



# **PEMROGRAMAN WEB I**

## **Pertemuan ke-4**

### **Heading-Komentar-Text Formatting di HTML**

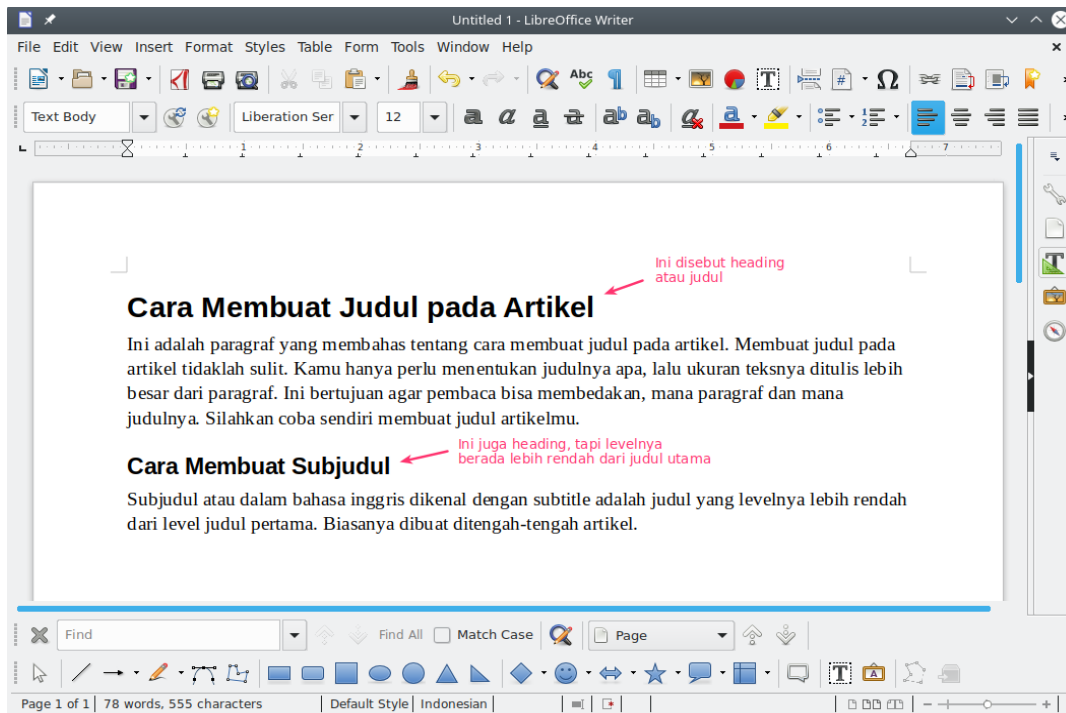
Riad Sahara, S.Si., MT



# I. Heading

**Heading adalah sebuah judul** yang biasanya diberikan pada halaman atau beberapa bagian dari artikel.

Jika kamu sering menulis artikel, pasti ini tidak asing buatmu.



Lalu, bagaimana caranya membuat judul di HTML?

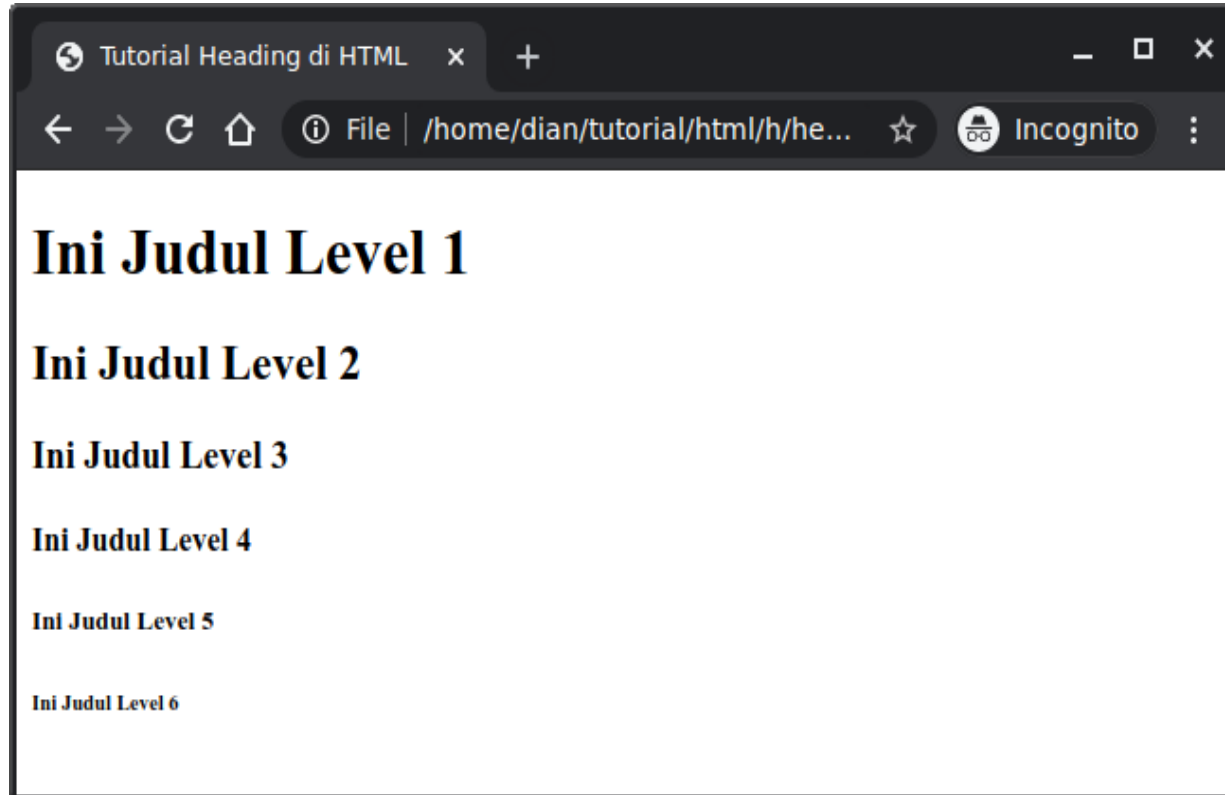
Mari kita bahas:



# 1.1. Cara Membuat Heading di HTML

Judul pada HTML dapat kita buat dengan tag `<h1>` sampai `<h6>`. Tag `<h1>` digunakan pada judul level pertama. Lalu tag lainnya digunakan pada sub heading atau level berikutnya.

Masing-masing judul akan ditampilkan dengan ukuran teks yang berbeda. Tag `<h1>` adalah yang paling besar, dan tag `<h6>` paling kecil.



```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<title>Tutorial Heading di HTML</title>
```

```
</head>
```

```
<body>
```

```
<h1>Ini Judul Level 1</h1>
```

```
<h2>Ini Judul Level 2</h2>
```

```
<h3>Ini Judul Level 3</h3>
```

```
<h4>Ini Judul Level 4</h4>
```

```
<h5>Ini Judul Level 5</h5>
```

```
<h6>Ini Judul Level 6</h6>
```

```
</body>
```

```
</html>
```



```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<title>Tutorial Heading di HTML</title>
```

```
</head>
```

```
<body>
```

```
<h1>Belajar Heading di HTML</h1>
```

```
<p>
```

Heading di HTML ada enam, yakni H1, H2, H3, H4, H5, dan H6.

Heading berfungsi untuk membuat judul untuk artikel dan juga

sub judul pada bagian artikel.

```
</p>
```

```
<h2>Membuat Sub Judul</h2>
```

```
<p>
```

Sub judul atau sub heading dimulai dengan tag H2. Lalu diikuti

dengan tag selanjutnya hingga sampai H6. Sementara itu, tag H1

hanya digunakan untuk judul artikel saja.

```
</p>
```

```
</body>
```

```
</html>
```



Hasilnya:





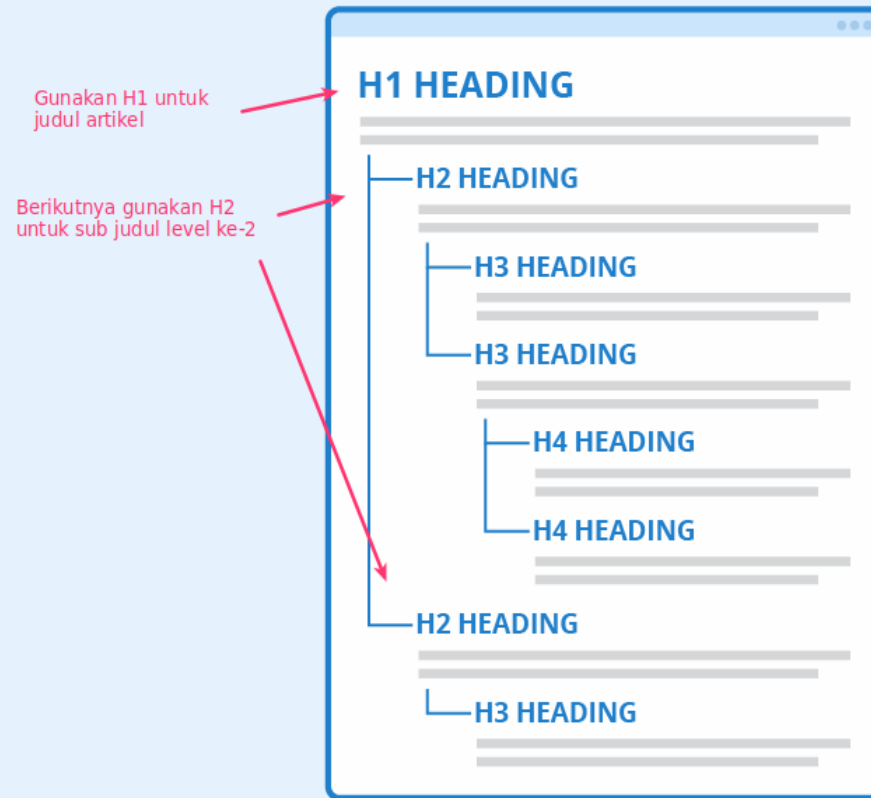
## 1.2. Urutan Penulisan Heading

Apakah boleh menulis judul yang tidak urut?

Maksudnya.. seperti menggunakan <h6> untuk judul awal, lalu berikutnya <h4> untuk sub judul.

Ini boleh-boleh saja, tapi kurang bagus untuk SEO.

Penulisan judul yang bagus adalah judul yang mengikuti levelnya.





## 1.3. Atribut untuk Heading

Masih ingat tentang atribut?

Coba buka lagi pembahasan tentang: Apa itu Tag, Elemen, dan Atribut di HTML?

Heading tidak memiliki atribut khusus. Ia biasanya menggunakan atribut global.

Contohnya seperti:

- id untuk memberikan nama id unik. Biasanya ini akan digunakan oleh link, CSS, dan Javascript;
- class untuk memberikan nama class yang akan dipakai oleh CSS;
- style untuk memberikan kode css secara langsung;
- title untuk menambahkan informasi tambahan; dll.

Daftar atribut global, bisa kamu lihat di MDN: [HTML Global Attributes](https://developer.mozilla.org/en-US/docs/Web/HTML/Global_attributes).





## 1.4. Perbedaan Heading dengan Title dan Header

Meskipun sama-sama digunakan untuk urusan judul, tapi tiga elemen ini berbeda.

- Heading adalah judul untuk **artikel** dan **bagian artikel** yang dibuat dengan tag `<h1>` sampai `<h6>`
- Title adalah **judul dari web** yang dibuat dengan tag `<title>`
- Header adalah **bagian kepala** (kop) pada web yang dibuat dengan tag `<header>`

Coba perhatikan gambar berikut:





## II. Komentar

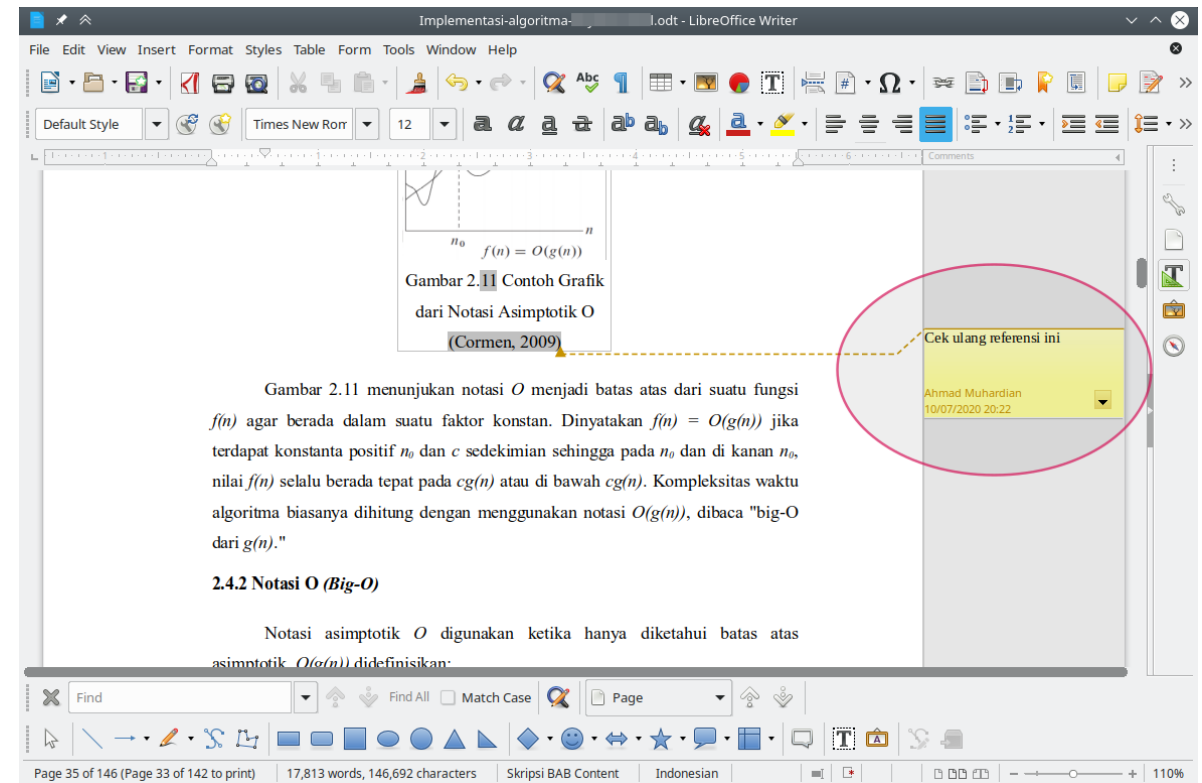
Sampai ditahap ini, apakah kamu sudah ingat semua tag HTML yang dipelajari?

Sudah lupa?

Tidak apa-apa, itu adalah sifat dari manusia. Otak kita memang tidak seperti komputer yang bisa mengingat dengan cepat dan permanen.

Karena itu, kita membutuhkan catatan agar bisa mengingat.

Coba perhatikan ini:





Yang saya lingkari merah itu adalah komentar atau catatan yang akan mengingatkan saya tentang pekerjaan yang harus dilakukan di bagian tersebut.

Karena tulisan saya sangat panjang, jadi komentar ini sangatlah berguna.

Nah, di HTML juga kita akan menemukan hal yang demikian. Nantinya, dokumen HTML kita akan semakin banyak dan panjang.

Agar bisa mengingat strukturnya, sebaiknya ditambahkan komentar.

Bagaimana cara membuatnya?

Mari kita pelajari.



## 2.1. Apa itu Komentar dalam HTML?

Komentar adalah elemen yang akan diabaikan oleh browser. Ia tidak akan ditampilkan di dalam web.

Komentar biasanya digunakan untuk memberikan informasi tambahan pada kode HTML dan kadang juga digunakan untuk menon-aktifkan beberapa kode HTML.



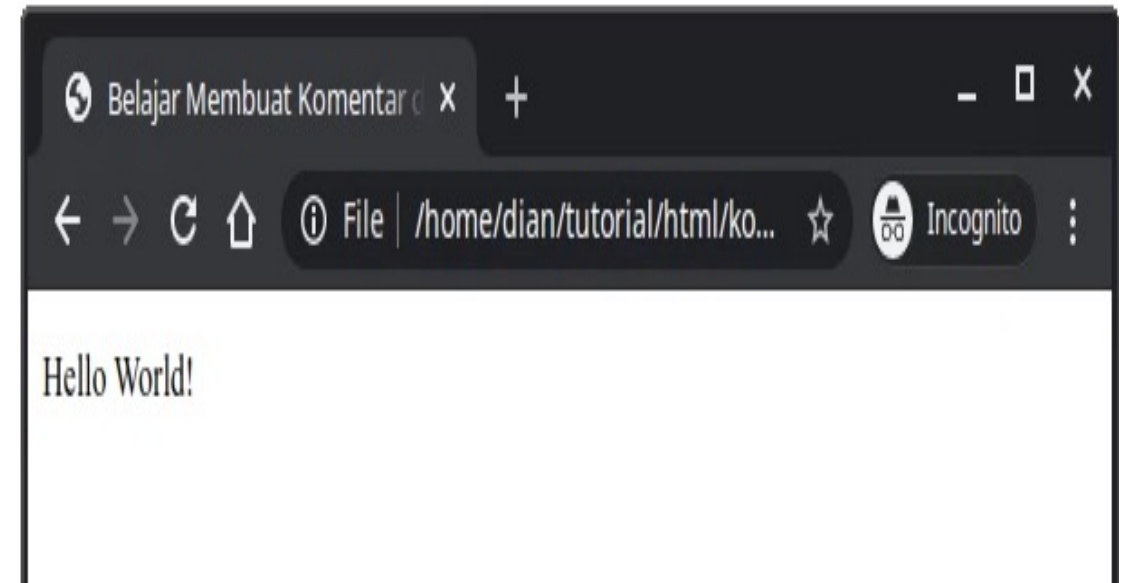
## 2.2. Cara Membuat Komentar di HTML

Komentar dapat kita buat dengan tag `<!-- -->`.





```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Belajar Membuat Komentar di HTML</title>
  </head>
  <body>
    <!-- ini adalah komentar -->
    <p>Hello World!</p>
    <!-- ini juga komentar
    dan ditulis dalam dua baris -->
  </body>
</html>
```





## 2.3. Jalan Pintas untuk Membuat Komentar

Jika kamu menggunakan teks editor Visual Studio Code, kamu bisa membuat komentar dengan menekan Ctrl+/.

```
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <title>Belajar Membuat Komentar di HTML</title>
5      </head>
6      <body>
7          <!-- ini adalah komentar -->
8          <p>Hello World!</p>
9          <!-- ini juga komentar
10         dan ditulis dalam dua baris -->
11         <!-- komentar -->
12      </body>
13 </html>
```

Cara ini lebih cepat dibandingkan dengan harus menulis secara manual tag komentarnya.



### III. Text Formatting di HTML

Coba perhatikan kedua paragraf ini:

Plain text	Rich Text
Ini adalah teks polos atau biasa, dalam bahasa inggris disebut plain text. Teks ini tidak menggunakan formatting apapun, sehingga terlihat biasa saja.	Ini adalah teks yang <u>terformat</u> , kadang disebut juga dengan <i>rich text</i> . Soalnya, teks ini memiliki <i>formatting</i> seperti <i>italic</i> , <b>bold</b> dan <u><b>underline</b></u> .

Kamu lebih tertarik baca yang mana? yang terformat, atau yang polos (*plain text*)?

Teks yang terformat akan lebih mudah dibaca, karena ada penegasan seperti **teks tebal**, *miring*, dan garis bawah.

Saat kita mencari kata atau kalimat penting.. mata kita akan lebih mudah menemukannya pada teks yang terformat dibandingkan *plain text*.

Karena itu, dalam membuat web.. ada baiknya menggunakan format teks.

Nah, di HTML terdapat tag-tag yang khusus digunakan untuk *text formatting*.





## 3.1. Membuat Teks Tebal

Teks tebal biasanya digunakan untuk memberikan penegasan pada teks tertentu, misalnya seperti judul, subjudul, huruf penting, dll.

Tag yang digunakan untuk membuat teks tebal di HTML adalah tag `<b>` (*bold*) dan tag `<strong>`.

Kamu bebas mau pakai yang mana saja, hasilnya akan sama-sama tebal.



<body>

<h1>Text Tebal di HTML</h1>

<p>

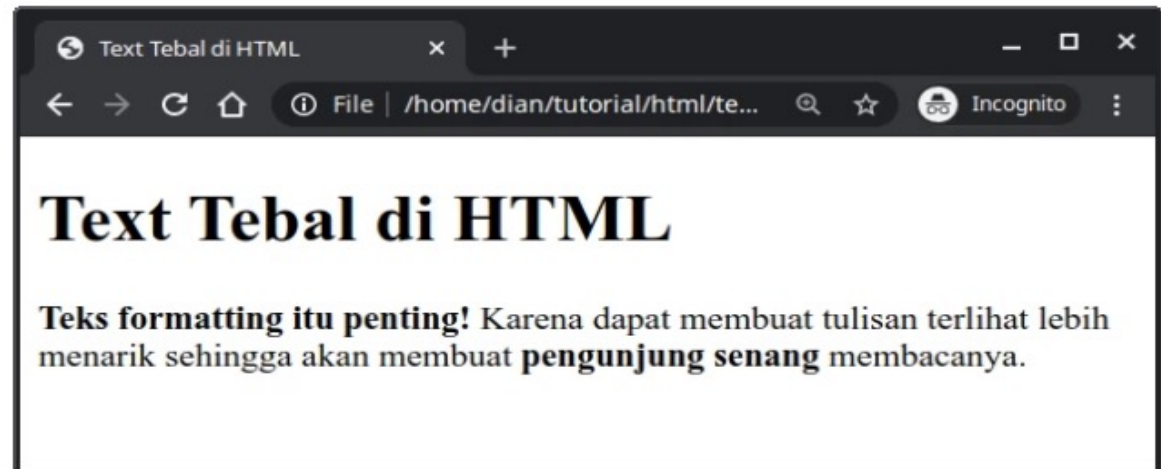
<strong>Teks formatting itu penting!</strong> Karena dapat membuat tulisan

terlihat lebih menarik sehingga akan membuat <b>pengunjung senang</b>

membacanya.

</p>

</body>





## 3.2. Membuat Teks Miring

Teks miring biasanya digunakan untuk menegaskan sebuah kata atau istilah baru. Teks miring di HTML dapat kita buat dengan tag `<i>` (*italic*) dan juga tag `<em>` (emphasis).

```
<body>
```

```
<h1>Text Miring di HTML</h1>
```

```
<p>
```

Gunakan `<i>teks miring</i>` untuk memberikan penekanan pada teks,

sehingga akan `<em>menarik perhatian</em>` pembaca. Biasanya

digunakan pada `<i>istilah asing</i>` atau kata serapan dari

`<em>bahasa daerah</em>`.

```
</p>
```

```
</body>
```





## 3.3. Membuat Garis Bawah pada Teks

Garis bawah biasanya digunakan untuk menandai teks yang disisipkan atau teks yang memiliki arti penting dibandingkan teks normal lainnya.

Garis bawah di HTML dapat kita buat dengan tag `<u>` (underline) atau juga tag `<ins>` (insert).

```
<body>
```

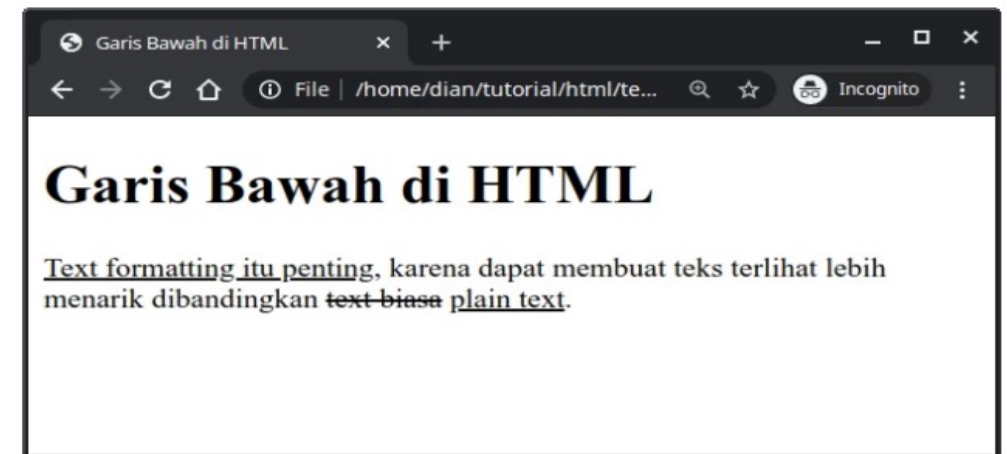
```
<h1>Garis Bawah di HTML</h1>
```

```
<p>
```

```
<u>Text formatting itu penting</u>, karena dapat membuat teks terlihat  
lebih menarik dibandingkan <del>text biasa</del> <ins>plain text</ins>.
```

```
</p>
```

```
</body>
```





## 3.4. Membuat Teks Tercoret

Teks tercoret memiliki arti teks yang dihapus. Biasanya untuk memberitahu pembaca bahwa teks tersebut tidak dipakai atau dihapus.

Tag untuk membuat teks tercoret di HTML adalah tag `<s>` (*strikethrough*) atau bisa juga dengan tag `<del>` (*delete*).

`<body>`

`<h1>Text Tercoret di HTML</h1>`

`<p>`

Coretlah teks yang tidak `<s>dibutuhkan</s>` terpakai, ini bisa memberitahu pembaca tentang perbaikan dari teks tersebut. Kadang juga teks `<del>tercoret</del>`

`<ins>yang dicoret</ins>`, diperbaiki dengan menambahkan teks dengan garis bawah.

`</p>`

`</body>`





## 3.5. Membuat Pangkat di HTML

Pangkat biasanya digunakan pada rumus. Ada dua jenis pangkat yang bisa dibuat di HTML, yakni pangkat yang berada di atas (*superscript*) dan pangkat di bawah (*subscript*). Tag untuk membuat pangkat di HTML adalah tag `<sup>` dan `<sub>`.

- `<sup>` untuk membuat pangkat di atas
- `<sub>` untuk membuat pangkat di bawah



<body>

<h1>Membuat Pangkat di HTML</h1>

<p>

Rumus luas persegi adalah  $S^2$ , dimana  $S$  adalah sisi dari persegi. Lalu  $O_2$  adalah rumus kimia dari oksigen.

</p>

</body>





## 3.6. Membuat Marker untuk Teks

Marker biasanya digunakan untuk menandai teks yang penting atau kata kunci yang penting. Marker di HTML dapat kita buat dengan tag `<mark>`.

```
<body>
```

```
<h1>Membuat marker teks di HTML</h1>
```

```
<p>
```

Marker biasanya digunakan untuk menandai bagian teks yang penting.

Kalau di dunia nyata, kita `<mark>menggunakan stabilo</mark>` untuk

membuat marker.

```
</p>
```

```
</body>
```



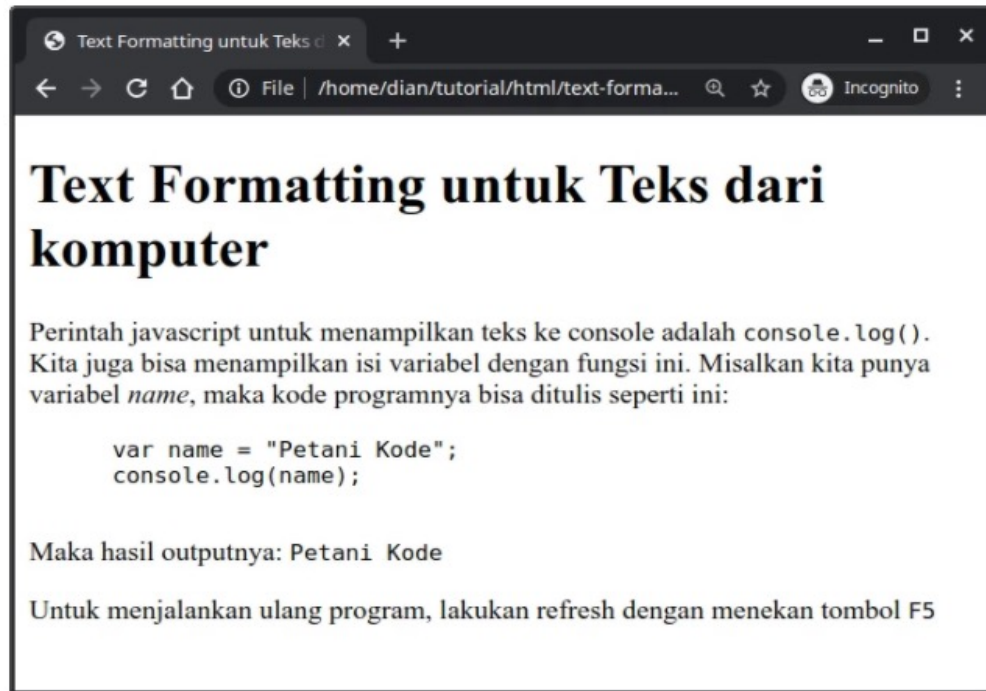




## 3.7. Teks Formatting untuk Komputer

Selain dari teks formatting di atas, ada juga teks formatting yang khusus untuk menandai teks yang berasal dari komputer. Berikut ini beberapa tag yang digunakan untuk memformat teks dari komputer:

- `<code>` untuk menandai bagian dari kode program;
- `<samp>` untuk menandai output dari program komputer;
- `<kbd>` untuk menandai tombol keyboard;
- `<var>` untuk menandai sebuah variabel;
- `<pre>` untuk preformatting teks.



<body>

<h1>Text Formatting untuk Teks dari komputer</h1>

<p>Perintah javascript untuk menampilkan teks ke console adalah

<code>console.log()</code>. Kita juga bisa menampilkan isi variabel dengan fungsi ini. Misalkan kita punya variabel <var>name</var>, maka kode programnya bisa ditulis seperti ini:

<pre>

var name = "Petani Kode";

console.log(name);

</pre>

Maka hasil outputnya: <samp>Petani Kode</samp> </p>

<p>Untuk menjalankan ulang program, lakukan refresh dengan menekan tombol

<kbd>F5</kbd></p>

</body>



## 3.8. Menggabungkan Format

Apakah format teks dapat digabungkan?

Misalkan kiga ingin membuat teks tebal dan garis bawah, apakah bisa?

Tentu saja bisa.

Caranya:

Ya tinggal dipakai saja tag-tagnya, misal mau menggabungkan *bold* dengan *underline*.. maka kita tinggal pakai tag `<b>` dan `<u>`.



<body>

<h1>Penggabungan Format Teks di HTML</h1>

<p>

Penggabungan format teks bisa dilakukan dengan menuliskan tag-tag yang akan dipakai. Misalkan <b><u>tebal dan garis bawah</u></b>, maka kita tinggal pakai tag <b>b</b> dan tag <b>u</b>.

</p>

</body>





# Referensi

1. Muhardian, Ahmad. 2016. Tutorial HTML untuk Pemula “Belajar HTML dari dasar hingga bisa hosting web sendiri”. PETANIKODE.COM  
<https://www.petanikode.com/tutorial/html/>
2. Sarwati Rahayu. 2011. *Pemrograman Web*. Universitas Mercu Buana, Jakarta.
3. Gregorius, Agung. *Buku Pintar HTML5 + CSS3 + DreamWeaver CS6*. Penerbit: Jubilee Enterprise. Yogyakarta, 13 September 2012.
4. Lubbers, P., Albers, B., & Salim, F. (2011). Overview of HTML5. In *Pro HTML5 Programming* (pp. 1-22). Apress.
5. West, W., & Pulimood, S. M. (2012). Analysis of privacy and security in HTML5 web storage. *Journal of Computing Sciences in Colleges*, 27(3), 80-87.
6. Zibula, A., & Majchrzak, T. A. (2012, April). Cross-platform development using HTML5, jQuery mobile, and phonegap: realizing a smart meter application. In *International Conference on Web Information Systems and Technologies* (pp. 16-33). Springer, Berlin, Heidelberg.
7. Garaizar, P., Vadillo, M. A., & Lopez-de-Ipina, D. (2012, July). Benefits and pitfalls of using HTML5 APIs for online experiments and simulations. In *2012 9th International Conference on Remote Engineering and Virtual Instrumentation (REV)* (pp. 1-7). IEEE.
8. Zibula, A., Majchrzak, T. A., Krempels, K. H., & Cordeiro, J. (2012). Developing a Cross-platform Mobile Smart Meter Application using HTML5, jQuery Mobile and PhoneGap. In *WEBIST* (pp. 13-23).
9. Kim, T., Kim, B., & Kim, J. (2013). Development of a lever learning webapp for an HTML5-based cross-platform. In *Multimedia and Ubiquitous Engineering* (pp. 313-320). Springer, Dordrecht.
10. Brandon, D. (2011). HTML 5 programming: conference tutorial. *Journal of Computing Sciences in Colleges*, 26(5), 61-61.



# Terima Kasih

Riad Sahara, S.SI., MT