

# BAB I

## DASAR-DASAR HTML

### A. KOMPETENSI DASAR

- Memahami struktur dasar dokumen HTML dan HTML5
- Mampu membuat dokumen HTML yang baik dan benar.
- Mampu memanfaatkan elemen-elemen dasar untuk menampilkan informasi.

### B. ALOKASI WAKTU

4 JS (4 x 50 menit)

### C. PETUNJUK

- Awali setiap aktivitas dengan do'a, semoga berkah dan mendapat kemudahan.
- Pahami tujuan, dasar teori, dan latihan-latihan praktikum dengan baik dan benar.
- Kerjakan tugas-tugas praktikum dengan baik, sabar, dan jujur.
- Tanyakan kepada asisten/dosen apabila ada hal-hal yang kurang jelas.

### D. DASAR TEORI

#### 1. HTML dan XHTML

HTML (*HyperText Markup Language*) merupakan sebuah bahasa *markup*, bukan bahasa pemrograman. Bahasa *markup* (Indonesia: markah) adalah bahasa yang mengombinasikan teks dan informasi tambahan mengenai teks tersebut. HTML merupakan dokumen standar yang digunakan untuk mendesain halaman *web*.

Pada awal tahun 2000, konsorsium W3C (*World Wide Web Consortium*) membuat perubahan besar melalui XHTML (*eXtensible Hypertext Markup Language*). Ide dasarnya, dalam upaya meningkatkan kompatibilitas dokumen HTML, W3C menambahkan struktur dan ekstensibilitas XML (*eXtensible Markup Language*) ke HTML.

Perkembangan HTML semakin pesat seiring diperkenalkannya HTML5 pada tahun 2009. HTML5 dibangun oleh konsorsium W3C untuk dimasukkan sebagai perubahan besar berikutnya pada standar HTML. Atas dasar ini, maka penting sekali untuk mengenal dan memahami HTML5 dengan baik.

## 2. Struktur Dokumen HTML

Setiap dokumen HTML harus diawali dengan tag `<html>` dan diakhiri dengan komplemennya, yakni tag `</html>` tag. Dokumen HTML juga menyertakan tiga pasang tag.

- Tag `<head>` dan `</head>`: digunakan untuk menyatakan informasi mengenai dokumen HTML.
- Tag `<title>` dan `</title>`: digunakan untuk menambahkan title di title bar browser.
- Tag `<body>` dan `</body>`: digunakan untuk melingkupi semua teks yang terdapat di halaman HTML.

Bentuk struktur dokumen HTML yang standar sebelum versi HTML5 diperlihatkan sebagai berikut:

```
<!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" xml:lang="en"
lang="en">

<head>
  <!-- Ini baris komentar, tidak diproses -->
  <title>Ini judul dokumen HTML4</title>
</head>

<body>
  Ini adalah teks di body
</body>

</html>
```

Pada HTML5, struktur penulisan lebih diringkas dengan bentuk sebagai berikut:

```
<!DOCTYPE html>
<html>

<head>
  <!-- Ini baris komentar, tidak diproses -->
  <title>Ini judul dokumen HTML5</title>
</head>

<body>
  Ini adalah teks di body
</body>

</html>
```

Di samping elemen utama di atas, masih terdapat berbagai jenis elemen yang dapat digunakan di dalam dokumen HTML. Salah satu elemen yang perlu

diperhatikan adalah untuk penulisan komentar. Di HTML, komentar dinyatakan dengan tag `<!--` dan diakhiri dengan tag `-->`.



Bagaimanapun, praktikum ini menuntut adanya standardisasi yang baik benar. Selain itu, penulisan dokumen HTML merujuk pada spesifikasi HTML5.

### 3. Persiapan Kebutuhan

Pada praktikum awal ini, meskipun dokumen HTML bisa ditampilkan tanpa menggunakan *web server*, namun di sini diwajibkan tetap menggunakan. Selain itu, praktikum ini juga menuntut pembuatan kode-kode yang baik, benar, dan valid. Berikut ini adalah kebutuhan-kebutuhan yang minimal diperlukan:

- a. WampServer  
Paket web server Apache, PHP, dan MySQL. Alasan pemilihan paket bundel seperti ini dikarenakan praktis sehingga tinggal memfokuskan pada pemrograman.
- b. Web Browser  
Sangat disarankan menggunakan browser utama Mozilla Firefox. Adapun untuk pembandingan, sebaiknya juga memanfaatkan browser lain.
- c. Editor Teks  
Editor teks untuk menuliskan kode-kode HTML pembentuk halaman aplikasi web (tidak diperkenankan menggunakan IDE seperti Dreamweaver dan sebagainya).
- d. HTML Validator  
Untuk menghasilkan dokumen HTML yang valid, sangat disarankan memasang validator, misalnya berupa *add-on* pada browser Firefox.

## E. LATIHAN

### 1. Membuat Dokumen HTML

Secara garis besar, struktur dokumen HTML terdiri dari dua bagian: *header* dan *bodi*. Di mana *header* mendefinisikan informasi mengenai dokumen, sedangkan *bodi* mendefinisikan tubuh atau isi dokumen.

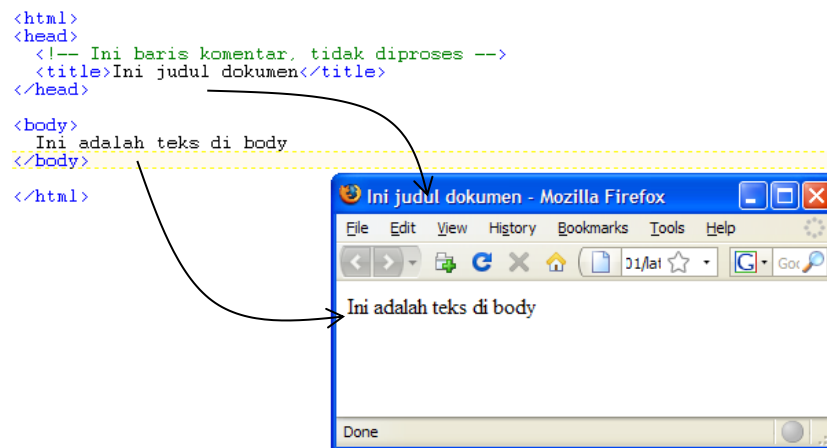
Langkah-langkah pembuatan dokumen HTML diperlihatkan sebagai berikut:

1. Buka editor teks.
2. Ketikkan teks (kode-kode HTML) berikut:

```
<!DOCTYPE html>
<html>
<head>
  <!-- Ini baris komentar, tidak diproses -->
  <title>Ini judul dokumen</title>
</head>

<body>
  Ini adalah teks di body
</body>
</html>
```

3. Simpan dokumen HTML dengan nama **latihan1.1.html** dan letakkan di lokasi direktori web, misal **C:\wamp\www**. Perhatikan, **www** merupakan direktori web dan sebaiknya buat subdirektori di dalamnya.



Gambar 1. Struktur Dokumen HTML

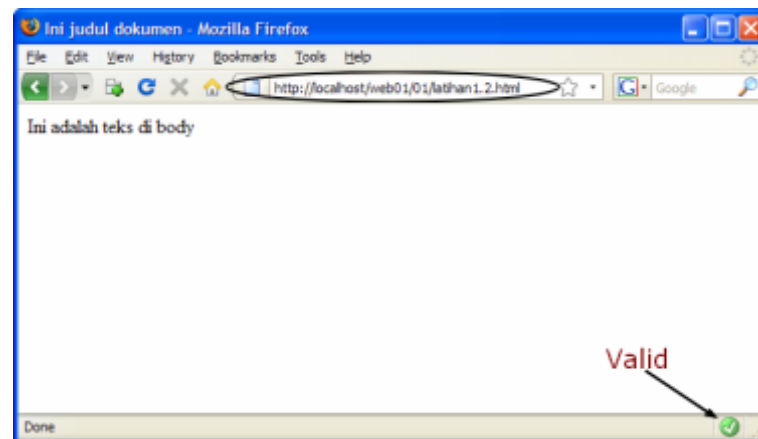
Bagaimanapun, dokumen HTML memang bisa ditampilkan dengan mengklik ganda di mana pun lokasinya. Namun perlu diperhatikan, langkah ini bukanlah cara untuk menjalankan halaman web di server lokal.

## 2. Publikasi Halaman Web

Untuk menguji aplikasi web, kita mempublikasikannya ke web server, baik secara lokal maupun Internet. Lingkungan lokal tentu merupakan pilihan yang efisien, khususnya ketika aplikasi masih dalam tahap pengembangan.

Langkah yang diperlukan untuk publikasi ini sangat sederhana.

1. Pastikan bahwa file dokumen sudah diletakkan di direktori web, misalnya untuk WampServer lokasi defaultnya adalah **www**. Untuk paket web server lainnya, termasuk Apache (versi tunggal) adalah **htdocs**.
2. Pastikan bahwa web server sudah dijalankan.
3. Buka web browser, kemudian ketikkan alamat URL yang merujuk ke lokasi dokumen. Perhatikan contohnya seperti Gambar 2.



Gambar 2. Mengakses halaman web dari server lokal

## 3. Format Teks

HTML menyediakan beragam elemen yang dapat dimanfaatkan untuk pemformatan teks.

- Heading

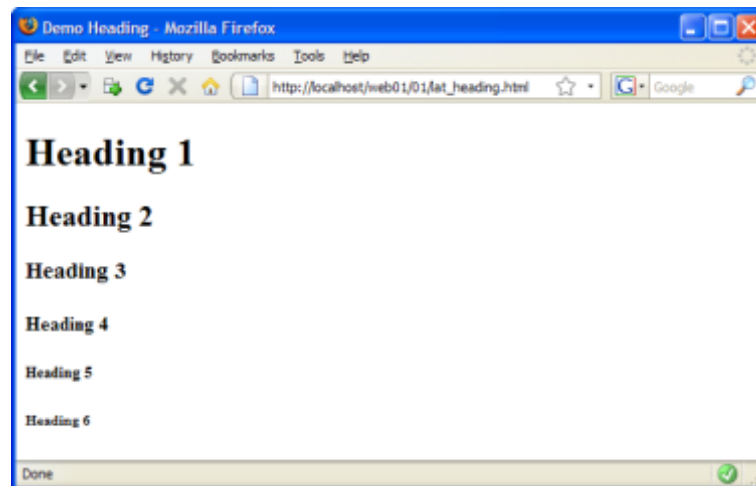
Heading merupakan salah satu elemen penting di dalam dokumen HTML. Heading didefinisikan menggunakan tag `<hn>` dan diakhiri dengan `</hn>`, di mana `n` menyatakan tipe dengan nilai 1-6.

Untuk mengetahui bentuk semua jenis heading, buat kode HTML seperti di bawah ini. Perhatikan, kode HTML ini sengaja diringkas guna memudahkan penulisan. Jadi, dalam implementasinya harus mendeklarasikan semua elemen-elemen dasar.

```
<body>
  <h1>Heading 1</h1>
  <h2>Heading 2</h2>
```

```
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>

</body>
```



Gambar 3. Tampilan heading 1-6

Elemen heading menyediakan atribut `align` yang dapat digunakan untuk mengatur posisi teks.

```
<body>

  <h1 align="right">Heading 1</h1>
  <h2 align="left">Heading 2</h2>
  <h3 align="center">Heading 3</h3>

</body>
```



Gambar 4. Mengatur posisi heading

- Paragraf

Sebagaimana teks pada umumnya, dokumen HTML dapat terdiri dari kumpulan paragraf. Dalam konteks HTML, paragraf direpresentasikan melalui tag <p>. Tag <p> sebenarnya merupakan tag pasangan, meski dalam implementasinya kerap kali diabaikan.

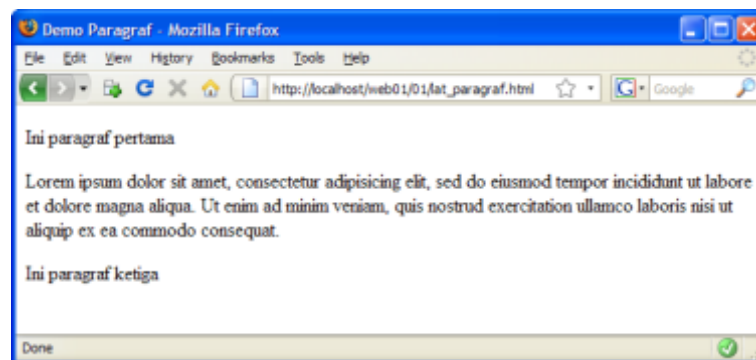
```
<body>

  <p>
  Ini paragraf pertama

  <p>
  Lorem ipsum dolor sit amet, consectetur adipiscing elit,
  sed do eiusmod tempor incididunt ut labore et dolore magna
  aliqua.
  Ut enim ad minim veniam, quis nostrud exercitation ullamco
  laboris
  nisi ut aliquip ex ea commodo consequat.

  <p>
  Ini paragraf ketiga

</body>
```



Gambar 5. Menggunakan paragraf

Sebagaimana heading, kita juga bisa mengatur posisi paragraf dengan memanfaatkan atribut `align`. Sebagai contoh, dokumen berikut akan menghasilkan paragraf rata tengah, kiri kanan, dan kanan.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Demo Paragraf</title>
</head>

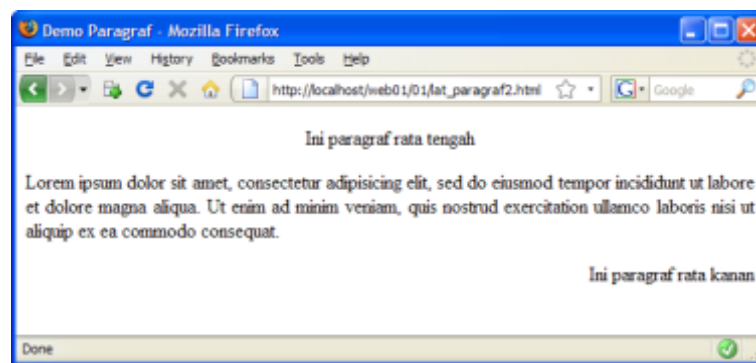
<body>

  <p align="center">
  Ini paragraf rata tengah
```

```
<p align="justify">
  Lorem ipsum dolor sit amet, consectetur adipisicing elit,
  sed do eiusmod tempor incididunt ut labore et dolore magna
  aliqua.
  Ut enim ad minim veniam, quis nostrud exercitation ullamco
  laboris
  nisi ut aliquip ex ea commodo consequat.

  <p align="right">
  Ini paragraf rata kanan

</body>
</html>
```



Gambar 6. Mengatur posisi paragraf

Secara normal, baris baru akan ditambahkan di antara dua paragraf. Adapun jika kita hanya ingin membuat baru, kita tidak harus menggunakan paragraf. Sebagai gantinya, gunakan tag line break `<br />`.

```
<body>

  Membuat baris baru <br />
  Membuat baris baru <br />

  <p>
  <!-- break di dalam paragraf -->
  Lorem ipsum dolor sit amet, consectetur adipisicing elit,
  <br />
  sed do eiusmod tempor incididunt ut labore et dolore magna
  aliqua.
  <br />
  Ut enim ad minim veniam, quis nostrud exercitation ullamco
  laboris
  nisi ut aliquip ex ea commodo consequat.

</body>
```





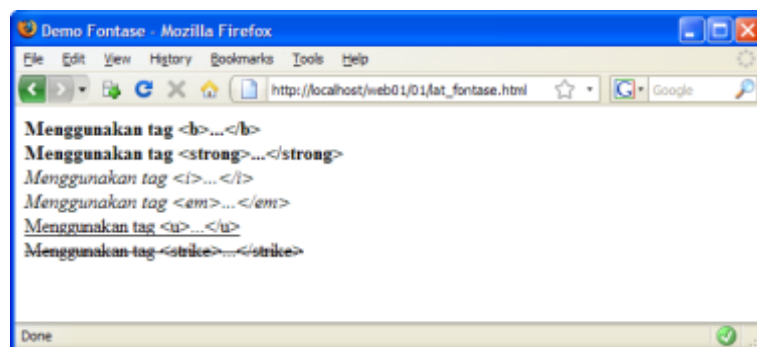
Gambar 7. Membuat baris baru

- Fontase

HTML menyediakan sejumlah elemen yang dapat dimanfaatkan untuk mengatur font, seperti huruf tebal, huruf miring, garis bawah, dan masih banyak lagi.

Sebagai tambahan, di sini juga akan dijelaskan mengenai cara mencetak tag. Seperti diketahui, tag <p> di dokumen secara otomatis akan diterjemahkan sebagai paragraf. Adapun untuk mencetak karakter <p> di layar, kita perlu menggunakan nama entitas. Sebagai contoh, karakter < > dinyatakan dengan nama entitas &lt; dan &gt;.

```
<body>
  <b>Menggunakan tag &lt;b&gt;...&lt;/b&gt;</b> <br />
  <strong>Menggunakan tag
&lt;strong&gt;...&lt;/strong&gt;</strong>
  <br />
  <i>Menggunakan tag &lt;i&gt;...&lt;/i&gt;</i> <br />
  <em>Menggunakan tag &lt;em&gt;...&lt;/em&gt;</em> <br />
  <u>Menggunakan tag &lt;u&gt;...&lt;/u&gt;</u> <br />
  <del>Menggunakan tag
&lt;del&gt;...&lt;/del&gt;</del> <br />
</body>
```



Gambar 8. Menggunakan style font

- Karakter Khusus

Di HTML, kita juga bisa menampilkan karakter-karakter khusus dengan memanfaatkan nama entitas. Tabel berikut memperlihatkan beberapa jenis karakter khusus yang dapat digunakan beserta nama entitasnya.

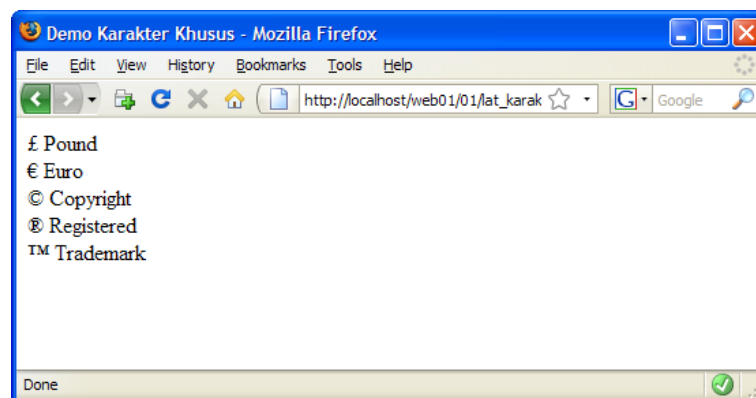
Karakter	Deskripsi	Nama Entitas
¢	Cent	&cent;
£	Pound	&pound;
¥	Yen	&yen;
€	Euro	&euro;
©	Copyright	&copy;
®	Registered	&reg;
™	Trademark	&trade;

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Demo Karakter Khusus</title>
</head>

<body>

  &pound; Pound <br />
  &euro; Euro <br />
  &copy; Copyright <br />
  &reg; Registered <br />
  &trade; Trademark <br />

</body>
</html>
```



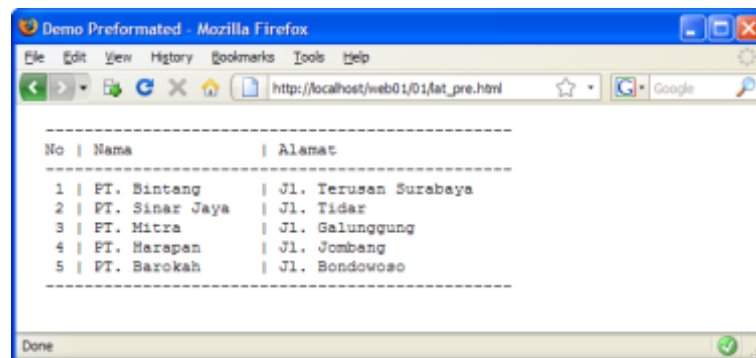
Gambar 9. Karakter-karakter khusus

#### 4. Preformatted Text

Adakalanya kita menginginkan agar teks yang tertulis di editor teks dapat ditampilkan apa adanya. Pada kasus seperti ini, kita bisa memanfaatkan keberadaan tag `<pre>`. Tag ini akan menampilkan teks dengan font Courier dan tetap mempertahankan spasi serta baris baru.

```
<body>

  <pre>
  -----
  No | Nama                | Alamat
  -----
  1 | PT. Bintang         | Jl. Terusan Surabaya
  2 | PT. Sinar Jaya      | Jl. Tidar
  3 | PT. Mitra           | Jl. Galunggung
  4 | PT. Harapan         | Jl. Jombang
  5 | PT. Barokah         | Jl. Bondowoso
  -----
  </pre>
</body>
```



Gambar 10. Tampilan preformatted text

Untuk melihat perbedaannya, hilangkan tag `<pre>` kemudian perhatikan hasilnya.

#### 5. Garis Horizontal

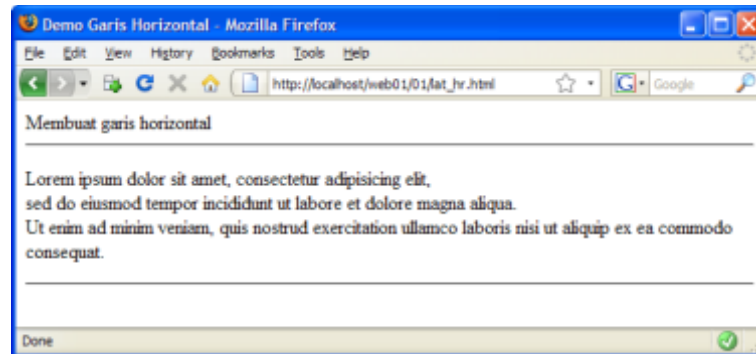
Di HTML, garis horizontal direpresentasikan melalui tag `<hr />`. Meskipun kebanyakan browser me-render elemen ini dengan visualisasi yang sedikit berbeda, namun pada hakekatnya mencerminkan sebuah bentuk garis horizontal.

```
<body>

  Membuat garis horizontal <hr />

  <p>
  Lorem ipsum dolor sit amet, consectetur adipiscing elit,
  <br />
```

```
sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.  
<br />  
Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris  
nisi ut aliquip ex ea commodo consequat.  
  
<hr />  
</body>
```



Gambar 11. Garis horizontal

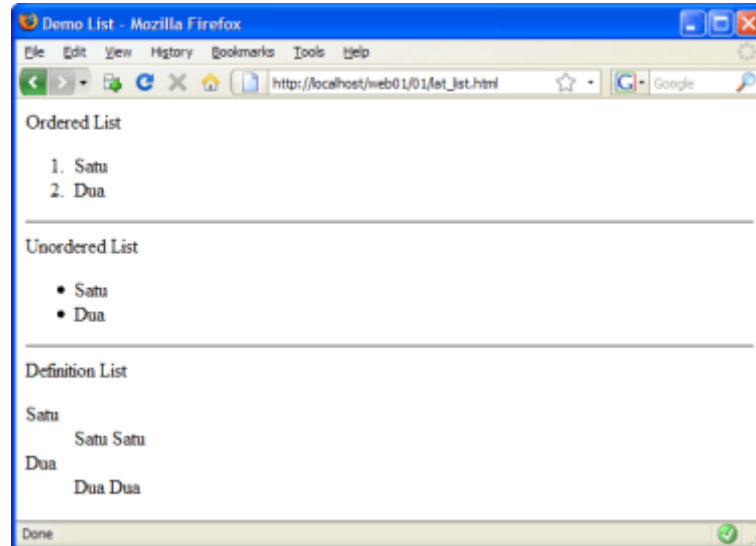
## 6. Menggunakan List

HTML mendukung *list* dalam bentuk terurut (*ordered*), tak terurut (*unordered*), dan definisi (*definition*). Untuk setiap bentuk *list* ini, terdapat *list item*—dinyatakan melalui tag `<li>` berpasangan—yang merepresentasikan item-item *list*.

```
<body>  
  
Ordered List  
<ol>  
  <li>Satu</li>  
  <li>Dua</li>  
</ol>  
  
<hr />  
  
Unordered List  
<ul>  
  <li>Satu</li>  
  <li>Dua</li>  
</ul>  
  
<hr />  
  
Definition List  
<dl>  
  <dt>Satu</dt>  
  <dd>Satu Satu</dd>  
  <dt>Dua</dt>
```

```
<dd>Dua Dua</dd>
</dl>

</body>
```



Gambar 12. Menggunakan list

## 7. Pewarnaan

Untuk memberikan warna background, HTML menyediakan atribut `bgcolor` or di tag `<body>`. Atribut ini dapat diisi dengan nama warna—misalnya `red`—atau kode heksadesimal—misalnya `#FFFFFF`.

Khusus untuk elemen-elemen lain tertentu, tersedia atribut `color` yang memungkinkan kita melakukan pewarnaan. Sama seperti `bgcolor`, nilai atribut ini juga dapat berupa nama warna atau kode heksadesimal.

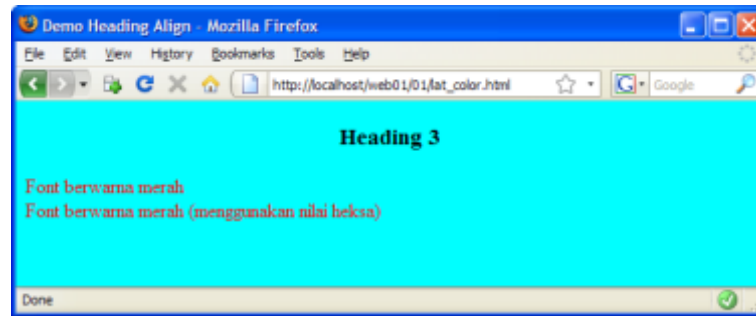
```
<!-- memberi warna aqua di body -->
<body bgcolor="aqua">

  <h3 align="center">Heading 3</h3>

  <font color="red">Font berwarna merah</font> <br />

  <font color="#FF0000">
  Font berwarna merah (menggunakan nilai heksa)</font>

</body>
```



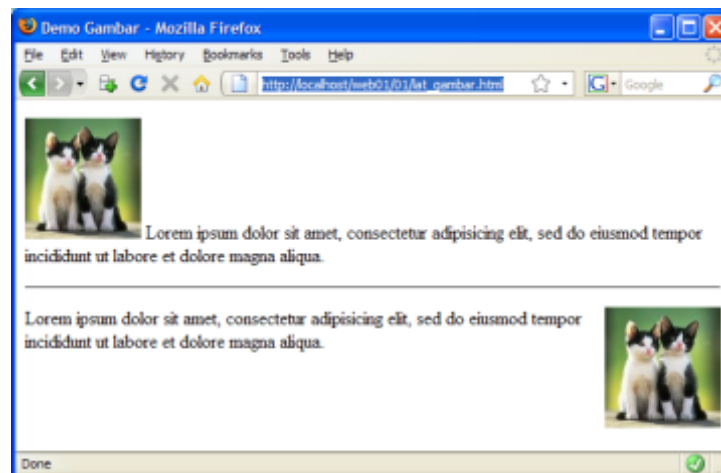
Gambar 13. Menambahkan warna di halaman web

## 8. Bekerja dengan Gambar

Tak hanya teks, kita juga bisa menyisipkan gambar di dalam dokumen HTML. Untuk keperluan ini, HTML menyediakan tag `<img>` yang didukung dengan sejumlah atribut.

```
<body>
  <p>
    <!-- menggunakan path relative -->
    
    Lorem ipsum dolor sit amet, consectetur adipiscing elit,
    sed do eiusmod tempor incididunt ut labore et dolore magna
    aliqua.
    <hr />

    <p>
    <!-- menggunakan path absolute -->
    
    Lorem ipsum dolor sit amet, consectetur adipiscing elit,
    sed do eiusmod tempor incididunt ut labore et dolore magna
    aliqua.
  </body>
```

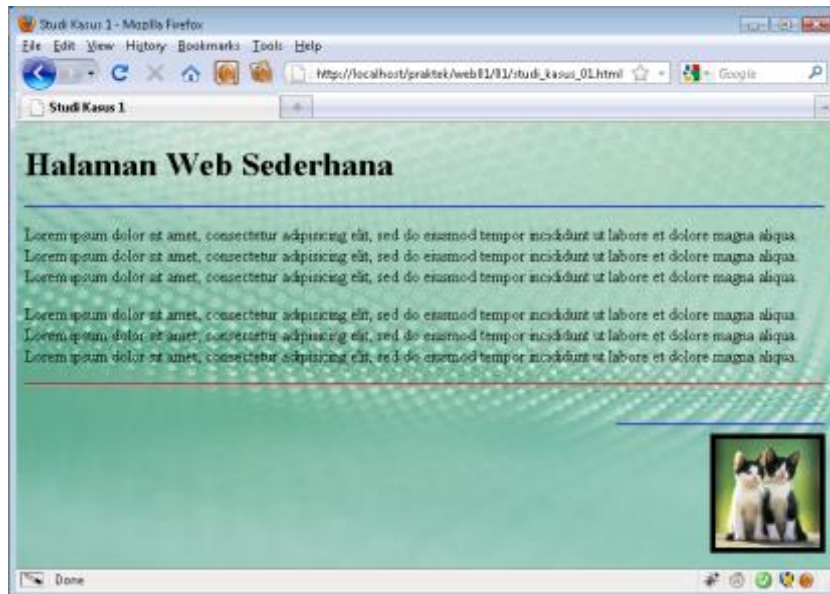


Gambar 14. Menyisipkan gambar



## F. STUDI KASUS

1. Buat halaman web sederhana yang melibatkan elemen teks, garis, warna, dan gambar. Di mana terdapat minimal sebuah heading dan garis berwarna, gambar dengan garis tepi (border), dan halaman utama dengan latar belakang gambar sembarang. Contoh tampilannya diperlihatkan seperti Gambar 15.



Gambar 15. Kombinasi elemen-elemen HTML