**Dokumentasi CSS**

## Apa Itu CSS ?

CSS (Cascade Style Sheet) adalah sebuah bahasa untuk mengatur tampilan web sehingga terlihat lebih menarik dan indah. Dengan CSS, kita dapat mengatur layout (tata letak), warna, font, garis, background, animasi, dan lain-lain.

## Sejarah dan Perkembangan CSS

Sebelum adanya CSS, tidak ada bahasa yang dipakai untuk memberikan style pada dokumen (web). Tampilan web terasa hambar dan kurang tertata. Ide awal tentang style sheet di browser bukan sesuatu yang baru. Pada tahun 1990, Tim Berners-Lee membuat web browser bernama NeXT browser/editor.

Pada browser ini, kita bisa membuat style sheet. Akan tetapi, sintaks style sheet-nya tidak ia publikasikan. Lalu pada akhirnya, di tanggal:

* 10 Oktober 1994, Håkon Wium Lie membuat draft proposal bahasa CSS dan mengirimnya ke mailist www-talk dan www-html.
* November 1994, Wium Lie mempresentasikan draft propsal bahasa CSS (Cascading HTML Style Sheets) di acara Web Conference di Chicago. Orang pertama yang merespon proposal ini adalah Bert Bos. Ia saat itu sedang mengembangkan browser bernama Argo. Kemudian ia memutuskan untuk bergabung dengan Wium Lie untuk mengembangkan CSS.
* April 1995, Draft bahasa CSS dipresentasikan lagi pada acara WWW Conference. Pada acara ini Wium Lie bertemu langsung dengan Bert Bos. Bert Bos membawakan presentasi tentang dukungan CSS pada Argo dan Wium Lie menampilkan demo Arena Browser yang sudah ia modifikasi agar mendukung penggunaan CSS.
* Mei 1995, Implementasi bahasa CSS mulai dikerjakan. Pada bulan ini mail list (group) baru bernama www-style dibuat. Mai list menjadi tempat diskusi pengembangan CSS.
* 17 Desember 1996, CSS versi pertama atau CSS 1 akhirnya dirilis.
* Mei 1998, CSS veri keuda atau CSS 2 dirilis. Pada versi ini ditambahkan dukungan media output.

## Sejarah dan Perkembangan CSS

Sejak awal rilis, CSS memiliki beberapa versi:

* CSS 1 (1996): adalah versi pertama (17 Desember 1996)
* CSS 2 (1998): adalah versi ke-2 (Mei 1998)
* CSS 2.1 (2011): dirilis pada 7 juni 2011
* CSS 3 (2012): versi yang banyak digunakan saat ini.

Sintaks dasar di tiap versi sama. Hanya saja ada beberapa perbedaan properti. Ada yang ditambahkan, ada yang dikurangi atau dihapus.

## Selektor CSS

Selektor adalah katakunci dan simbol yang digunakan pada CSS untuk menyeleksi atau memilih elemen HTML.

Ada 6 macam selektor di CSS:

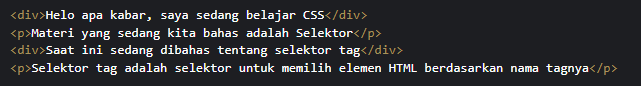
* Selektor Tag
* Selektor Class
* Selektor ID
* Selektor Atribut
* Selektor Universal
* Selektor Pseudo

### Selektor Tag

Selektor Tag disbut juga Type Selector. Selektor ini akan memilih elemen berdasarkan nama tag.

Contoh :





Artinya: Pilih semua elemen <p> lalu atur warna teksnya menjadi biru. Semua elemen <p> berwarna biru, sedangkan elemen <div> tidak.

### Selektor Class

Selektor class adalah selektor yang memilih elemen berdasarkan nama class yang diberikan. Selektor class dibuat dengan tanda titik di depannya. Contoh :



Saya memiliki selector class bernama .pink. Nah cara menggunakan selektor ini di HTML adalah dengan menambahkan atribut class di dalamnya. Contoh :



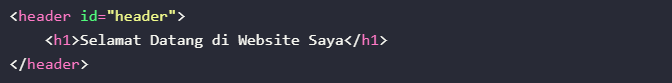
Selector class dapat kita gunakan pada elemen yang kita inginkan. Sebuah elemen HTML dapat menggunakan satu atau lebih class.

### Selektor ID

Selektor ID hampir sama dengan class. Bedanya, ID bersifat unik. Hanya boleh digunakan oleh satu elemen saja. Selektor ID ditandai dengan tanda pagar (#) di depannya. Contoh :



Lalu pada kode HTML :



### Selektor Atribut

Selektor atribut adalah selektor yang memiliki elemen berdasarkan atribut. Selektor ini hampir sama seperti selektor Tag. Contoh :



Artinya kita akan memilih semua elemen <input> yang memiliki atribut type='text'.

Contoh kode HTML :



### Selektor Universal

Selektor universal adalah selektor yang digunakan untuk menyeleksi semua elemen pada jangkaua (scope) tertentu. Contoh



Artinya semua elemen akan memiliki garis solid dengan ukuran 1px dan berwarna grey.

Selektor universal bisanya digunakan untuk me-reset CSS. Kenapa harus di-reset? Pada halaman HTML, ada beberapa CSS bawaan browser seperti padding dan margin pada elemen tertentu. Reset bertujuan untuk menghilangkan padding dan margin tersebut.

Contoh CSS reset :



Maka semua elemen tidak akan memiliki padding dan margin.

### Pseudo Selektor

Pseudo selektor adalah selektor untuk memilih elemen semu seperti state pada elemen, elemen before dan after, elemen ganjil, dan sebagainya.

Ada dua macam pseudo selektor:

* pseudo-class selektor untuk state elemen;
* pseudo-element selektor untuk elemen semu di HTML.
* Pseudo Class

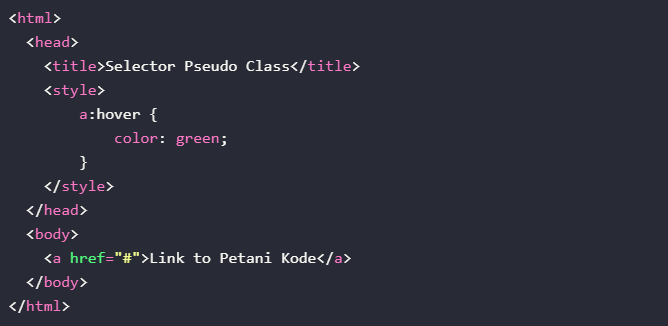
Pseudo-class adalah selektor untuk memilih state pada elemen. Contohnya seperti elemen saat diklik, saat fokus, saat disentuh, dan lain sebagainya. Dengan selektor ini kita bisa menentukan tampilan dari elemen pada state tersebut.

Format penulisan pseduo-class menggunakan titik dua setelah elemen.



Contoh :





Artinya: Kita akan memberikan warna hijau pada elemen <a> saat dia di-hover atau disentuh pointer.

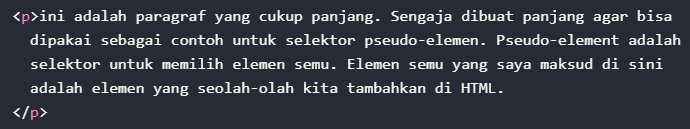
Selain :hover ada beberapa selektor pseudo-class lainnya:

* **:link** untuk memlih link yang belum dikunjungi;
* **:visited** untuk memlih link yang sudah dikunjungi;
* **:active** untuk memilih elemen yang sedang aktif, seperti saat diklik;
* :**focus** untuk memlih elemen yang sedang dalam keadaan fokus, misal teks lagi diinput.
* **:checked** untuk memilih elemen yang dicentang pada checkbox.

Masih banyak lagi, list lengkapnya bisa cek di MDN Doc.

* Pseudo Element

Pseudo-element adalah selektor untuk memilih elemen semu. Elemen semu yang dimaksud di sini adalah elemen yang seolah-olah di tambahkan di HTML. Misalnya saya punya elemen paragraf seperti ini:



Lalu saya ingin memberikan style hanya pada baris pertama saja. Tanpa pseudo-element, kita bisa tambahkan elemen <span> di baris pertama menjadi seperti ini.



Kemudian memilih elemen span tersebut di CSS seperti ini:



Cara ini sebenarnya tidak disarankan, karena kita harus membuat elemen baru di dalam paragraf. Belum lagi, jika paragrafnya cukup panjang baris pertama akan tambah panjang pula. Maka di sinilah kita bisa pakai selektor pseudo-element ::first-line.

Sehingga tanpa menambahkan elemen <span>, kode CSS-nya akan menjadi seperti ini.



Apa bedanya dengan pseudo-class? Selektor pseudo-element menggunakan tanda titik dua ganda (::) sedangkan pseudo-class pakai satu titik dua (:).

Berikut ini beberapa contoh selektor pseudo-element:

* **::before** untuk memilh elemen semu sebelum elemen;
* **::after** untuk memilh elemen semu setelah elemen;
* **::marker** untuk memilh marker pada list;
* **::placeholder** untuk memilih teks placeholder pada elemen input teks;

List lengkap pseudo-element bisa di cek di MDN Doc.

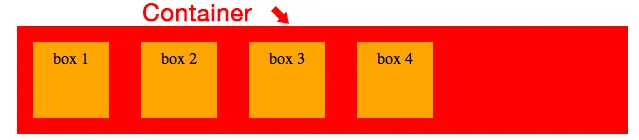
## Flex

Flex (flexibel) adalah salah satu super power display dari CSS3. Memungkinkan untuk pengaturan layout, posisi dan tampilan dari suatu konten yang ukurannya belum diketahui atau bernilai dinamis. Oleh sebab itu, nilai ini dinamakan Flex diambil dari kata flexibel yang berarti mudah diatur, disesuaikan dan ditukar-tukar.

Flex sangat cocok digunakan untuk komponen sebuah aplikasi dan layout skala kecil. Tujuan utama penggunaan layout flex adalah untuk memberi kemampuan container agar dapat memanipulasi item-item (elemen) yang ada didalamnya, baik itu merubah tinggi dan lebar, urutan posisi elemen, dan spasi (jarak) diantara elemen tersebut.

Secara garis besar flex dibagi menjadi dua :

* **Flex Container**



Pada bagian ini flex container bertindak sebagai parent. Terdapat beberapa nilai yang dapat diatur pada flex container, diantaranya :

1. Display

Pengaturan display flex dilakukan pada container. Terdapat dua value display, yaitu flex (block) dan inline-flex (inline).

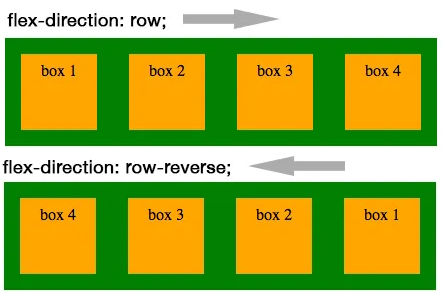


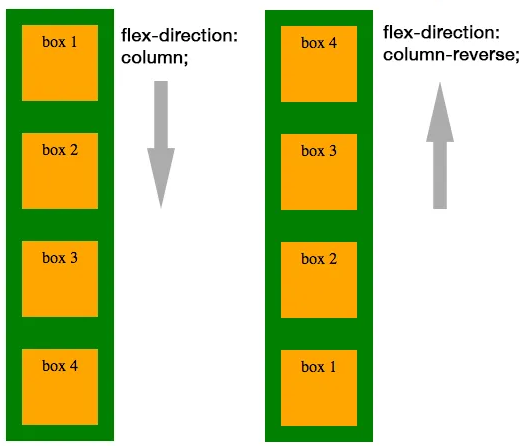


1. Flex Direction

Digunakan untuk mengatur arah alur dari items yang ada pada container flex. Default valuenya row. Terdapat empat nilai :

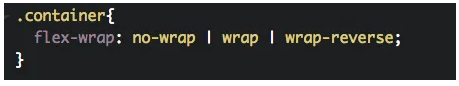


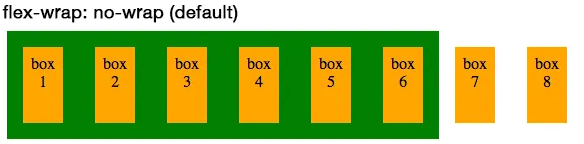


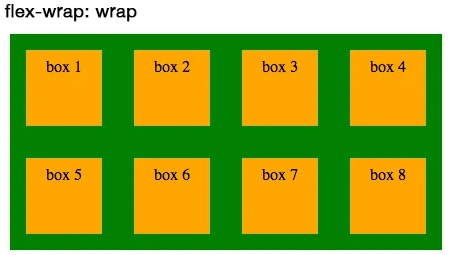


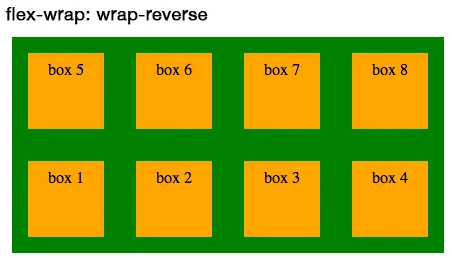
1. Flex Wrap

Menentukan apakah item fleksibel harus dibungkus atau tidak.



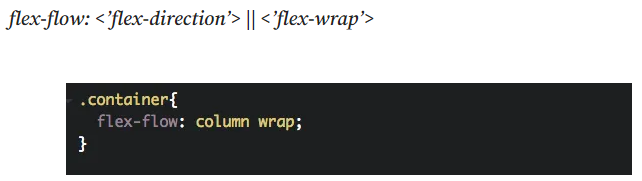






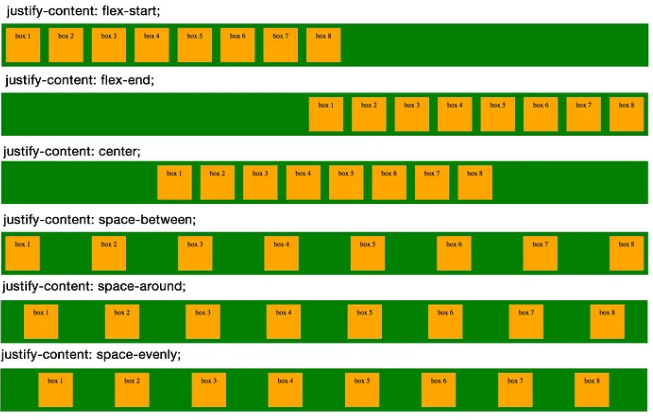
1. Flex Flow

Flex-flow merupakan shorthand dari flex-direction dan flex-wrap properti.



1. Justify Content

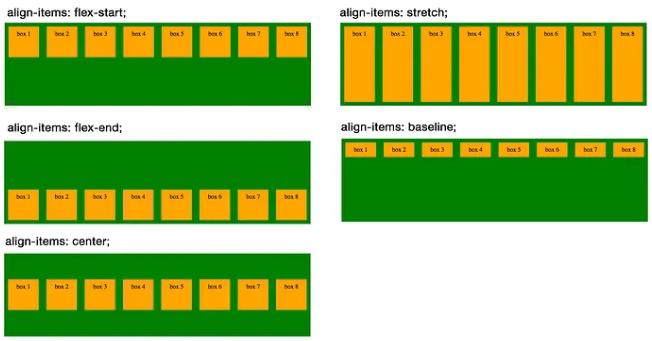
Justify-content digunakan untuk mensejajarkan setiap items yang ada pada flex container, agar container tersebut bisa mendistribusikan ruang kosong yang tersisa ketika item flex dalam satu baris tersebut tidak flexsibel atau meskipun flexsibel tapi sudah mencapai batas ukuran maksimum. Pengaturan justify-content berdasarkan pada sumbu-x atau pengaturan items secara horizontal.



* flex-start (default): item dimulai dari awal (kiri / kanan sesuai flex-direction)
* flex-end: item dimulai dari akhir (kiri / kanan sesuai flex-direction)
* center: item berpusat ditengah-tengah di sepanjang garis container flex
* space-between: item merata dalam container; item pertama ada di garis awal, item terakhir di garis akhir
* space-around: item didistribusikan secara merata di dalam container flex dengan ruang yang sama di sekitar mereka.
* space-evenly: item didistribusikan sehingga jarak antara dua item (dan ruang ke tepi) sama.

1. Align Items

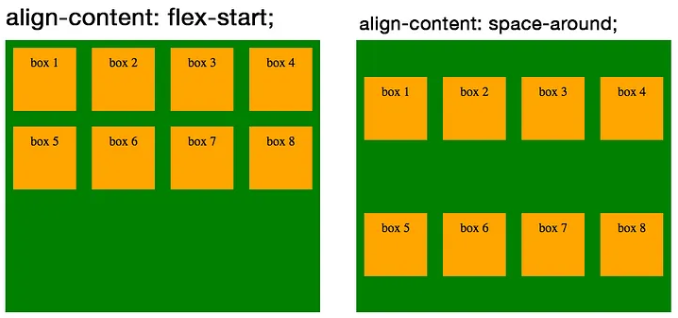
Align-items hampir sama dengan konsep justify-content hanya saja pada align-items, item-item flex diatur berdasarkan sumbu-y atau sepanjang garis tegak-lurus atau secara vertikal.

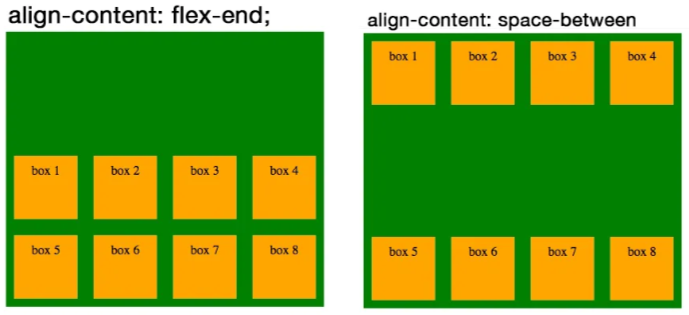


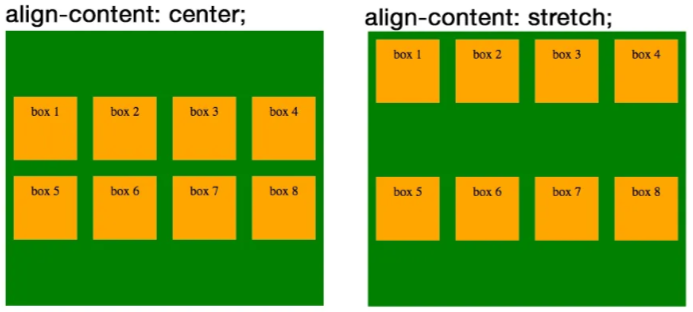
* flex-start: margin tepi cross-start item ditempatkan pada garis cross-start (rata atas) container.
* flex-end: margin tepi cross-end dari item ditempatkan pada garis cross-end (rata bawah) container.
* center: item berpusat pada sumbu silang (ditengah) container.
* baseline: item sejajar seperti garis dasar text mereka sejajar.
* stretch (default): peregangan untuk mengisi kontainer (tetap mengutamakan nilai min-width / max-width)

1. Align Items

Algin-content memadukan konsep align-items dan justify-content dimana digunakan untuk mensejajarkan garis flex container ketika ada ruang kosong secara garis tegak lurus pada sumbu-y. Antar items menggunakan konsep align items dan antar line menerapkan konsep justify-content.







* flex-start: baris dimulai dari awal container
* flex-end: baris dimulai dari akhir container
* center: baris yang diletakkan pada tengah container
* space-between: baris terdistribusi merata; Baris pertama ada di awal container sementara yang terakhir ada di bagian akhir
* space-around: baris merata didistribusikan dengan ruang yang sama di sekitar setiap baris
* stretch (default): baris peregangan untuk mengambil ruang yang tersisa

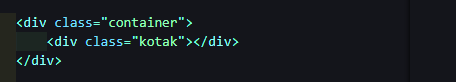
Property align-content tidak dapat diterapkan jika item-item flex terdiri hanya satu line.

## Transition

Transition dalam CSS memungkinkan kita mengatur animasi perubahan properti dari satu nilai ke nilai lainnya. Ini memberikan efek perubahan yang lebih halus dan lebih menarik.

* transition-property: Menentukan properti mana yang akan mengalami transisi.
* transition-duration: Menentukan durasi transisi dalam satuan detik atau milidetik.
* transition-timing-function: Menentukan jenis perubahan kecepatan selama transisi.
* transition-delay: Menunda awal transisi.

Contoh transition sederhana :



Hasilnya :



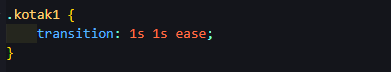
 

Contoh transiton menggunakan timing function :

* ease

ease – mengatur kecepatan transisi dengan permulaan yang pelan dan di akhirnya juga pelan.

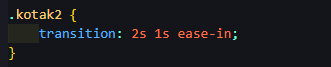




* ease-in

ease-in – kecepatan efek transisi dengan permulaan yang pelan.

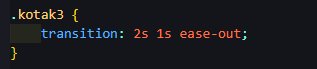




* ease-out

ease-out – kecepatan efek transisi dengan akhir yang pelan.

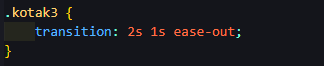




* ease-in-out

ease-in-out – kecepatan efek transisi dengan akhir dan awal pelan.

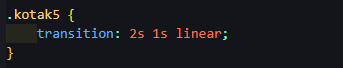




* linear

linear – kecepatan efek tansition/transisi dengan kecepatan yang sama dari awal hingga akhir.

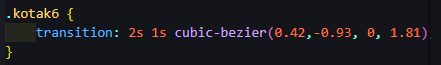




* cubic-bezier

cubic-bezier(n,n,n,n) – kita bisa memberikan nilai pada n untuk menentukan nilai speed transition (bisa menggunakan inspect element)





**Properti-properti CSS**

|  |  |
| --- | --- |
| **Properti** | **Fungsi** |
| background | Menentukan berbagai properti latar belakang dalam satu deklarasi |
| background-color | Menentukan warna latar belakang elemen pada halaman web |
| background-image | Menentukan gambar latar belakang suatu elemen |
| background-clip | Menentukan seberapa jauh gambar latar atau warna memanjang untuk suatu elemen |
| background-origin | Menentukan area posisi gambar latar belakang |
| background-position | Mengatur posisi awal dari gambar latar belakang |
| background-repeat | Mengatur bagaimana gambar latar belakang akan diulangi |
| background-size | Menentukan ukuran gambar latar belakang |
| border | Mengatur lebar, gaya, dan warna batas untuk keempat sisi elemen |
| border-bottom | Mengatur lebar, gaya, dan warna untuk batas bawah suatu elemen |
| border-bottom-color | Mengatur warna batas bawah suatu elemen |
| border-bottom-left-radius | Menentukan bentuk sudut batas kiri bawah elemen |
| border-bottom-right-radius | Menentukan bentuk sudut batas kanan bawah elemen |
| border-bottom-style | Mengatur gaya batas bawah elemen |
| border-bottom-width | Mengatur lebar batas bawah suatu elemen |
| border-color | Mengatur warna batas pada keempat sisi elemen |
| border-left | Mengatur lebar, gaya, dan warna batas kiri suatu elemen |
| border-left-color | Mengatur warna batas kiri elemen |
| border-left-style | Mengatur gaya batas kiri elemen |
| border-left-width | Mengatur lebar batas kiri suatu elemen |
| border-radius | Menentukan bentuk sudut batas elemen |
| border-right | Mengatur lebar, gaya, dan warna batas kanan suatu elemen |
| border-right-color | Mengatur warna batas kanan suatu elemen |
| border-right-style | Mengatur gaya batas kanan elemen |
| border-right-width | Mengatur lebar batas kanan elemen |
| border-style | Mengatur gaya perbatasan pada keempat sisi elemen |
| border-top | Mengatur lebar, gaya, dan warna batas atas suatu elemen |
| border-top-color | Mengatur warna batas atas suatu elemen |
| border-top-left-radius | Menentukan bentuk sudut batas kiri atas suatu elemen |
| border-top-right-radius | Menentukan bentuk sudut batas kanan atas suatu elemen |
| border-top-style | Mengatur gaya batas atas elemen |
| border-top-width | Mengatur lebar batas atas suatu elemen |
| border-width | Mengatur lebar batas pada keempat sisi elemen |
| outline | Mengatur semua properti outline dalam satu deklarasi |
| outline-color | Mengatur warna garis besar |
| color | Menentukan dan menetapkan warna untuk teks |
| opacity | Menentukan transparansi elemen. |
| max-height | Menentukan ketinggian maksimum suatu elemen |
| max-width | Menentukan lebar maksimum suatu elemen |
| min-height | Menentukan ketinggian minimum suatu elemen |
| min-width | Menentukan lebar minimum suatu elemen |
| height | Menentukan tinggi suatu elemen |
| width | Menentukan lebar elemen |