.gif)  
}
```

dan hasilnya adalah



Suatu list dapat digunakan untuk menu navigasi. Berikut ini contoh tampilannya:



Meskipun secara sekilas mirip dengan penggunaan tabel, namun sebenarnya tampilan menu di atas dibuat dengan list.

Berikut ini stylenya:

```
ul {  
  list-style-type: none; /* tidak menampilkan marker */  
  padding: 2px;  
  border: 1px solid black;  
}  
  
li {  
  padding: 2px;  
  margin-bottom: 1px;  
  background: red;  
  border: 1px solid black;  
  font-family: arial;  
  font-size: 15px;  
  font-weight: bold;  
  color: #FFFFFF;  
}
```

Style Pada Link

CSS juga dapat digunakan untuk mengatur style link. Berikut ini beberapa jenis selector yang terkait dengan aktifitas link.

```
a:hover {                /* style yang muncul apabila link didekati kursor */
    ...
}

a:visited {              /* style yang muncul pada link yang pernah diklik */
    ...
}
```

Sebagai contoh, kita akan memodifikasi menu navigasi dari contoh bab sebelumnya dengan menambahkan link

Berikut ini code dasarnya

```
<h3>Menu Navigasi</h3>
<ul>
    <li><a href="menu1.htm">Menu 1</a></li>
    <li><a href="menu2.htm">Menu 2</a></li>
    <li><a href="menu3.htm">Menu 3</a></li>
    <li><a href="menu4.htm">Menu 4</a></li>
    <li><a href="menu5.htm">Menu 5</a></li>
    <li><a href="menu6.htm">Menu 6</a></li>
</ul>
```

Sedangkan style yang telah kita buat sebelumnya adalah

```
ul {
    list-style-type: none; /* tidak menampilkan marker */
    padding: 2px;
    border: 1px solid black;
}

li {
    padding: 2px;
    margin-bottom: 1px;
    background: red;
    border: 1px solid black;
    font-family: arial;
    font-size: 15px;
    font-weight: bold;
    color: #FFFFFF;
}
```

Secara default, tampilan dari code di atas adalah sebagai berikut



Sebuah tampilan yang kurang menarik karena bukan kombinasi warna yang bagus antara biru dan merah. Secara default, warna link adalah biru. Namun kita dapat mengubah warnanya dengan yang lain. Berikut ini style untuk memberi warna putih pada link.

```
a {  
    color: white;  
}
```



Kita pun dapat bermain-main sedikit dengan animasi link dengan CSS. Animasi yang dimaksud yaitu perubahan warna link ketika kursor didekatkan pada link tersebut. Misalkan kita ingin memberi warna kuning ketika kursor didekatkan link, maka stylenya adalah

```
a:hover {  
    color: yellow;  
}
```

Adapun efek dari penggunaan hover tampak pada gambar berikut ini. Namun sayang, gambar kursornya gak kena screen shot.



Link-link yang telah diklik biasanya akan berubah warna, dengan defaultnya adalah ungu. Untuk mengubah warna link yang telah dikunjungi tersebut, Anda dapat menggunakan selector `a:visited`

Berikut ini contohnya:

```
a:visited {  
    color: black;  
}
```



Pada gambar di atas, dua link pertama (menu 1 dan 2) adalah link yang pernah diklik, ditunjukkan dengan warna hitam.