

Mata Kuliah	:	Pemograman Web
Bobot Sks	:	3
Dosen Pengembang	:	Riad Sahara, S.Si, M.T
Tutor	:	Muhammad Ikhwani Saputra, S.Kom, M.Kom
Capaian Pembelajaran Mata Kuliah	:	1. 2.
Kompetensi Akhir Di Setiap Tahap (Sub-Cpmk)	:	1. Mahasiswa mampu memahami dan dapat membuat Heading dan Komentar HTML.
Minggu Perkuliahan Online Ke-	:	4

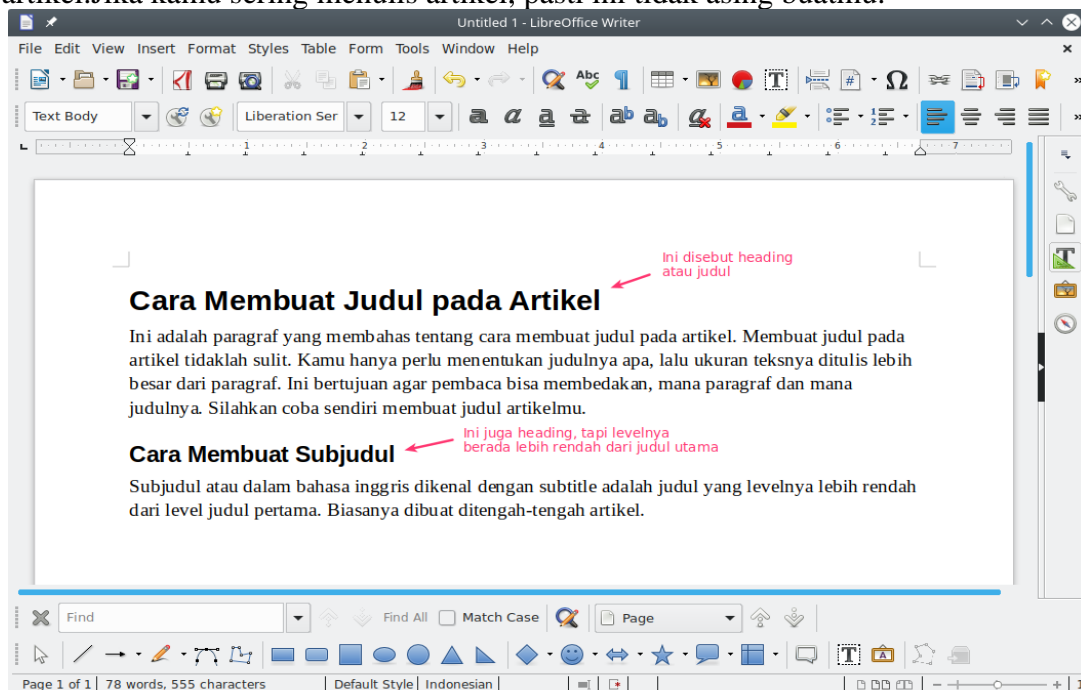
JUDUL TOPIK – Heading, Komentar, dan Text Formatting di HTML

1. Pendahuluan

2. Isi

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.

Contoh:

```
<!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>

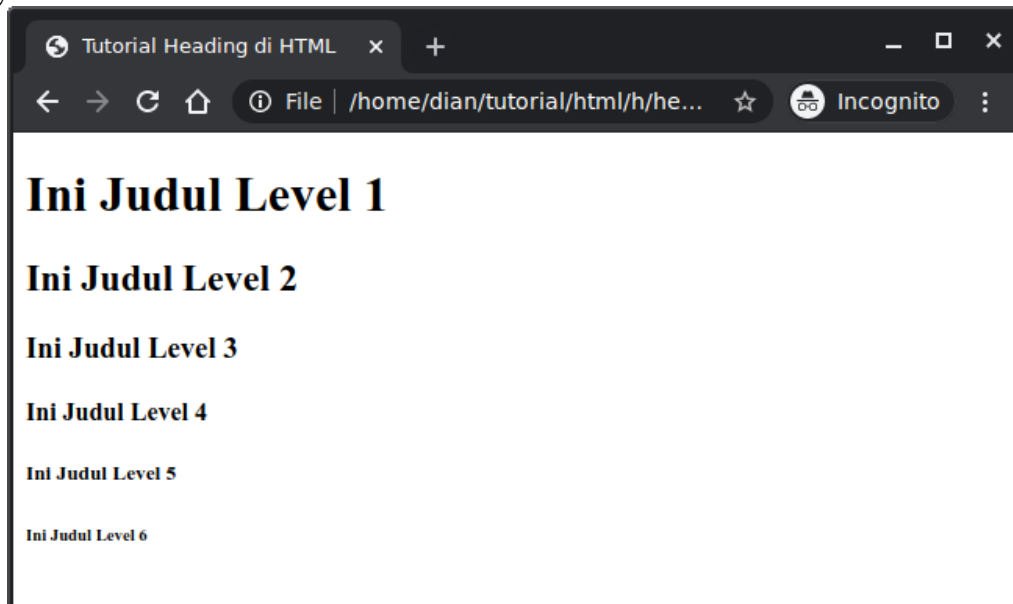
    <h5>Ini Judul Level 5</h5>

    <h6>Ini Judul Level 6</h6>

  </body>

</html>
```

Hasilnya:



Tag `<h1>` biasanya dipakai pada judul artikel. Lalu tag `<h2>`, `<h3>`, `<h4>`, `<h5>`, dan `<h6>` dipakai pada sub judul atau sub heading.

Mari kita coba membuat sebuah artikel yang dilengkapi dengan heading. Buatlah file HTML baru kemudian isi dengan kode berikut:

```
<!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:

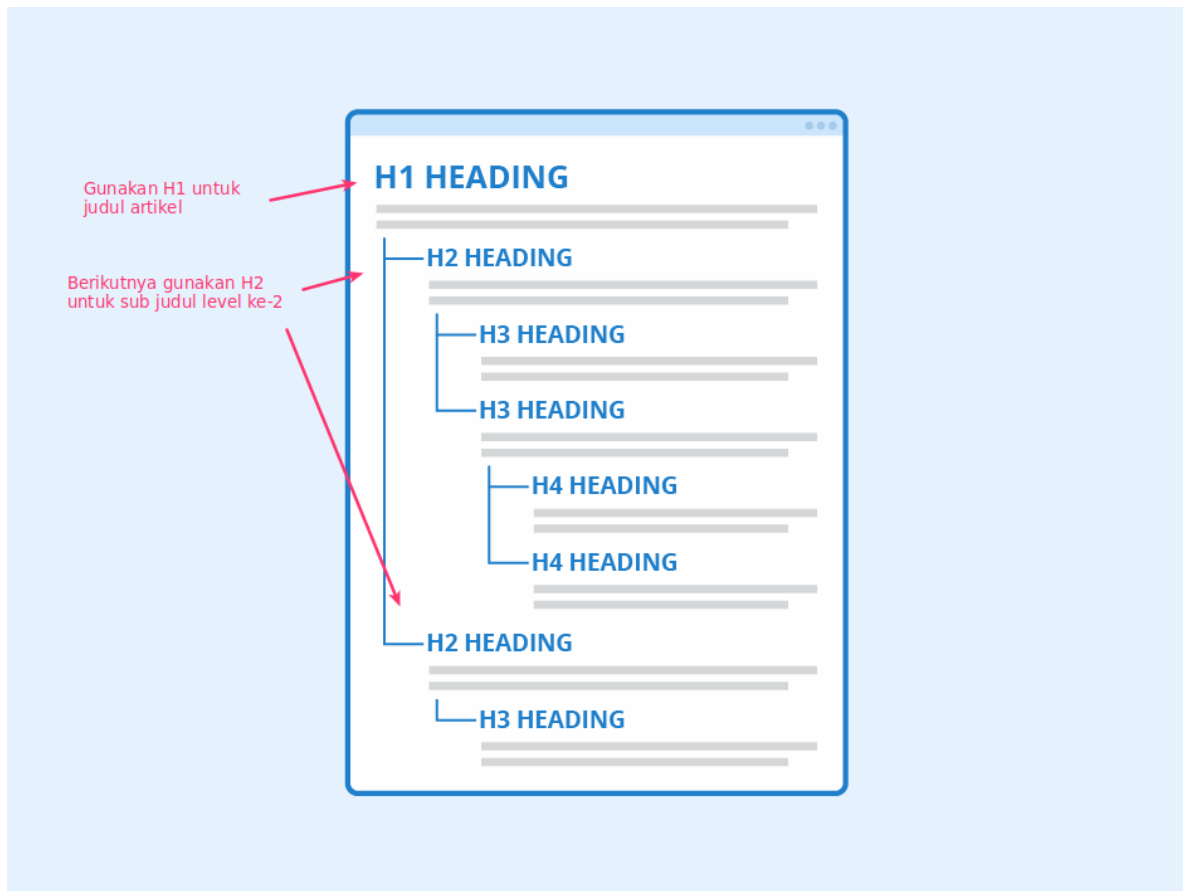


Berdasarkan pengalaman saya—dalam menulis artikel—tag `<h5>` dan `<h6>` jarang sekali dipakai, karena judul yang dibuat hanya sampai pada level 2, 3, dan 4.

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. ¹ Penulisan judul yang bagus adalah judul yang mengikuti levelnya.



Judul <h1> pada level pertama, <h2> pada level ke-2, dan seterusnya.

1.3. Atribut untuk Heading

Masih ingat tentang atribut?

Ya, dia adalah _____

Sudah lupa ya?

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](#).

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:



Sudah jelas kan bedanya?

Tag `<title>` untuk judul yang ditampilkan di web browser. Tag `<header>` sama seperti kop surat. Lalu heading `<h1>` sampai `<h6>` menjadi judul untuk artikel.

1.5. Heading Style

Sebenarnya ini termasuk dalam pembahasan materi tentang CSS. Karena itu, saya menjadikannya bonus yang boleh kamu baca dan juga boleh tidak. Jadi. Heading Style adalah style CSS yang diberikan pada heading agar terlihat menarik.

Berikut ini beberapa style yang sering digunakan pada heading:

1.5.1. Heading dengan Garis Bawah

Heading dengan garis bawah bisa kita buat dengan memanfaatkan tag `<hr>` dan juga CSS. Contoh menggunakan tag `<hr>`:

```
<!DOCTYPE html>
```

```
<html lang="en">

<head>

  <title>Tutorial Heading di HTML</title>

</head>

<body>

  <h1>Tutorial Membuat Heading di HTML</h1>

  <hr />

  <p>

    Heading adalah judul sebuah artikel dan bagian dari artikel.

    Ada Enam level heading pada HTML, yakni dimulai dari H1,

    H2, H3, sampai H6.

  </p>

</body>

</html>
```

Hasilnya:



Contoh menggunakan CSS:

```
<!DOCTYPE html>

<html lang="en">
```

```
<head>
```

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

```
</head>
```

```
<body>
```

```
  <h1 style="text-decoration: underline;">Tutorial Membuat Heading di HTML</h1>
```

```
  <p>
```

Heading adalah judul sebuah artikel dan bagian dari artikel.

Ada Enam level heading pada HTML, yakni dimulai dari H1,

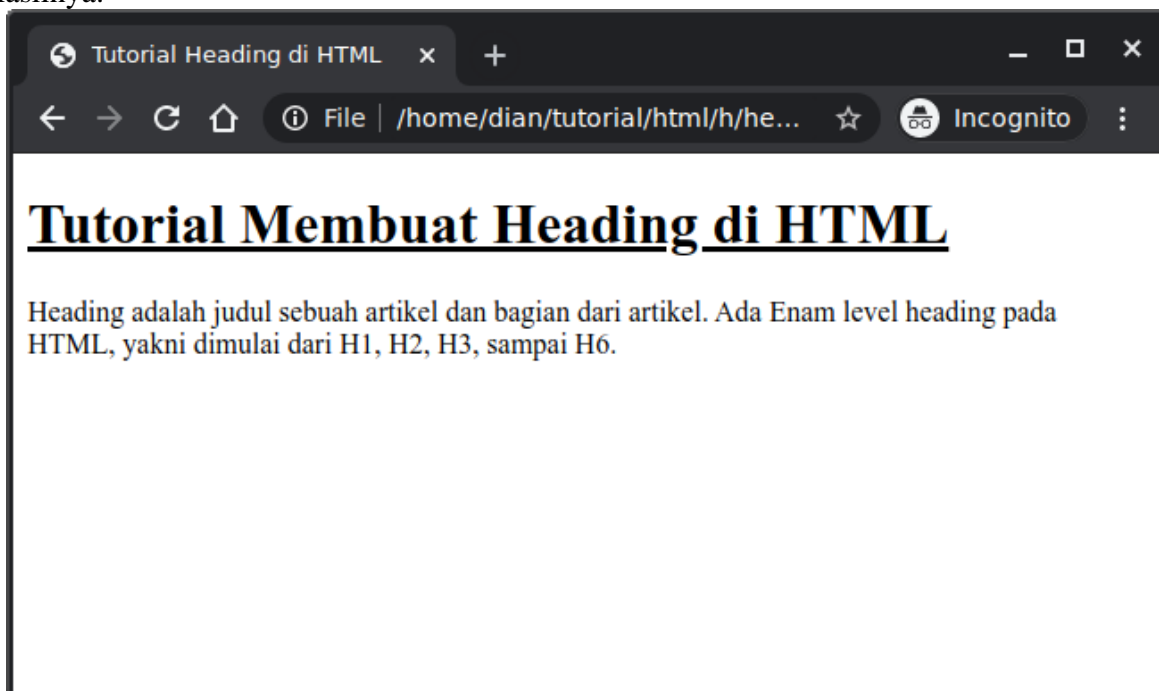
H2, H3, sampai H6.

```
</p>
```

```
</body>
```

```
</html>
```

Hasilnya:



1.5.2. Text Case untuk Heading

Heading kadang ditulis dengan berbagai style case. Ada yang menggunakan huruf besar semua, ada juga yang menggunakan huruf besar di awal kata saja.

Contoh:

INI HEADING DENGAN HURUF BESAR SEMUA

Ini Heading Dengan Huruf Besar Di Depan Saja

Nah, untuk membuat style case heading ini, Kita bisa menggunakan CSS dengan atribut text-transform.

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

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

```
</head>
```

```
<body>
```

```
<h1 style="text-transform: uppercase;">Heading dengan huruf Besar semua</h1>
```

```
<p>
```

Heading adalah judul sebuah artikel dan bagian dari artikel.

Ada Enam level heading pada HTML, yakni dimulai dari H1,

H2, H3, sampai H6.

```
</p>
```

```
<h1 style="text-transform: capitalize;">Heading dengan huruf Besar di awal kata</h1>
```

```
<p>
```

Properti text-transform berfungsi untuk menentukan style case dari

teks. Properti ini bisa diberikan nilai capitalize, uppercase,

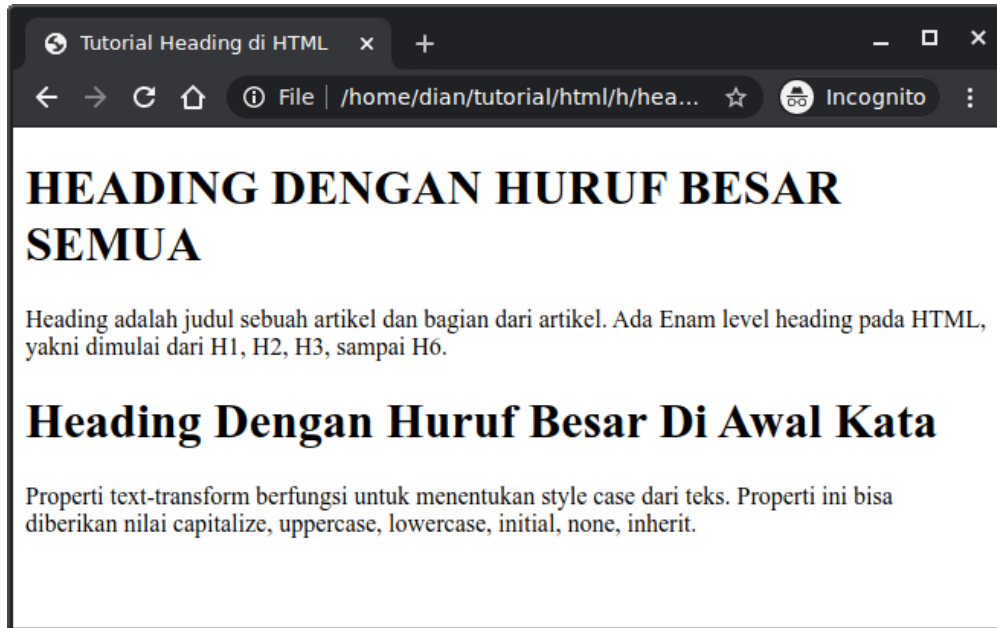
lowercase, initial, none, inherit.

</p>

</body>

</html>

Hasilnya:



1.5.3. Warna untuk Heading

Sama seperti elemen yang lainnya, heading juga dapat kita berikan warna dengan bantuan CSS. Warna bisa kita berikan pada teks dan latar belakang atau background.

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

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

```
</head>
```

```
<body>
```

```
<h1 style="color: red;">Heading dengan warna Merah</h1>
```

```
<p>
```

Atribut color adalah atribut CSS yang berfungsi untuk memberikan warna pada teks. Pada contoh ini, teks pada heading akan berwarna merah, karena kita memberikan warna red untuk color.

</p>

<h2 style="background-color: yellow;">Heading dengan Latar Kuning</h2>

<p>

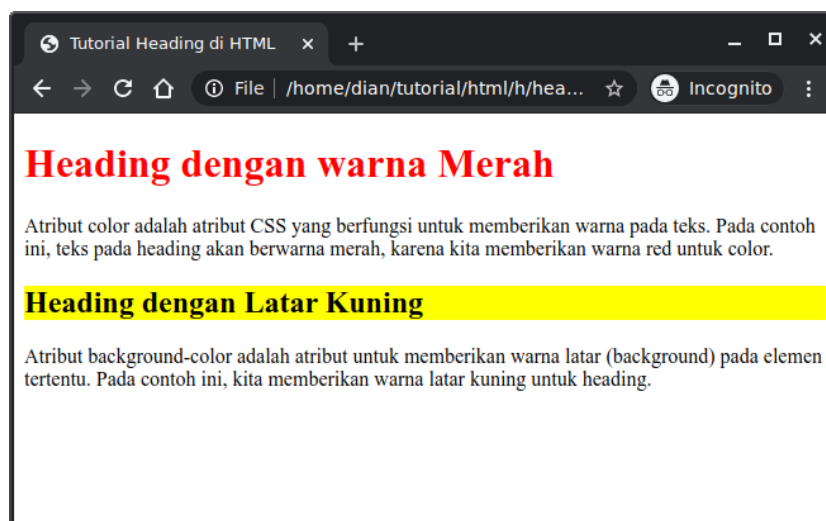
Atribut background-color adalah atribut untuk memberikan warna latar (background) pada elemen tertentu. Pada contoh ini, kita memberikan warna latar kuning untuk heading.

</p>

</body>

</html>

Hasilnya:



Nah untuk style lainnya, silahkan berkreasi sendiri.

1.6. Kesimpulan (1)

Kita sudah belajar tentang cara membuat heading di HTML. Hal yang perlu kamu ingat adalah tag-tag untuk membuat heading, yakni dimulai dari <h1> sampai <h6>.

II. Komentar

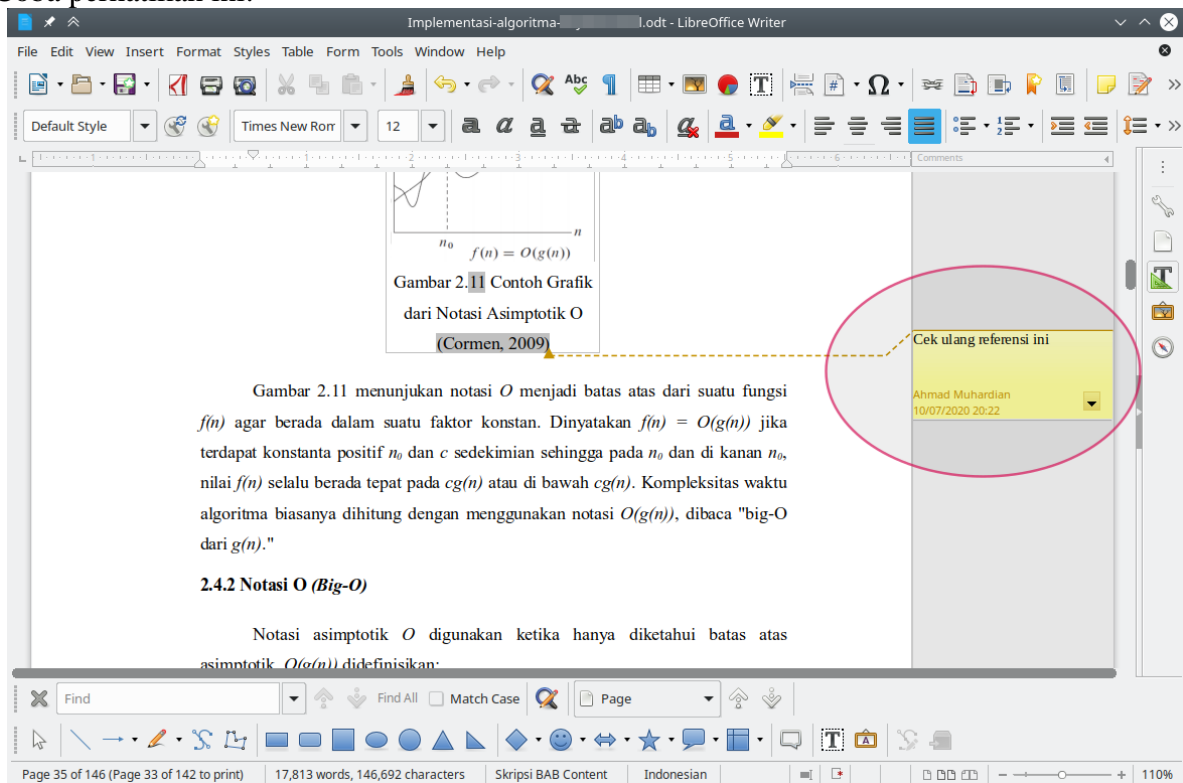
Sampai ditahap ini.. apakah kamu sudah ingat semua tag HTML yang dipelajari?

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 `<!-- -->`.



Contoh:

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <title>Belajar Membuat Komentar di HTML</title>

  </head>

  <body>

    <!-- ini adalah komentar -->

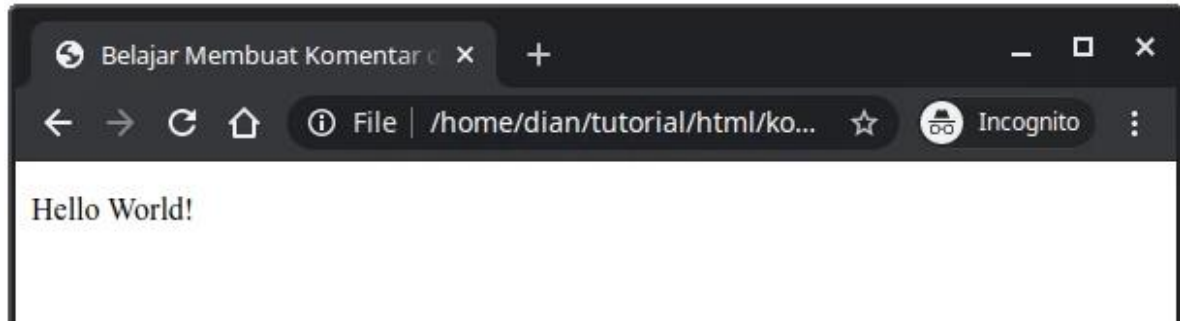
    <p>Hello World!</p>

    <!-- ini juga komentar
    dan ditulis dalam dua baris -->

  </body>

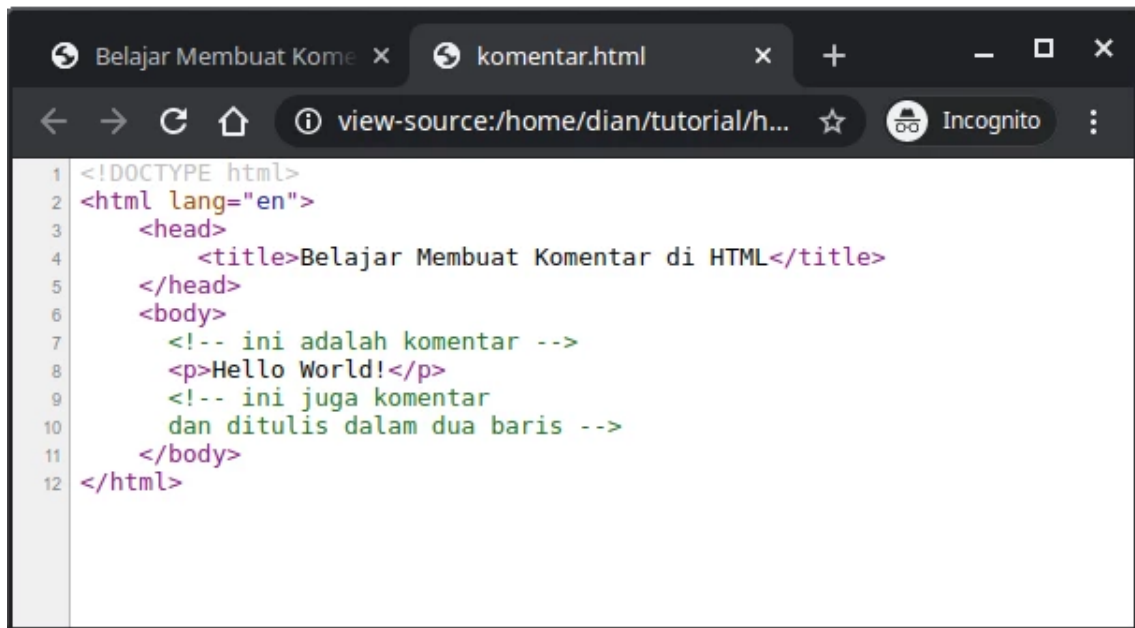
</html>
```

Hasilnya:



Komentar tidak akan ditampilkan oleh browser, tapi kita bisa melihatnya dengan cara view source.

Klik kanan pada browser, lalu pilih **view page source**.



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.

2.4. Atribut untuk Komentar

Komentar tidak bisa ditambahkan atribut. Jika kamu mencoba menambahkannya, itu akan sia-sia. Soalnya komentar merupakan elemen yang tidak akan diproses oleh browser.

2.5. Fungsi Komentar

Komentar memang tidak akan ditampilkan di web, namun bukan berarti ia tidak memiliki fungsi. Berikut ini beberapa fungsi komentar:

2.5.1. Komentar untuk Menjelaskan Arti Tag

Saat belajar HTML, kamu mungkin akan kesulitan mengingat fungsi dan arti dari tag HTML. Karena itu, kamu harus membuat catatan untuk mengingatnya. Ini bisa kamu lakukan dengan mencatat di buku, maupun media lain. Tapi, alangkah baiknya memanfaatkan komentar untuk mencatat.

Contoh:

```
<!DOCTYPE html> <!-- ini tag untuk menentukan type dokumen -->
```

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

```
<head>
```

```
<title>Belajar Membuat Komentar di HTML</title>
```

```
<!-- tag title berfungsi untuk membuat judul web dan akan ditampilkan
```

```
pada title bar di browser -->
```

```
</head>

<body>

  <p>Ini tag paragraf yang aktif</p>

  <!--

    tag <p> adalah tag untuk membuat paragraf

  -->

</body>

</html>
```

Jika kita lupa, kita bisa membuka kembali kode tersebut. Tapi ingat, jangan buat komentar seperti pada web yang akan kamu upload ke server atau produksi.

2.5.2. Komentar untuk Menyimpan Todo List

Komentar kadang sering digunakan juga untuk menyimpan todo list, ini akan membantu kita mengingat apa yang harus dikerjakan pada bagian HTML tertentu.

Contoh:

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <title>Belajar Membuat Komentar di HTML</title>

    <!-- TODO: Tambahkan tag meta di sini -->

  </head>

  <body>

    <h1>Komentar di HTML</h1>
```



```
<!-- TODO: Buat konten web di ini -->
```

```
</body>
```

```
</html>
```

Jika kita sudah mengerjakan apa yang dituliskan di todo list tersebut, kita bisa hapus komentarnya.

2.5.3. Komentar untuk Menonaktifkan Kode HTML

Komentar kadang seding digunakan untuk menon-aktifkan kode HTML.

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<title>Belajar Membuat Komentar di HTML</title>
```

```
</head>
```

```
<body>
```

```
<p>Ini tag paragraf yang aktif</p>
```

```
<!--
```

```
<p>ini tag paragraf yang tidak aktif
```

```
karena berada di dalam komentar
```

```
</p>
```

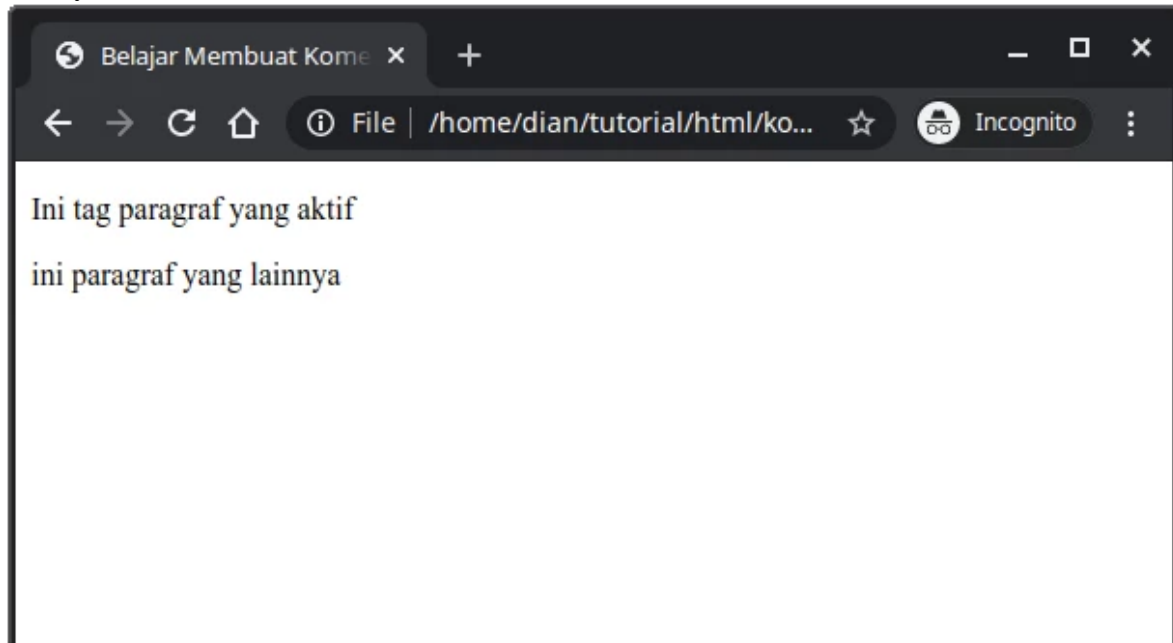
```
-->
```

```
<p>ini paragraf yang lainnya</p>
```

```
</body>
```

```
</html>
```

Hasilnya:



2.6. Kesimpulan (2)

Itulah cara membuat komentar di HTML, intinya kamu hanya perlu mengingat tag untuk membuatnya.

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> , bold dan <u>underline</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*.

Apa saja itu? Mari kita bahas..

3.1. Membuat Teks Tebal

Contoh:

```
<!DOCTYPE html>
```

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 `` (*bold*) dan tag ``. Kamu bebas mau pakai yang mana saja, hasilnya akan sama-sama tebal.

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Text Tebal di HTML</title>
```

```
</head>
```

```
<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>
```

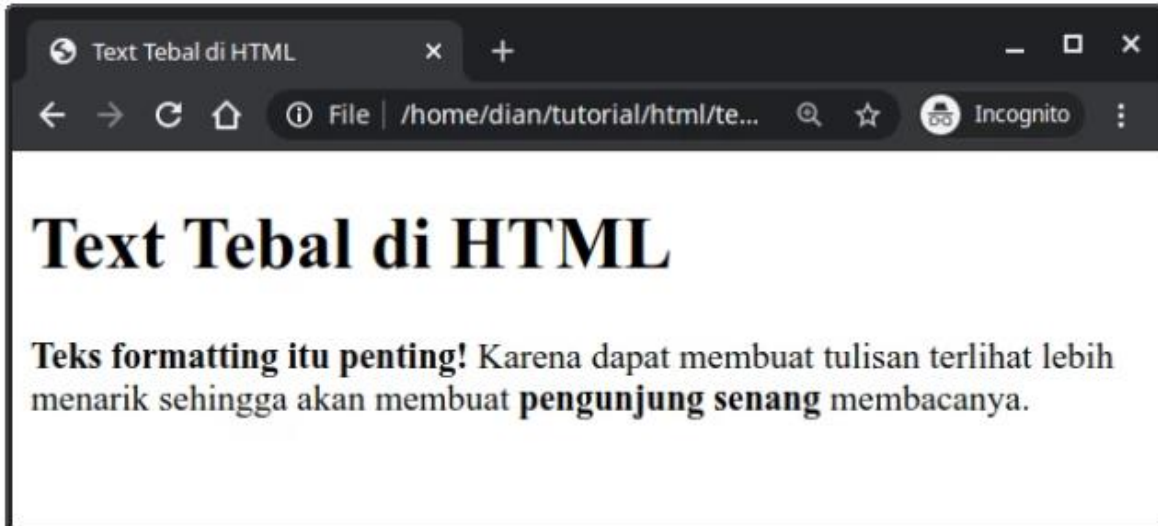
```
membacanya.
```

```
</p>
```

```
</body>
```

```
</html>
```

Hasilnya:



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 `` (emphasis).

Contoh:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Text Miring di HTML</title>

</head>

<body>

  <h1>Text Miring di HTML</h1>

  <p>
```

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

sehingga akan `menarik perhatian` pembaca. Biasanya

digunakan pada *istilah asing* atau kata serapan dari

bahasa daerah.

Hasilnya:



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>` (underlaine) atau juga tag `<ins>` (insert). Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

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

```
</head>

<body>

  <h1>Garis Bawah di HTML</h1>

  <p>

    <u>Text formatting itu penting</u>, karena dapat membuat teks terlihat

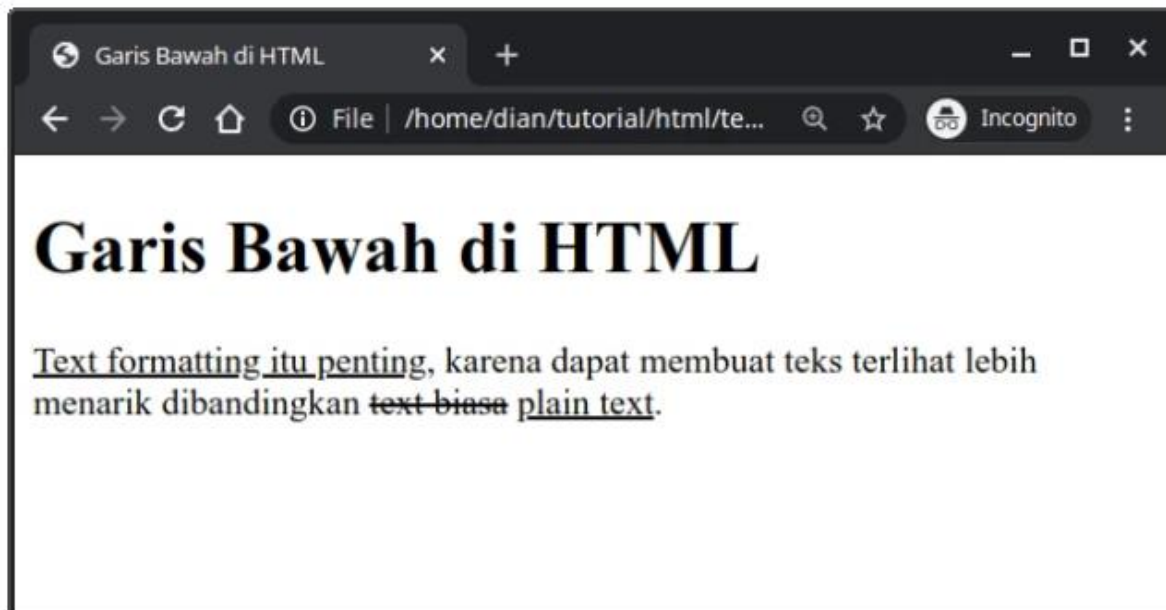
    lebih menarik dibandingkan <del>text biasa</del> <ins>plain text</ins>.

  </p>

</body>

</html>
```

Hasilnya:



Pada contoh di atas, kita menggunakan tag `` untuk membuat teks tercoret. Lalu diikuti dengan teks yang ditambahkan (*insert*).

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 `` (*delete*).

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

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

```
</head>
```

```
<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 tercoret

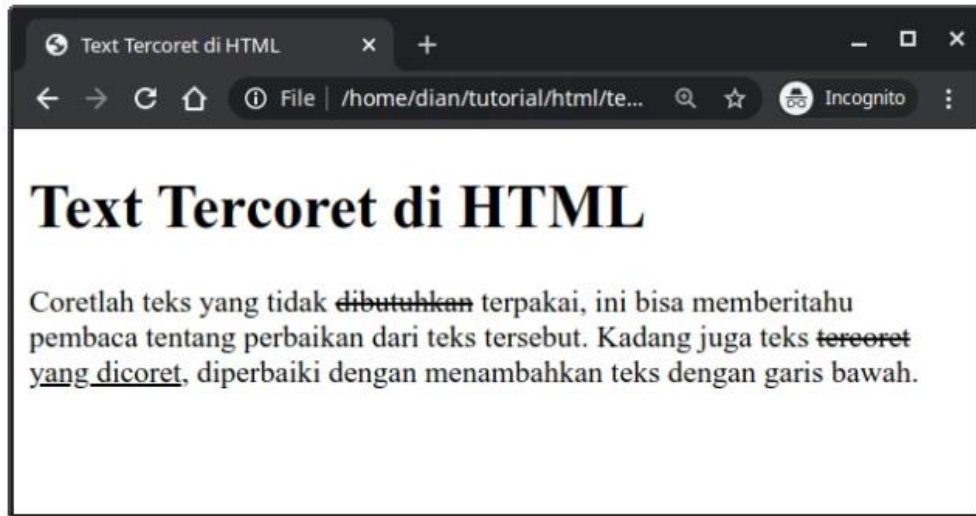
<ins>yang dicoret</ins>, diperbaiki dengan menambahkan teks dengan garis bawah.

```
</p>
```

```
</body>
```

```
</html>
```

Hasilnya:



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

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Membuat Pangkat di HTML</title>
```

```
</head>
```

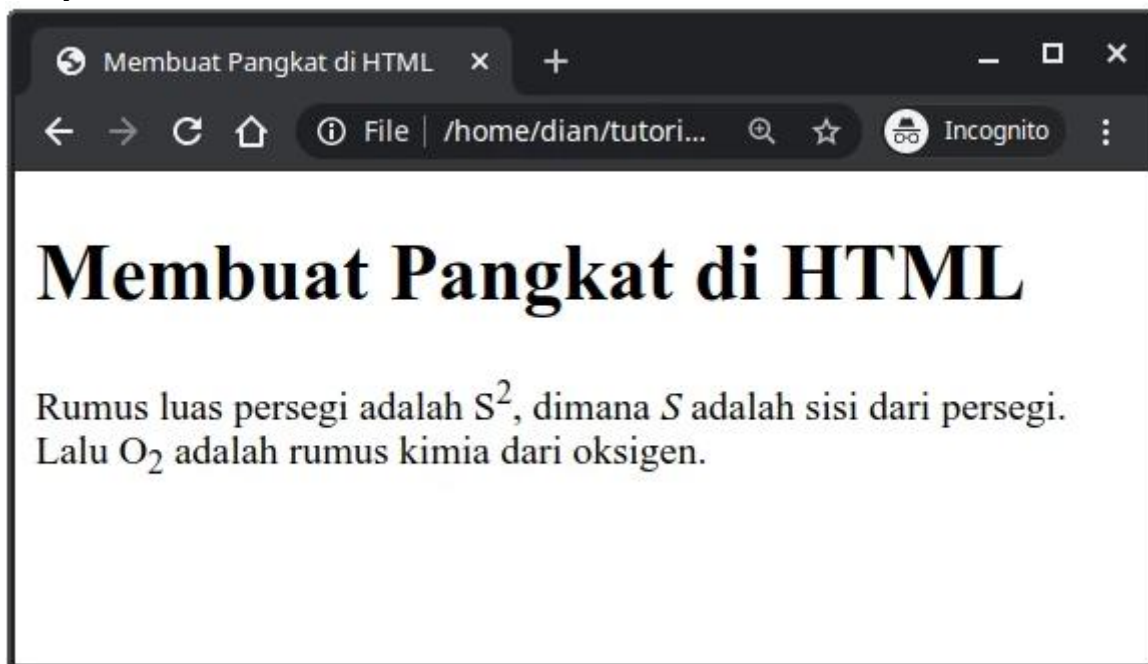
```
<body>
```

```
<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.

Hasilnya:



3.6. Membuat Marker untuk Teks

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

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

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

```
</head>
```

```
<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>
```

```
</html>
```

Hasilnya:



Warna default marker adalah kuning. Warna ini bisa kita ubah dengan style CSS.

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

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

```
</head>
```

```
<body>
```

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

```
<p>
```

Marker biasanya digunakan untuk menandai bagian teks yang penting.

Kalau di dunia nyata, kita `<mark style="background-color: pink">menggunakan
stabilo</mark>` untuk

membuat marker.

```
</p>
```

```
</body>
```

```
</html>
```

Maka hasilnya, marker akan berwarna pink:



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.

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Text Formatting untuk Teks dari komputer</title>
```

</head>

<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

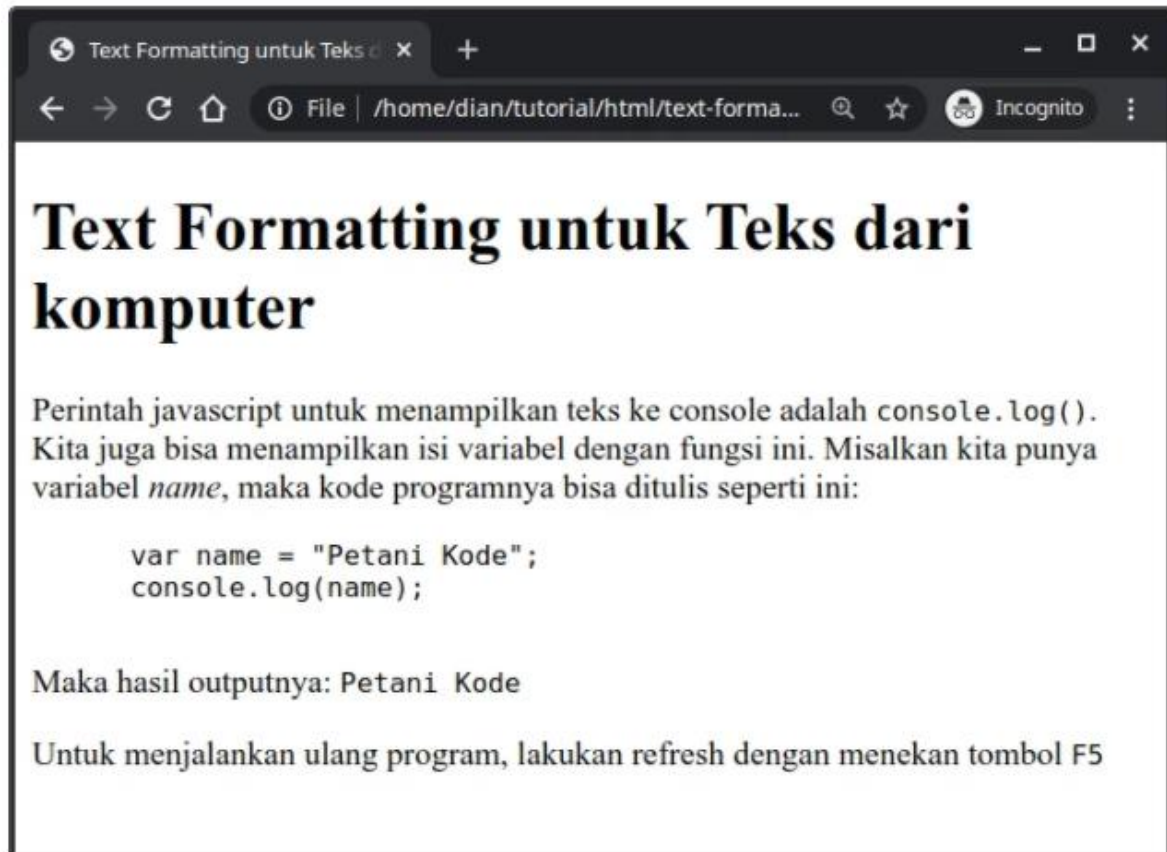
<kbd>F5</kbd>

</p>

</body>

</html>

Hasilnya:



3.8. Menggabungkan Format

Apakah format teks dapat digabungkan?

Misalkan kita 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 `` dan `<u>`.

Contoh:

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Penggabungan Format Teks di HTML</title>
```

```
</head>
```

```
<body>
```

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

```
<p>
```

Penggabungan format teks bisa dilakukan dengan menuliskan tag-tag yang

akan dipakai. Misalkan `<u>tebal dan garis bawah</u>`, maka kita

tinggal pakai tag `b` dan tag `u`.

```
</p>
```

```
</body>
```

```
</html>
```

Hasilnya:



Dalam menggabungkan format, kamu perlu memperhatikan tag mana yang ditulis duluan dan yang terakhir.

Jangan sampai salah menutup..

Yang dibuka duluan, harus ditutup terakhir.

Ingatlah konsep “ibu memasak ubi”:



3.9. Kesimpulan (3)

Teks formatting akan sering kita pakai dalam membuat konten di web, pastikan kamu mengingat tag-tag yang digunakan untuk teks formatting.

3. Contoh Soal dan Pengerjaan/Studi Kasus

4. Soal Mandiri

5. REFERENSI/DAFTAR PUSTAKA

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., Vadiello, 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.