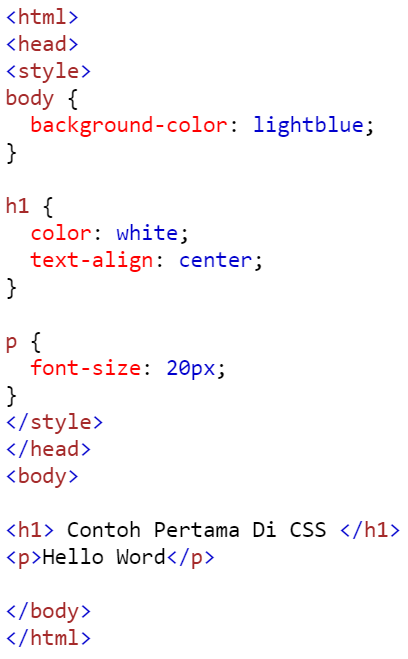
**Tugas Resume PWD di Web W3schools**

**CSS HOME :**

CSS adalah bahasa yang digunakan untuk menata gaya dokumen di HTML.

CSS menjelaskan bagaimana elemen HTML harus ditampilkan.

Contoh :

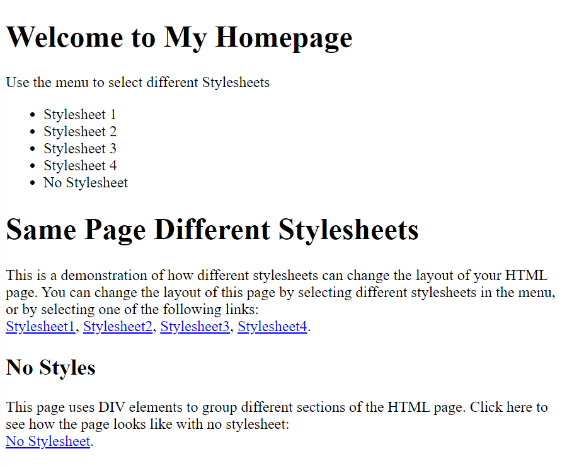


**CSS INTRODUCTION :**

Apa itu CSS ?

* CSS adalah singkatan dari Cascading Style Sheet.
* CSS menjelaskan bagaimana elemen HTML ditampilkan di layar, kertas, atau di media lain.
* CSS menghemat banyak pekerjaan. Itu dapat mengontrol tata letak beberapa halaman web sekaligus.
* Stylesheet eksternal disimpan dalam file CSS.

Contoh Style Sheet :

<- Tampilan tanpa CSS

Tampilan menggunakan CSS ->

Kenapa CSS ?

Karena CSS digunakan untuk menentukan gaya halaman web, termasuk desain, tata letak, dan variasi tampilan untuk berbagai perangkat dan ukuran layar.

**CSS SYNTAX :**



* Selector ke elemen HTML yang ingin menentukan gaya halaman.
* Blok deklarasi berisi satu atau lebih deklarasi yang dipisahkan oleh titik koma.
* Setiap deklarasi menyertakan nama properti CSS dan nilainya, dipisahkan oleh titik dua.
* Beberapa deklarasi CSS dipisahkan dengan titik koma, dan blok deklarasi diapit oleh kurung kurawal.

Contoh :

Penjelasan :

* **p** adalah pemilih dalam CSS (menunjuk ke elemen HTML yang ingin Anda gaya: <p>).
* **Color** adalah properti, dan **Green** adalah nilai properti
* **text-align** adalah properti, dan **center** adalah nilai property

**CSS SELECTORS :**

Pemilih CSS digunakan untuk "menemukan" (atau memilih) elemen HTML yang ingin Menentukan gaya.

CSS menjadi lima kategori:

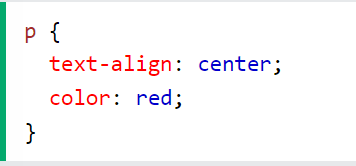
* ***Simple Selectors*** (pilih elemen berdasarkan nama, id, kelas)
* ***Combinator Selector*** (memilih elemen berdasarkan hubungan tertentu di antara elemen tersebut)
* ***Pseudo- Class Selector*** (memilih elemen berdasarkan keadaan tertentu)
* ***Pseudo- Element Selectors*** (memilih dan memberi gaya pada bagian elemen)
* ***Attribute Selector*** (memilih elemen berdasarkan atribut atau nilai atribut)

**CSS ELEMENT SELECTOR :**

Pemilih elemen memilih elemen HTML berdasarkan nama elemen.

Contoh :

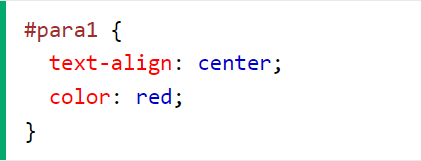
* Elemen <p> pada halaman akan center-aligned, dengan warna teks merah.



**CSS ID SELECTOR :**

* Pemilih id menggunakan atribut id dari elemen HTML untuk memilih elemen tertentu.
* Id suatu elemen bersifat unik dalam suatu halaman, sehingga pemilih id digunakan untuk memilih satu elemen unik!
* Untuk memilih elemen dengan id tertentu, tulis karakter hash (#), diikuti dengan id elemen tersebut.

Contoh :

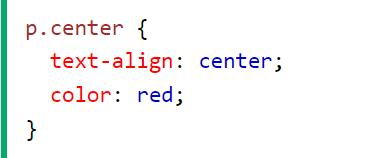


**CSS CLASS SELECTOR** :

* Pemilih Class Selector elemen HTML dengan atribut kelas tertentu.
* Untuk memilih elemen dengan kelas tertentu, tuliskan karakter titik (.), diikuti dengan nama kelas.

Contoh :

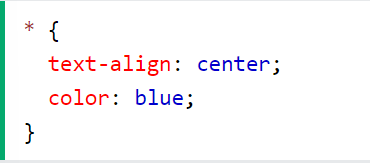
Dalam contoh ini hanya elemen <p> dengan class="center" yang akan berwarna merah dan rata tengah:



**CSS UNIVERSAL SELECTOR :**

Universal Selector (\*) memilih semua elemen HTML pada halaman.

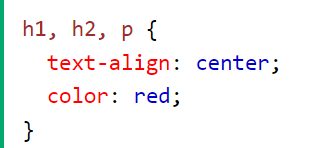
Contoh :



**CSS GROUPING SELECTOR**

Pemilih pengelompokan memilih semua elemen HTML dengan definisi gaya yang sama.

Misalkan kode CSS berikut (elemen h1, h2, dan p memiliki definisi gaya yang sama) Maka :



**CSS HOW TO :**

Saat browser membaca style sheet, browser akan memformat dokumen HTML sesuai dengan informasi di style sheet.

Tiga Cara Memasukkan CSS

Ada tiga cara menyisipkan style sheet:

* **External CSS**
* **Internal CSS**
* **Inline CSS**

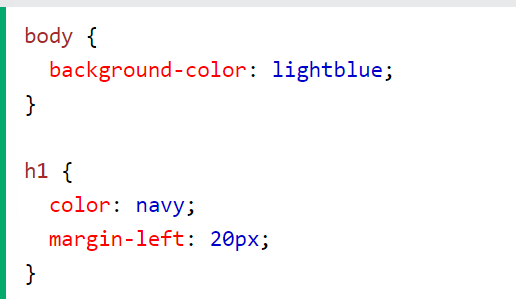
**EXTERNAL CSS :** Setiap halaman HTML harus menyertakan referensi ke file style sheet eksternal di dalam elemen <link>, di dalam bagian head.

Contoh :

External Style ditentukan dalam elemen <link>, di dalam bagian <head> halaman HTML



“mystyle.css”



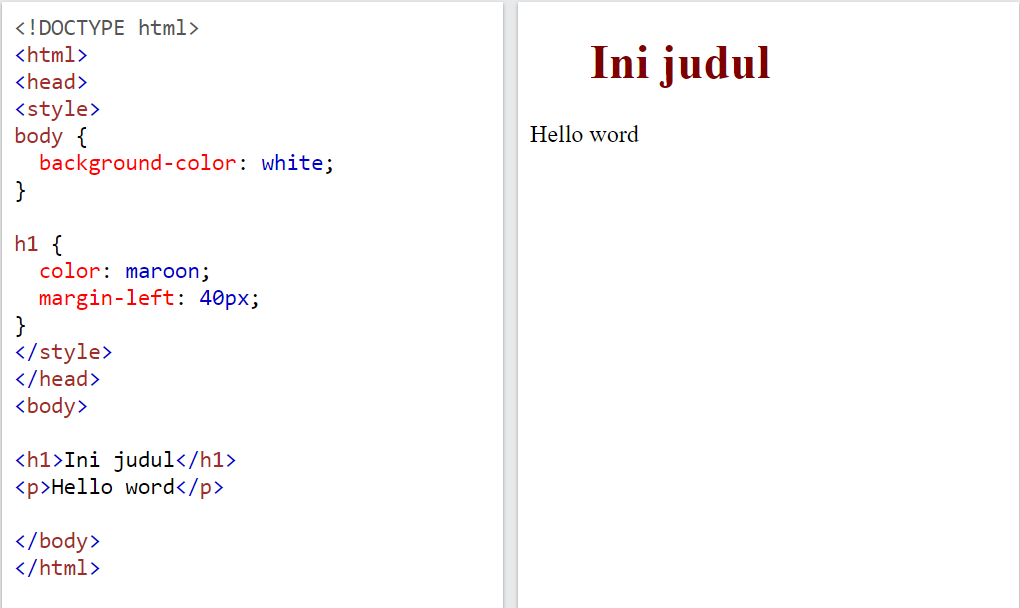
**INTERNAL CSS :**

Internal Style dapat digunakan jika satu halaman HTML memiliki gaya yang unik.

Gaya internal didefinisikan di dalam elemen <style>, di dalam bagian head.

Contoh :

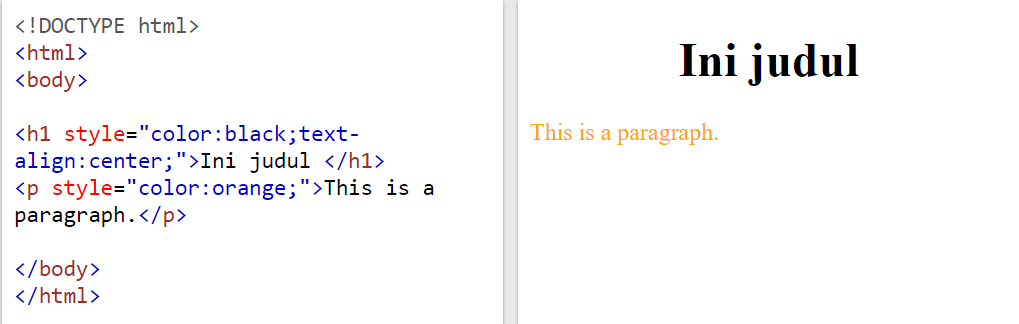
Internal Styles didefinisikan dalam elemen <style>, di dalam bagian <head> halaman HTML:



**INLINE CSS :**

Inline style dapat digunakan untuk menerapkan gaya unik pada satu elemen.

Untuk menggunakan Inline Style, tambahkan atribut style ke elemen yang relevan. Atribut style dapat berisi properti CSS apa pun.



**Multiple Style Sheets**

Jika beberapa properti telah ditentukan untuk pemilih (elemen) yang sama dalam style sheet yang berbeda, nilai dari style sheet yang terakhir dibaca akan digunakan.

**Cascading Order**

Gaya apa yang akan digunakan ketika ada lebih dari satu gaya yang ditentukan untuk sebuah elemen HTML?

Semua gaya dalam halaman akan "mengalir" ke dalam lembar gaya "virtual" baru dengan aturan berikut, di mana nomor satu memiliki prioritas tertinggi:

1. **Inline style** (di dalam elemen HTML)
2. **Style sheet eksternal dan internal** (di bagian kepala)
3. **Browser default**

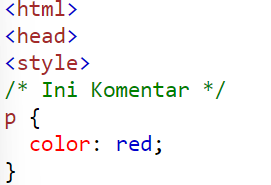
Jadi, gaya sebaris memiliki prioritas tertinggi, dan akan menggantikan gaya eksternal dan internal serta default browser.

**CSS COMMENTS :**

Comments digunakan untuk menjelaskan kode, dan mungkin membantu ketika mengedit kode sumber di kemudian hari. Jadi kolom komentar tidak akan dieksekusi oleh program.

Komentar CSS ditempatkan di dalam elemen <style>, dan dimulai dengan /\* dan diakhiri dengan \*/:

Contoh :



**CSS COLORS :**

Colors ditentukan menggunakan nama warna yang telah ditentukan sebelumnya, atau nilai RGB, HEX, HSL, RGBA, HSLA.

Warna Terbagi menjadi tiga:

* **RGB** (Dalam CSS, suatu warna dapat ditentukan sebagai nilai RGB)
* **HEX** (Di CSS, warna dapat ditentukan menggunakan nilai heksadesimal dalam formular)
* **HSL** (Dalam CSS, suatu warna dapat ditentukan menggunakan hue, saturation, dan lightness (HSL))

**CSS COLOR NAMES :**

Di CSS, warna dapat ditentukan dengan menggunakan nama warna yang telah ditentukan sebelumnya:



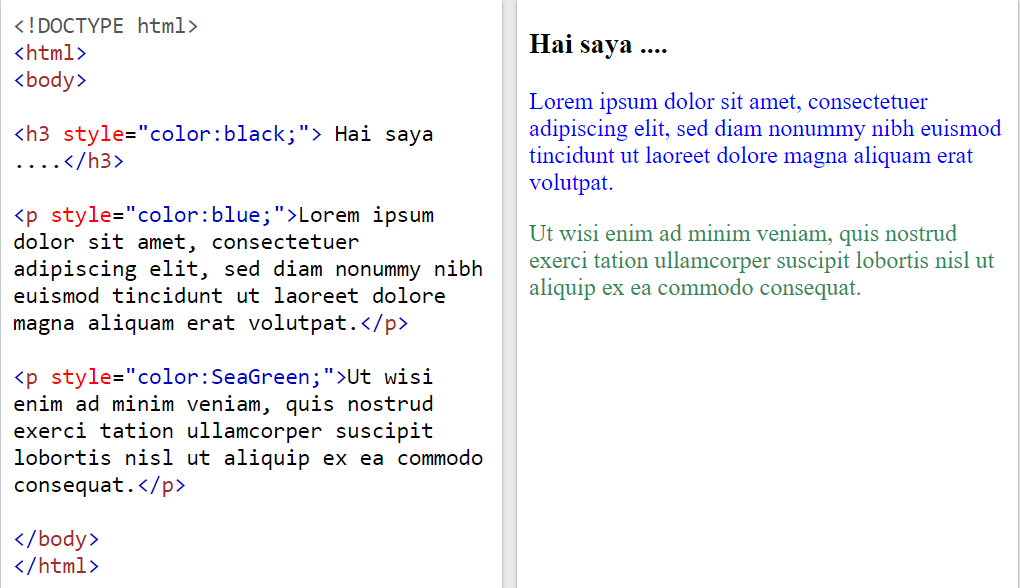
**CSS BACKGROUND COLOR:**

Mewarnai latar belakang, contoh :



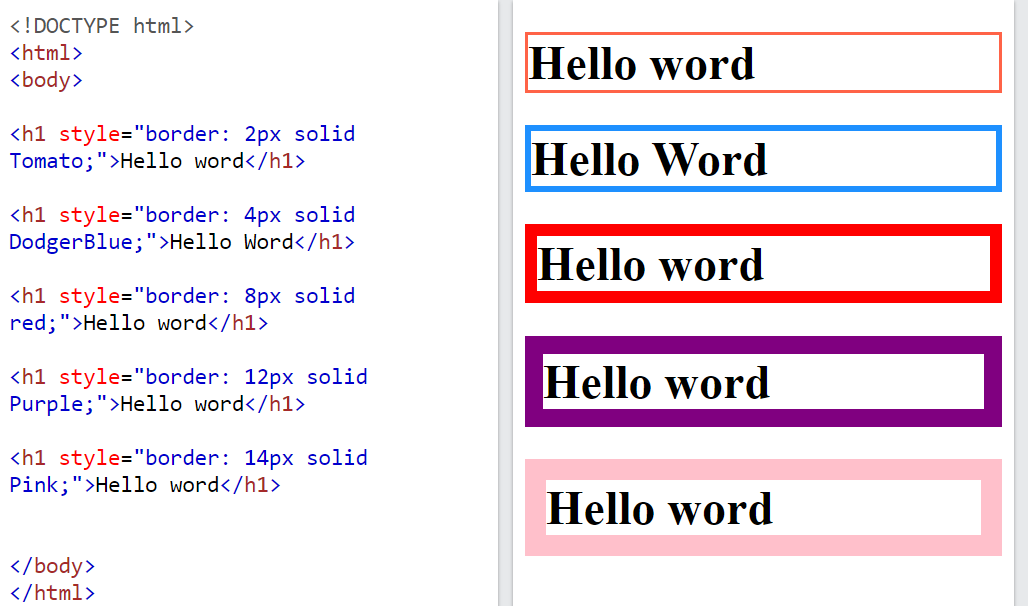
**CSS TEXT COLOR :**

Mewarnai text, contoh :



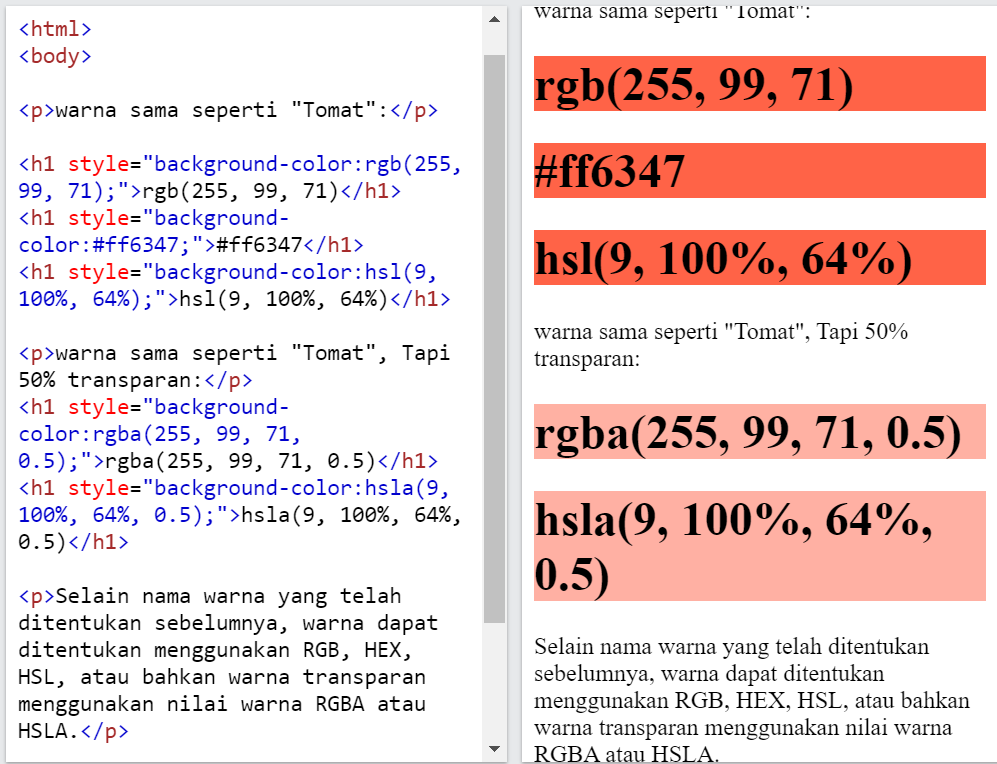
**CSS BORDER COLOR :**

Mewarnai border, contoh:



**CSS COLOR VALUES :**

Dalam CSS, warna juga dapat ditentukan menggunakan nilai RGB, nilai HEX, nilai HSL, nilai RGBA, dan nilai HSLA:



**CSS BACKGROUNDS :**

BACKGROUND COLOR : Properti background-color menentukan warna latar belakang suatu elemen.

Contoh :



**BACKGROUNDS IMAGE :** Properti background-image menentukan gambar yang akan digunakan sebagai latar belakang suatu elemen. Secara default, gambar diulang sehingga menutupi keseluruhan elemen.

Contoh :



**BACKGROUND REPEAT :** Secara default, properti background-image mengulang gambar secara horizontal dan vertikal. Beberapa gambar sebaiknya diulang hanya secara horizontal atau vertikal, atau gambar tersebut akan terlihat aneh.

Jika gambar hanya diulang secara horizontal (background-repeat: repeat-x;), maka background akan terlihat lebih baik.

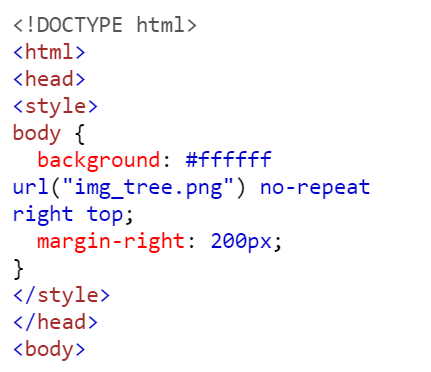
Contoh ;



**BACKGROUND ATTACHMENT :** Properti background-attachment menentukan apakah gambar latar belakang harus di-scroll atau diperbaiki. Jadi untuk background image yang sudah diperbaiki kalau akan di scroll background akan tetap ada. Sedangkan, harus di scroll background akan hilang setelah di scroll.

**BACKGROUND SHORTHAND PROPERTY :** Untuk mempersingkat kode, dimungkinkan juga untuk menentukan semua properti latar belakang dalam satu properti. Ini disebut shorthand property.

Contoh :



**CSS BORDER**

Properti CSS Border memungkinkan untuk menentukan gaya, lebar, dan warna batas elemen.

Properti border-style menentukan jenis perbatasan yang akan ditampilkan.

Nilai-nilai berikut diperbolehkan:

* **Dotted** - Mendefinisikan batas dotted
* **Dashed** - Mendefinisikan batas dashed
* **solid** - Mendefinisikan batas yang solid
* **double** - Mendefinisikan batas double
* **groove** - Mendefinisikan batas groove 3D. Efeknya bergantung pada nilai warna batas
* **ridge** - Mendefinisikan batas ridge 3D. Efeknya bergantung pada nilai warna batas
* **inset** - Mendefinisikan batas inset 3D. Efeknya bergantung pada nilai warna batas
* **outset** - Mendefinisikan batas inset 3D. Efeknya bergantung pada nilai warna batas
* **none** - Tidak mendefinisikan batas
* **hidden** - Mendefinisikan batas hidden

Properti gaya perbatasan dapat memiliki satu hingga empat nilai (untuk batas atas, batas kanan, batas bawah, dan batas kiri).

Dalam CSS border terbagi menjadi 5 yaitu:

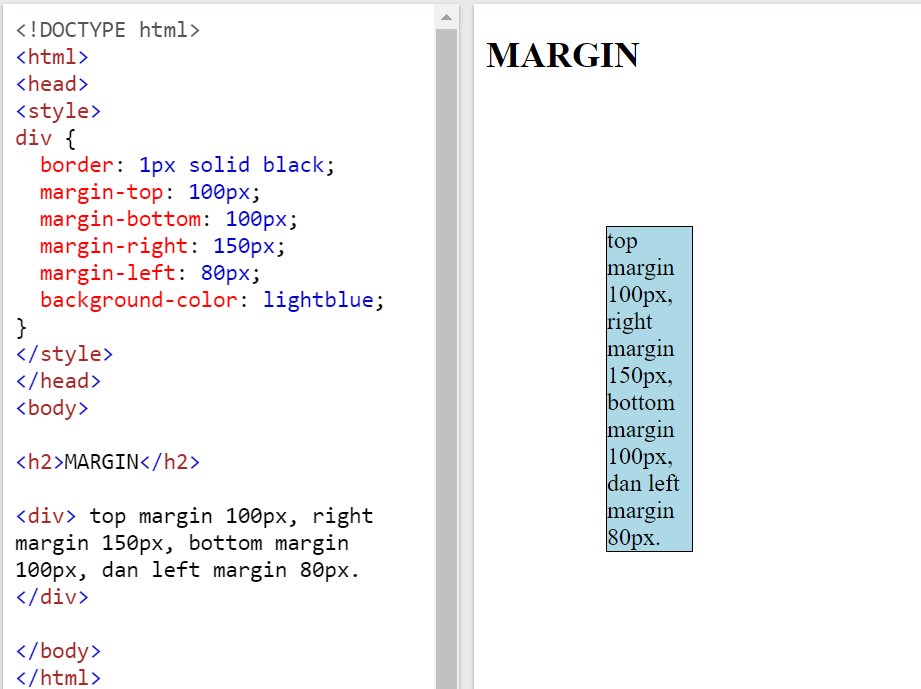
* **Border Width**
* **Border Color**
* **Border Shorthand**
* **Rounded Borders**

**CSS MARGINS**

Properti CSS MARGINS digunakan untuk membuat ruang di sekitar elemen, di luar batas yang ditentukan.

Dengan CSS, memiliki kendali penuh atas margin. Terdapat properti untuk mengatur margin untuk setiap sisi elemen (atas, kanan, bawah, dan kiri).

Contoh :

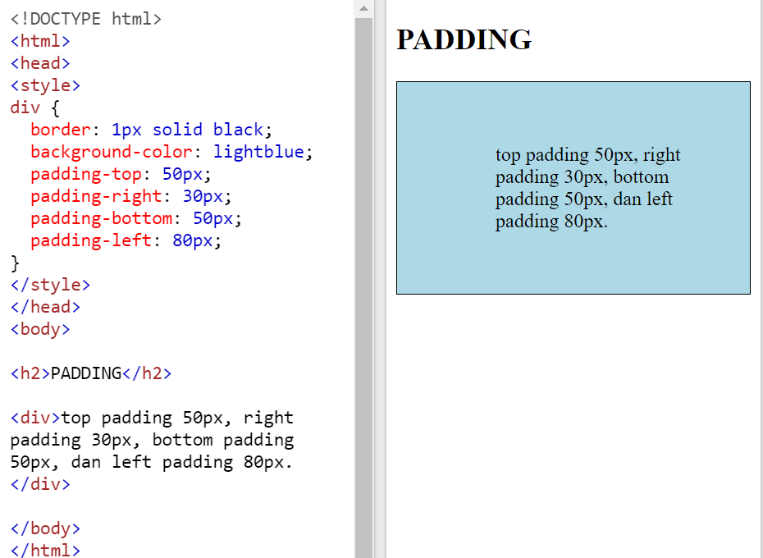


**CSS PADDING**

Properti CSS PADDING digunakan untuk menghasilkan ruang di sekitar konten elemen, di dalam batas yang ditentukan.

Dengan CSS, memiliki kendali penuh atas padding. Terdapat properti untuk mengatur padding untuk setiap sisi elemen (atas, kanan, bawah, dan kiri).

Contoh ;



**CSS SETTING HEIGHT DAN WIDHT**

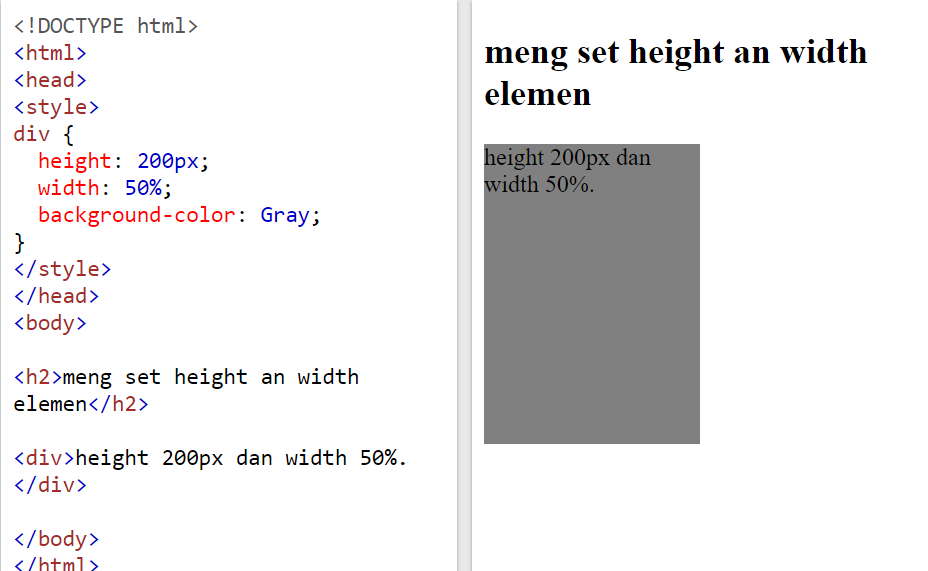
Properti HEIGHT dan WIDHT digunakan untuk mengatur tinggi dan lebar suatu elemen.

Properti HEIGHT dan WIDHT tidak menyertakan padding, batas, atau margin. Ini mengatur tinggi/lebar area di dalam padding, batas, dan margin elemen.

Properti tinggi dan lebar mungkin memiliki nilai berikut:

* **auto** - Ini adalah default. Browser menghitung tinggi dan lebar
* **length** - Mendefinisikan tinggi/lebar dalam px, cm, dll.
* **%** - Mendefinisikan tinggi/lebar dalam persen blok yang memuatnya
* **initial** - Menyetel tinggi/lebar ke nilai defaultnya
* **inherit** - Tinggi/lebar akan diwarisi dari nilai induknya

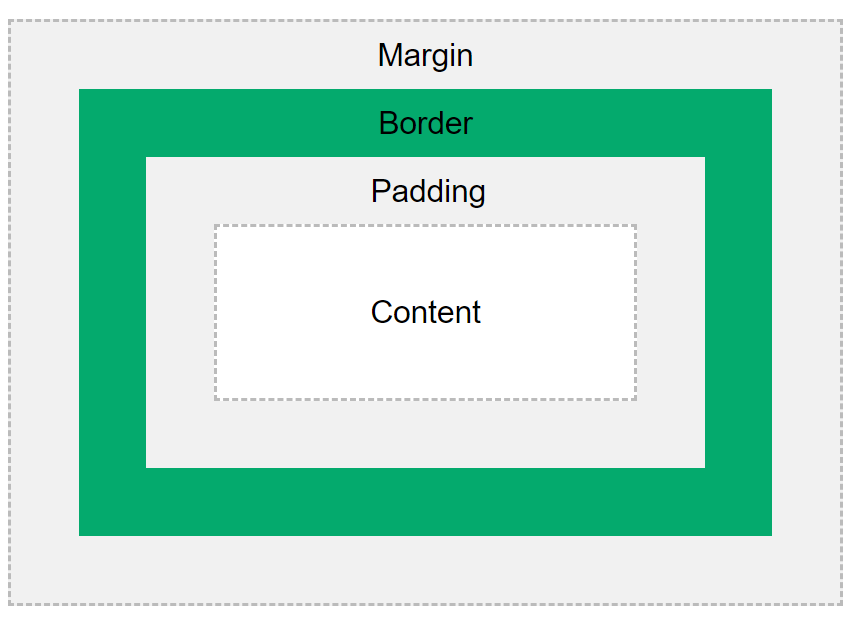
contoh :



**CSS BOX MODEL**

Dalam CSS, istilah "model kotak" digunakan ketika berbicara tentang desain dan tata letak.

CSS BOX MODEL pada dasarnya adalah sebuah kotak yang membungkus setiap elemen HTML. Terdiri dari: konten, padding, batas dan margin. Gambar di bawah mengilustrasikan model kotak:



Penjelasan dari berbagai bagian:

* **Content** - Isi kotak, tempat teks dan gambar muncul
* **Padding** - Menghapus area di sekitar konten. Bantalannya transparan
* **Border** - Perbatasan yang mengelilingi padding dan konten
* **Margin** - Membersihkan area di luar perbatasan. Marginnya transparan

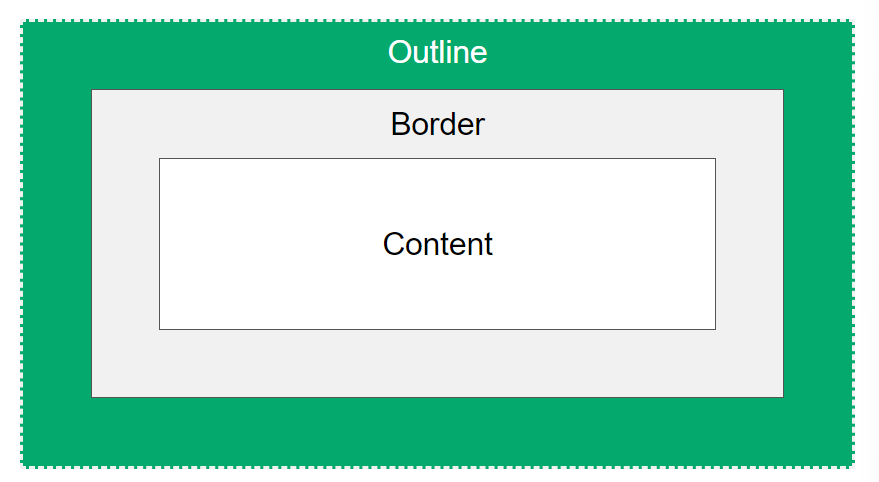
Box model memungkinkan kita menambahkan batas di sekitar elemen, dan menentukan ruang antar elemen.

**CSS OUTLINE**

Garis besar adalah garis yang digambar di sekeliling elemen, OUTSIDE, untuk membuat elemen "stand out".

Di CSS OUTLINE terbagi menjadi 4 yaitu:

* **Outline Width**
* **Outline Color**
* **Outline Shorthand**
* **Outline Offset**

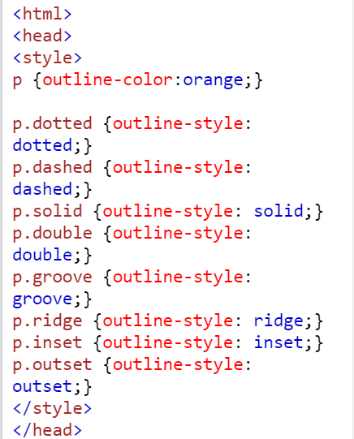


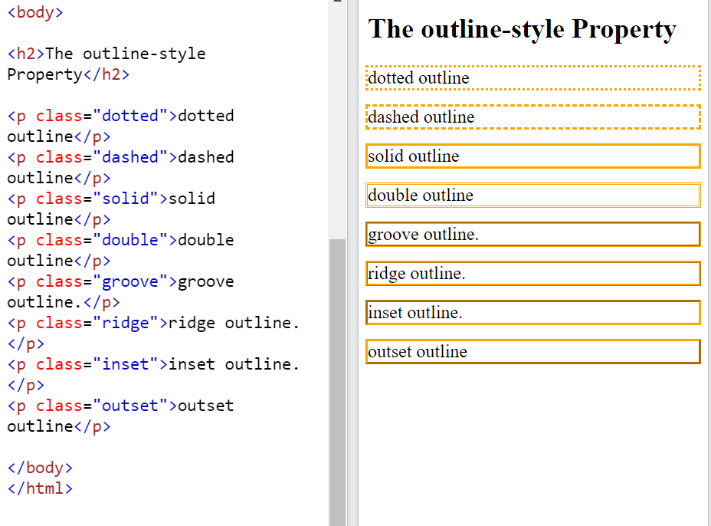
Properti CSS OUTLINE:

* **outline-style**
* **outline-color**
* **outline-width**
* **outline-offset**
* **outline**

Properti outline-style :

* **Dotted** - Mendefinisikan batas dotted outline
* **Dashed** - Mendefinisikan batas dashed outline
* **solid** - Mendefinisikan batas yang solid outline
* **double** - Mendefinisikan batas double outline
* **groove** - Mendefinisikan batas groove 3D outline
* **ridge** - Mendefinisikan batas ridge 3D outline
* **inset** - Mendefinisikan batas inset 3D outline
* **outset** - Mendefinisikan batas inset 3D outline
* **none** - Tidak mendefinisikan outline
* **hidden** - Mendefinisikan batas hidden outline

Contoh :



**CSS TEXT**

Properti COLOR digunakan untuk mengatur warna teks. Warnanya ditentukan oleh:

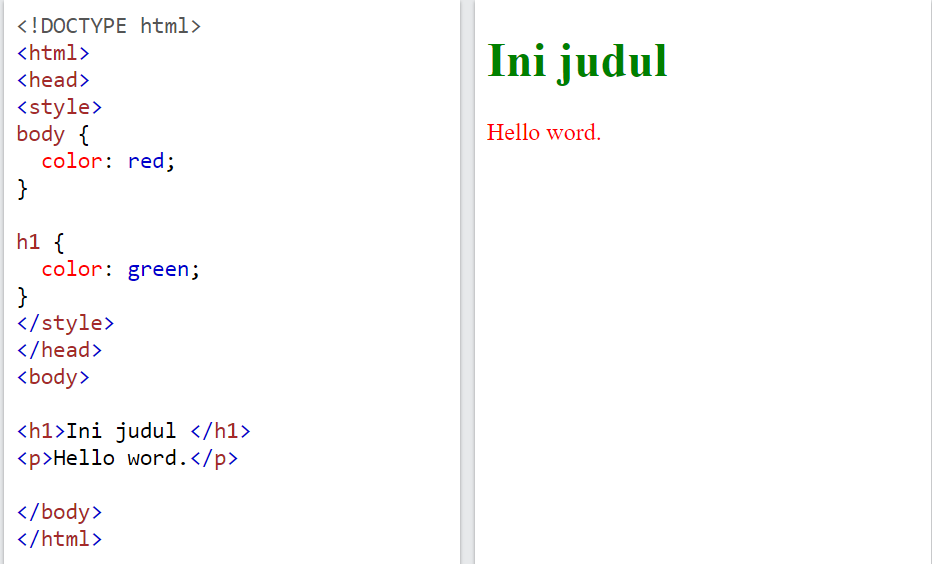
* **Color name** - seperti "merah"
* **HEX value** - seperti "#ff0000"
* **RGB value** - seperti "rgb(255,0,0)"

Dalam CSS TEXT memiliki beberapa style yaitu:

* **Text Color**
* **Text Aligment**
* **Text Decoration**
* **Text Transformation**
* **Text Spacing**
* **Text Shadow**

Warna teks default untuk halaman ditentukan di pemilih isi.

Contoh :



**CSS FONTS**

Memilih font yang tepat memiliki dampak besar pada pengalaman pembaca terhadap situs web.

Font yang tepat dapat menciptakan identitas yang kuat untuk brand.

Menggunakan font yang mudah dibaca itu penting. Font menambah nilai pada teks. Penting juga untuk memilih warna dan ukuran teks yang tepat untuk font.

Dalam CSS FONT memiliki style font yaitu:

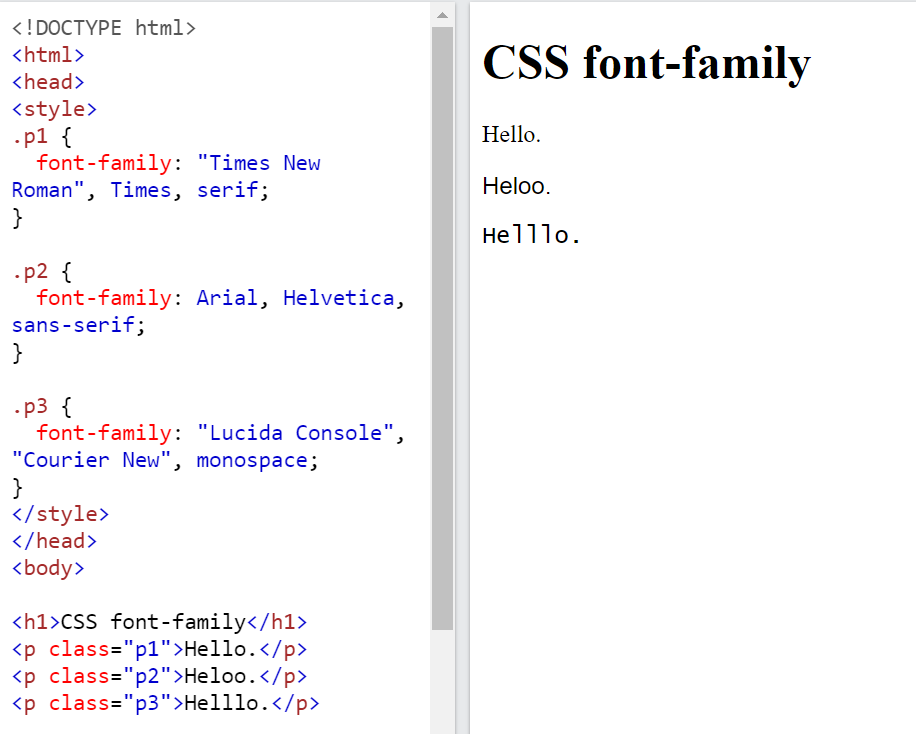
* **Font family**
* **Font websafe**
* **Font fallback**
* **Font style**
* **Font size**
* **Font google**
* **Font pairing**
* **Font shorthand**

**Generic Font Families**

Di CSS ada lima font generic umum:

* Serif font memiliki guratan kecil di tepi setiap huruf. Mereka menciptakan kesan formalitas dan keanggunan.
* Sans-serif font memiliki garis yang bersih (tidak ada guratan kecil yang menempel). Mereka menciptakan tampilan modern dan minimalis.
* Monospace font - di sini semua huruf memiliki lebar tetap yang sama. Mereka menciptakan tampilan mekanis.
* Cursive font meniru tulisan tangan manusia.
* Fantasy font adalah font dekoratif/menyenangkan.
* Semua nama font yang berbeda termasuk dalam salah satu keluarga font umum.

Contoh :



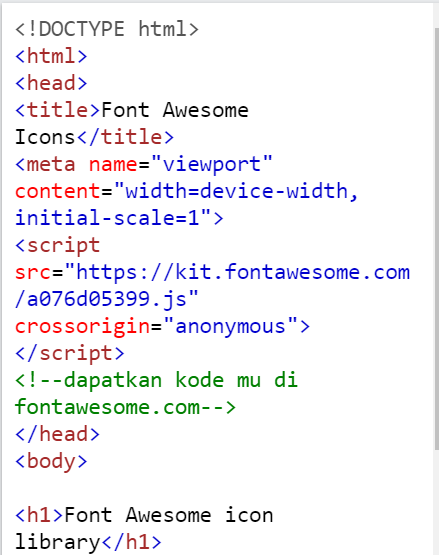
**CSS ICONS**

Bagaimana cara menambahkan Icons?

Cara paling sederhana untuk menambahkan ikon ke halaman HTML Anda adalah dengan perpustakaan ikon, seperti Font Awesome. Tambahkan nama kelas ikon yang ditentukan ke elemen HTML sebaris mana pun (seperti <i> atau <span>).

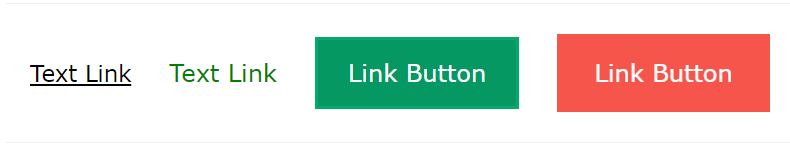
Semua ikon di perpustakaan ikon di bawah, adalah vektor terukur yang dapat disesuaikan dengan CSS (ukuran, warna, bayangan, dll.)

Contoh :



**CSS LINK**

Dengan CSS LINK akan memiliki banyak style yang berbeda.



**STYLING LINKS**

LINKS ditata dengan properti CSS apa pun (misalnya warna, font-family, latar belakang, dll.).

Contoh :

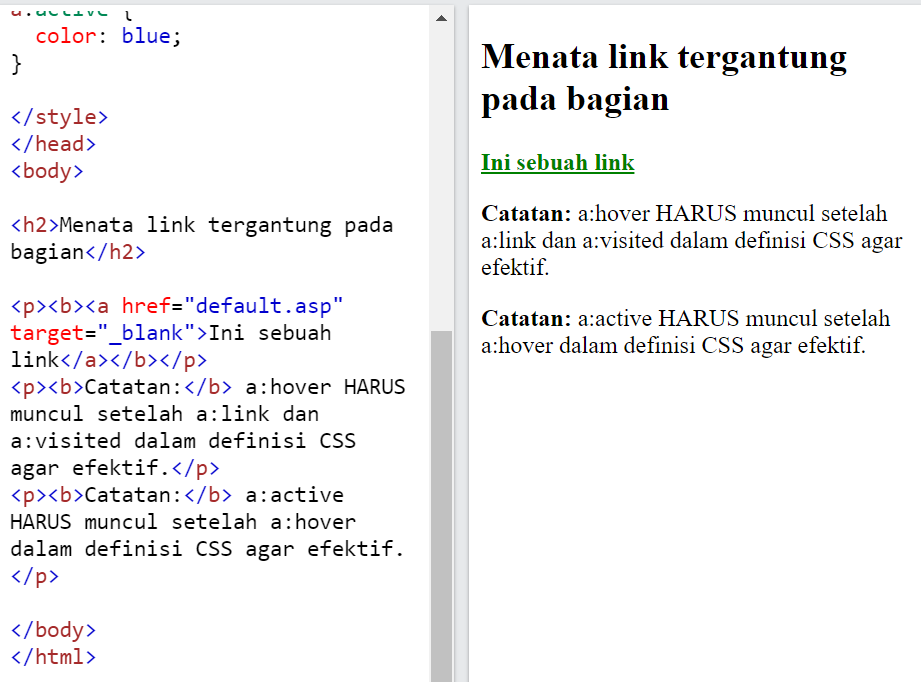


Selain itu, style links dapat berbeda-beda, bergantung pada statusnya.

Empat status links adalah:

* **a:link** – link normal yang belum dikunjungi
* **a:visited** - link yang telah dikunjungi pengguna
* **a:hover** - link saat pengguna mengarahkan mouse ke atasnya
* **a:active** - link saat diklik

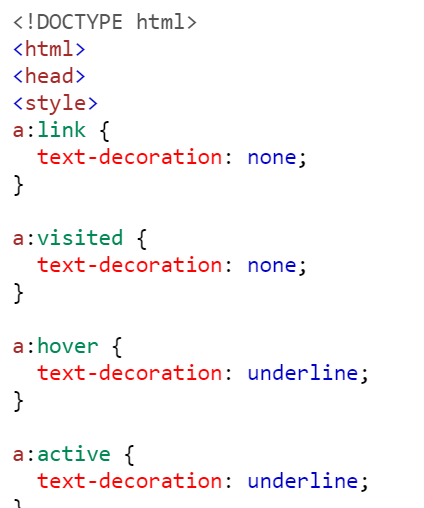
contoh:



**TEXT DECORATION**

Properti text-decoration sebagian besar digunakan untuk menghilangkan garis bawah pada tautan:

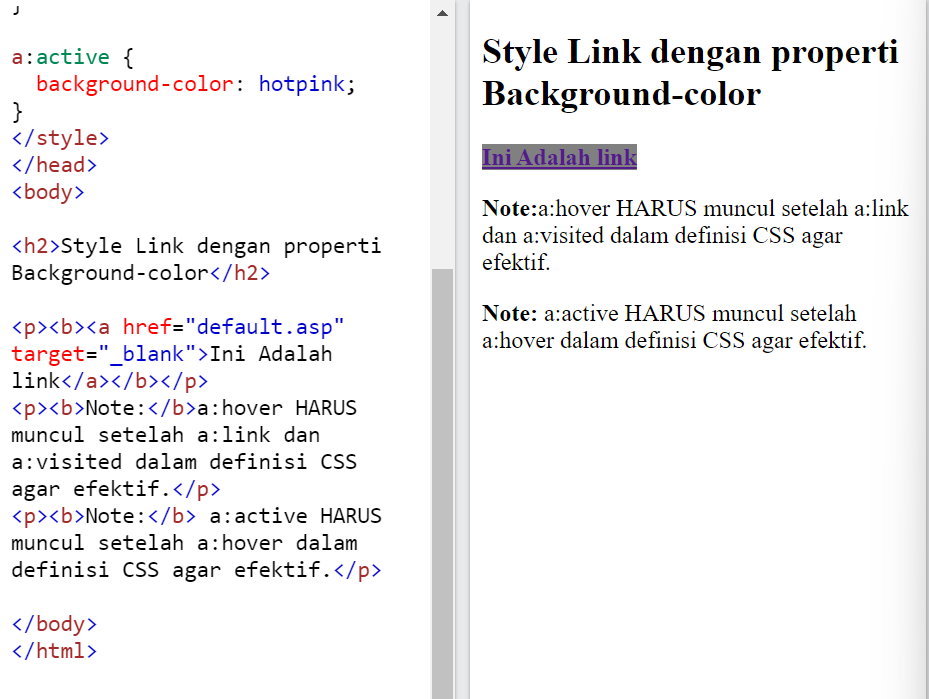
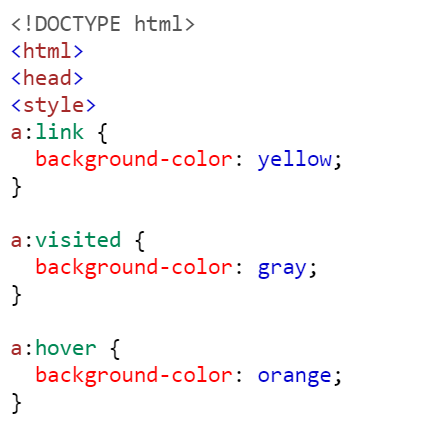
Contoh :



**BACKGROUND COLOR**

Properti background-color dapat digunakan untuk menentukan warna latar belakang link:

Contoh :



**LINK BUTTONS**

ini menunjukkan contoh lebih lanjut di mana menggabungkan beberapa properti CSS untuk menampilkan tautan sebagai kotak/tombol.

Contoh :



**CSS LIST**

HTML LIST dan CSS LIST Properti

Dalam HTML, ada dua tipe utama daftar:

* **Unorder list (<ul>)** - item daftar ditandai dengan poin
* **Ordered list (<ol>)** - item daftar ditandai dengan angka atau huruf

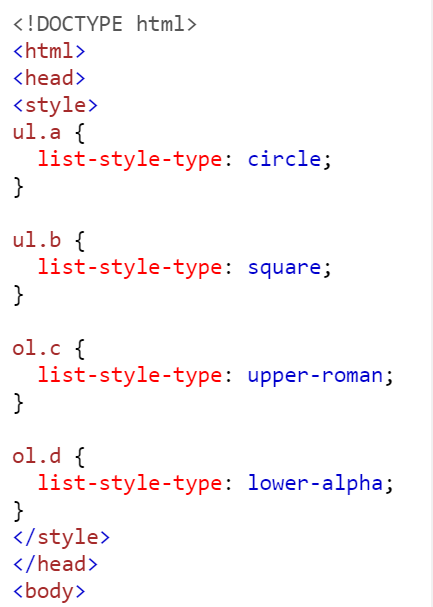
Properti daftar CSS memungkinkan Anda untuk:

* Tetapkan penanda item daftar yang berbeda untuk daftar yang dipesan
* Tetapkan penanda item daftar yang berbeda untuk daftar tidak berurutan
* Tetapkan gambar sebagai penanda item daftar
* Tambahkan warna latar belakang ke daftar dan item daftar

**DIFFERENT LIST ITEM MAKERS**

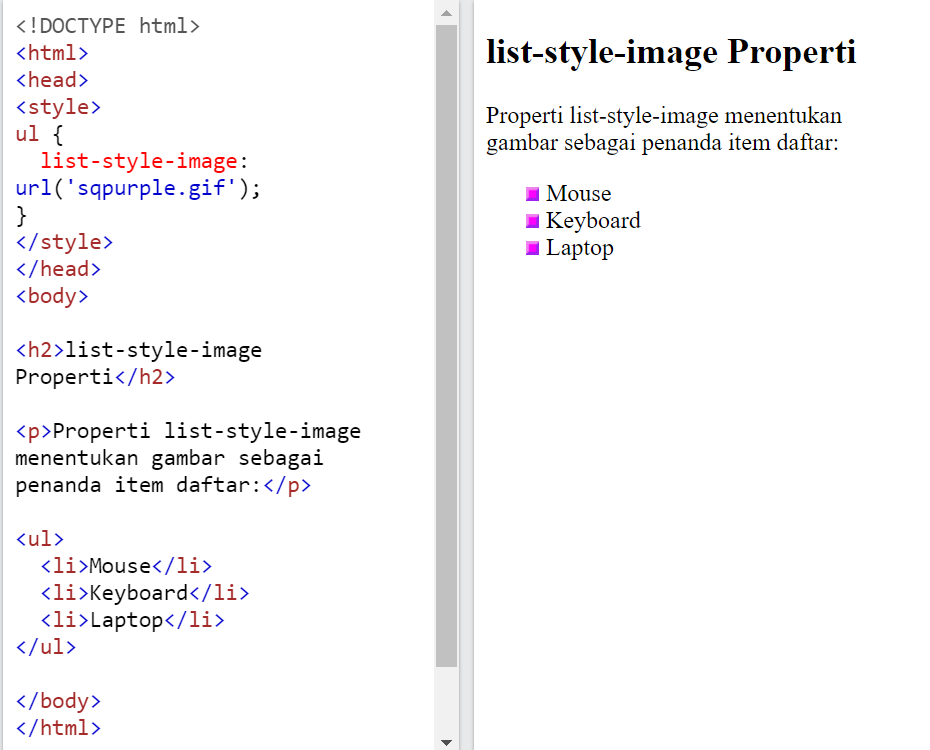
Properti list-style-type menentukan tipe penanda item daftar.

Contoh berikut memperlihatkan beberapa penanda item daftar yang tersedia:



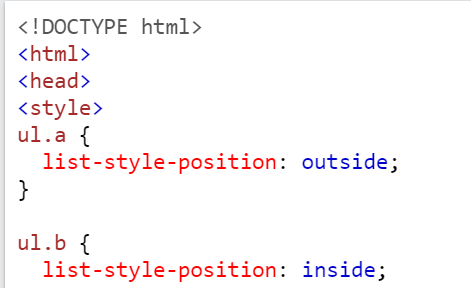
**IMAGE AS THE LIST ITEM MAKER**

Properti list-style-image menentukan gambar sebagai penanda item daftar:



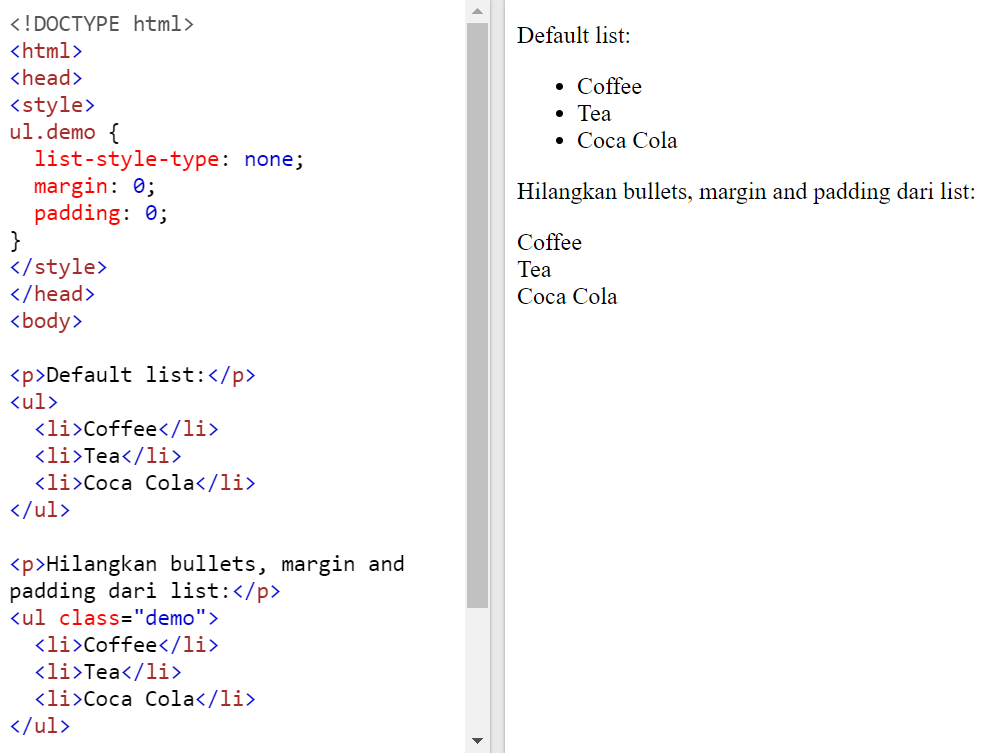
**POSITION LIST ITEM MAKERS**

Properti list-style-position menentukan posisi penanda item daftar (poin-poin):



**REMOVE DEFAULT SETTINGS**

Properti list-style-type:none juga dapat digunakan untuk menghilangkan penanda/poin. Perhatikan bahwa daftar ini juga memiliki margin dan padding default. Untuk menghapusnya, tambahkan margin:0 dan padding:0 ke <ul> atau <ol>:



**LIST- SHORTHAND PROPERTI**

Properti list-style property adalah properti singkatan. Ini digunakan untuk mengatur semua properti daftar dalam satu deklarasi.

Saat menggunakan Shorthand properti, urutan nilai properti adalah:

* **list-style-type** (jika gambar gaya daftar ditentukan, nilai properti ini akan ditampilkan jika gambar karena alasan tertentu tidak dapat ditampilkan)
* **list-style-position** (menentukan apakah penanda item daftar harus muncul di dalam atau di luar aliran konten)
* **list-style-image** (menentukan gambar sebagai penanda item daftar)

Jika salah satu nilai properti di atas tidak ada, maka nilai default untuk properti yang hilang akan dimasukkan, jika ada.

**STYLING LIST WITH COLORS**

Menata daftar dengan warna, untuk membuatnya terlihat sedikit lebih menarik.

Apa pun yang ditambahkan ke tag <ol> atau <ul>, akan memengaruhi keseluruhan daftar, sedangkan properti yang ditambahkan ke tag <li> akan memengaruhi masing-masing item daftar:

