

CSS DASAR



Soetrisno Soekarno

IT Konsultan, Web Developer, Agro Bisnis,
Digital Marketing, SEO Specialist, Internet
Provider

www.rumah-belajar.com

www.mitrainformatikaindonesia.co.id

www.soetrisnogaletti.com

www.jualbibit.my.id

SEBELUM BELAJAR

Sudah menguasai HTML Dasar, HTML Form

Bagi yang belum bisa bisa di tonton di video di bawah ini:

- Belajar HTML Part 1: <https://www.youtube.com/watch?v=zOJkjs6p4NE&t=30s>
- Belajar HTML Part 2: <https://www.youtube.com/watch?v=CkxWo1Tx-TU&t=294s>
- Belajar HTML Part 3: <https://www.youtube.com/watch?v=2--H8x0VeK8&t=48s>
- Belajar HTML Part 4: <https://www.youtube.com/watch?v=JdeJwnznIXI&t=593s>
- Belajar HTML Part 5: <https://www.youtube.com/watch?v=q4Yitm0fr2E&t=0s>
- Belajar HTML Part 6: <https://youtu.be/BHNqTnitzX4?si=yVTR87HpkE2BxLPp>



MEMBUAT PROJECT



MEMBUAT PROJECT

- Buatlah folder :
belajar-css-1

LIVE PREVIEW

- Untuk mempermudah belajar CSS, silahkan install extension Live Preview di Visual Studio Code
- <https://marketplace.visualstudio.com/items?itemName=ms-vscode.live-server>



PENGENALAN CSS



PENGENALAN CSS

- CSS singkatan dari Cascading Style Sheets
- HTML digunakan untuk membuat struktur konten web secara semantic, dan CSS digunakan untuk memberi style (gaya) dan layout (tata letak) pada konten HTML
- Contohnya, kita bisa menggunakan CSS untuk mengubah font, color, size, dan lain-lain pada konten yang sudah kita buat menggunakan HTML



APA ITU CSS?

- CSS adalah bahasa yang digunakan untuk mendeskripsikan bagaimana sebuah dokumen yang sudah dibuat menggunakan HTML, ditampilkan ke pengguna
- Setiap browser biasanya punya standar masing-masing ketika menampilkan dokumen HTML
- Dengan menggunakan CSS, kita bisa membuat semua browser menampilkan dokumen HTML dengan cara yang sama

CSS SYNTAX

- CSS adalah bahasa yang berbasis rule/aturan
- Kita akan mendefinisikan rule/aturan aturan untuk element yang terdapat di dokumen HTML yang kita buat
- Membuat rule di CSS biasanya dimulai dengan menyebutkan element yang akan dipilih, lalu diikuti dengan kurung kurawal buka, dilanjutkan dengan aturan-aturan yang akan kita gunakan, dan diakhiri dengan kurung kurawal tutup
- Aturan dalam CSS disebutkan dengan
property: value ;

KODE : CONTOH CSS SYNTAX

```
h1 {  
  color: red;  
  font-size: 20px;  
}
```

```
h2 {  
  color: blue;  
  font-size: 15px;  
}
```



MENAMBAH CSS

MENAMBAH CSS KE HTML

- Terdapat tiga cara untuk menambah CSS ke HTML
- Cara internal, dimana CSS disimpan di file yang sama dengan HTML. Yaitu dengan menggunakan tag style
- Cara external, dimana CSS disimpan di file yang berbeda dengan HTML. Yaitu menggunakan tag link,
- Cara inline, dimana CSS disimpan dalam atribut styles pada tag html, ini pernah kita lakukan di kelas HTML, dan cara ini tidak direkomendasikan

KODE : CSS INTERNAL

```
8 <title>hello CSS</title>
9 <style>
10
11     h1 {
12         color: red;
13         font-size: 20px;
14     }
15
16     h2 {
17         color: blue;
18         font-size: 15px;
19     }
20
21 </style>
22 </head>
23 <body>
24     <h1>Hello CSS</h1>
25     <h2>Selamat belajar CSS</h2>
26 </body>
27 </html>
```

KODE : CSS EXTERNAL (1)

hello.css ×

hello.css >  h2

```
1  h1 {  
2    color:  red;  
3    font-size: 20px;  
4  }  
5  
6  h2 {  
7    color:  blue;  
8    font-size: 15px;  
9  }
```

KODE : CSS EXTERNAL (2)

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Hello CSS</title>
<link rel="stylesheet" href="hello.css">
</head>
<body>
  <h1>Hello CSS</h1>
  <h2>Selamat belajar CSS</h2>
</body>
</html>
```



COMMENT

COMMENT

- Saat membuat kode CSS, kadang kita ingin menyisipkan komentar
- Komentar adalah kode yang tidak akan berdampak apapun
- Untuk membuat komentar kita harus mengawali dengan `/*` dan diakhiri dengan `*/`
- Komentar bisa multi baris

KODE : COMMENT

```
/* Ini adalah komentar satu baris */
```

```
h1 {  
  color:  red;  
  font-size: 20px;  
}
```

```
/*  
Ini adalah komentar  
multi baris
```

```
*/  
h2 {  
  color:  blue;  
  font-size: 15px;  
}
```



ID



ID

- Selain menggunakan memilih element menggunakan HTML Tag, kita juga bisa memilih element menggunakan ID di CSS
- Untuk menggunakan ID, kita perlu menggunakan # diawal nama ID nya

KODE : ID

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>ID</title>
<style>
  #judul {
    size: 10px;
    color: red;
  }
</style>
</head>
<body>
  <h1 id="judul">Ini adalah judul</h1>
  <h1>Ini judul lainnya</h1>
</body>
</html>
```



CLASS

CLASS

- Kadang ada kasus kita ingin menambahkan style CSS ke beberapa element tag yang jenisnya berbeda
- Jika kita harus buat aturan nya per tag, maka akan menyulitkan ketika banyak sekali
- Di HTML, semua tag bisa memiliki atribut class, dan di CSS, kita bisa menambahkan aturan ke class dengan menggunakan awalan . (titik)

KODE : CLASS

```
<style>
  .red {
    color: red;
  }
</style>
</head>
<body>
  <h1 class="red">This Is Title</h1>
  <h2 class="red">This is Sub Title</h2>
  <p class="red">This is artitle</p>
</body>
</html>
```




MULTIPLE CLASS

- Nilai dari atribut class bisa menggunakan beberapa nilai, caranya kita bisa tambahkan pemisah menggunakan spasi

KODE : MULTIPLE CLASS

```
<style>
  .red {
    color: red;
  }
  .upper {
    text-transform: uppercase;
  }
</style>
</head>
<body>
  <h1 class="red upper">This Is Title</h1>
  <h2 class="red upper">This is Sub Title</h2>
  <p class="red">This is artitle</p>
</body>
```



SELECTOR



SELECTOR

- Sebelumnya kita sudah tahu cara memilih elemen yang akan ditambahkan aturan di CSS, yaitu menggunakan tag, #id atau .class
- Memilih elemen di CSS dinamakan Selector
- Ada banyak jenis Selector, dan kita akan bahas tiap selector di materi masing-masing
- Di materi ini kita akan bahas tentang simple selector




SIMPLE SELECTOR

- Simple selector adalah selector untuk memilih elemen berdasarkan nama (tag), #id atau .class
- Jika kita ingin membuat selector untuk beberapa element, kita bisa gunakan , (koma) sebagai pemisah

TYPE SELECTOR

- Type Selector melakukan seleksi element berdasarkan tag html
- <https://css-tricks.com/almanac/selectors/t/type/>

```
h1 {  
  color:  red;  
}
```

ID SELECTOR

- ID Selector melakukan seleksi element berdasarkan ID
- <https://css-tricks.com/almanac/selectors/i/id/>

```
#title {  
  color:  red;  
}
```


CLASS SELECTOR

- Class Selector melakukan seleksi element berdasarkan nama class
- <https://css-tricks.com/almanac/selectors/c/class/>

```
.sample {  
  color:  red;  
}
```


SELECTOR LIST

- Selector list melakukan seleksi beberapa element sekaligus, menggunakan pemisah , (koma)
- Gambar disamping artinya kita melakukan seleksi element untuk tag h1, h2, dan ID title

```
h1, h2, #title {  
  color:  red;  
}
```



COMBINATOR SELECTOR

COMBINATOR SELECTOR

- Combinator adalah sesuatu yang menjelaskan relasi antar selector
- Terdapat 4 selector untuk combinator
- descendant selector (space)
- child selector ($>$)
- adjacent sibling selector ($+$)
- general sibling selector (\sim)

DESCENDANT SELECTOR

- Descendant Selector adalah selector untuk memilih element anak dari element yang dipilih
- Gambar sebelah artinya kita memilih semua tag p yang terdapat di dalam elemen div
- <https://css-tricks.com/almanac/selectors/d/descendant/>

```
div p {  
  color: red;  
}
```

```
<div>  
  <article>  
    <p>Ini Dipilih</p>  
  </article>  
  <p>Ini Dipilih</p>  
</div>  
<p>Ini Tidak Dipilih</p>
```

CHILD SELECTOR


- Child Selector adalah selector untuk memilih child / anak dari element yang dipilih
- Gambar disamping artinya kita memilih semua elemen p yang anak dari div (div adalah parent element untuk p)
- <https://css-tricks.com/almanac/selectors/c/child/>

```
div > p {  
  color: red;  
}
```

```
<div>  
  <article>  
    <p>Ini Tidak Dipilih</p>  
  </article>  
  <p>Ini Dipilih</p>  
</div>  
<p>Ini Tidak Dipilih</p>
```

ADJACENT SIBLING SELECTOR

- Adjacent Sibling Selector digunakan untuk memilih element setelah element yang dipilih
- Sibling (saudara) element harus memiliki parent element yang sama
- Gambar disamping artinya memilih element p yang diletakkan setelah element div dimana p dan div memiliki parent yang sama
- <https://css-tricks.com/almanac/selectors/a/adjacent-sibling/>

```
div + p {  
  color:  red;  
}
```

```
<body>  
  <div>  
    <article>  
      <p>Ini Tidak Dipilih</p>  
    </article>  
    <p>Ini Tidak Dipilih</p>  
  </div>  
  <p>Ini Dipilih</p>  
  <p>Ini Tidak Dipilih</p>  
</body>
```

GENERAL SIBLING SELECTOR

- General Sibling Selector digunakan untuk memilih seluruh element saudara dari element yang dipilih, dimana harus memiliki parent element yang sama
- Contoh gambar disamping adalah memilih semua element p dari saudara element div, dimana div dan parent harus memiliki parent yang sama
- <https://css-tricks.com/almanac/selectors/g/general-sibling/>

```
div ~ p {  
  color: red;  
}
```

```
<body>  
  <div>  
    <article>  
      <p>Ini Tidak Dipilih</p>  
    </article>  
    <p>Ini Tidak Dipilih</p>  
  </div>  
  <p>Ini Dipilih</p>  
  <p>Ini Dipilih</p>  
</body>
```



ATTRIBUTE SELECTOR




ATTRIBUTE SELECTORS

- CSS juga mendukung selector menggunakan atribut yang terdapat di tag HTML
- Terdapat banyak cara untuk menggunakan Attribute Selector


CSS [ATTRIBUTE] SELECTOR

- Gambar disamping artinya memilih semua element a yang memiliki atribut target
- <https://css-tricks.com/almanac/selectors/a/attribute/>

```
a[target] {  
  color:  red;  
}
```


CSS [ATTRIBUTE="VALUE"] SELECTOR

- Gambar disamping artinya memilih semua element a yang memiliki atribut target dengan nilai “_blank”
- <https://css-tricks.com/almanac/selectors/a/attribute/>

```
a[target="_blank"] {  
  color:  red;  
}
```

CSS [ATTRIBUTE~="VALUE"] SELECTOR

- Gambar disamping artinya memilih semua element p yang memiliki atribut title yang terdapat kata belajar
- <https://css-tricks.com/almanac/selectors/bute/>

```
p[title~="belajar"] {  
  color:  red;  
}
```

```
<p title="belajar pemrograman">Ini Dipilih</p>  
<p title="belajarpemrograman">Ini Tidak Dipilih</p>
```

CSS [ATTRIBUTE | ="VALUE"] SELECTOR

- Gambar disamping artinya memilih semua element p yang memiliki atribut title yang memiliki value “belajar” atau “belajar” yang diikuti dengan karakter -
- <https://css-tricks.com/almanac/selectors/b/attribute-selector/>

```
p[title|="belajar"] {  
  color: red;  
}
```

```
<p title="belajar pemrograman">Ini Tidak Dipilih</p>  
<p title="belajar">Ini Dipilih</p>  
<p title="belajar-pemrograman">Ini Dipilih</p>  
<div>
```

CSS [ATTRIBUTE^="VALUE"] SELECTOR

- Gambar disamping artinya memilih semua element p yang memiliki atribut title yang memiliki value dimulai dengan “belajar”
- <https://css-tricks.com/almanac/selectors/a/attribute/>

```
p[title^="belajar"] {  
  color: red;  
}
```

```
<p title="belajar pemrograman">Ini Dipilih</p>  
<p title="belajar">Ini Dipilih</p>  
<p title="belajar-pemrograman">Ini Dipilih</p>  
<p title="pemrograman belajar">Ini Tidak Dipilih</p>  
</div>
```

CSS [ATTRIBUTE\$="VALUE"] SELECTOR

- Gambar disamping artinya memilih semua element p yang memiliki atribut title yang memiliki value didiakhiri dengan “belajar”
- <https://css-tricks.com/almanac/selectors/a/attribute/>

```
p[title$="belajar"] {  
  color: red;  
}
```

```
<p title="belajar pemrograman">Ini Tidak Dipilih</p>  
<p title="belajar">Ini Dipilih</p>  
<p title="belajar-pemrograman">Ini Tidak Dipilih</p>  
<p title="pemrograman belajar">Ini Dipilih</p>  
</div>
```

CSS [ATTRIBUTE*="VALUE"] SELECTOR

- Gambar disamping artinya memilih semua element p yang memiliki atribut title yang mengandung kata “belajar”
- <https://css-tricks.com/almanac/selectors/a/attribute/>

```
p[title*="belajar"] {  
  color: red;  
}
```

```
<p title="belajar pemrograman">Ini Dipilih</p>  
<p title="belajar">Ini Dipilih</p>  
<p title="belajar-pemrograman">Ini Dipilih</p>  
<p title="pemrograman belajar">Ini Dipilih</p>  
</div>
```


ATTRIBUTE SELECTOR TANPA TAG

- Saat menggunakan Attribute Selector, nama tag sebenarnya tidak wajib, jadi kita bisa langsung menggunakan Attribute Selector
- Attribute Selector juga bisa digunakan pada Class atau ID
- <https://css-tricks.com/almanac/selectors/a/attribute/>

```
[title*="belajar"] {  
  color: red;  
}
```

```
.article[title*="belajar"] {  
  color: red;  
}
```



COLOR


COLOR

- Color merupakan rule di CSS untuk mengubah warna, biasa digunakan pada teks
- CSS mendukung banyak cara menggunakan color
- <https://developer.mozilla.org/en-US/docs/Web/CSS/color>

HTML COLOR NAMES

- Sebelumnya kita pernah menggunakan color red, selain red, HTML mendukung banyak nama color
- Kita bisa liat dihalaman web ini untuk daftar warna apa saja yang didukung oleh HTML
- <https://htmlcolorcodes.com/color-names/>

KODE : COLOR NAME




```
<style>
  h1 {
    color:  red;
  }
  h2 {
    color:  indianred;
  }
  p {
    color:  salmon;
  }
</style>
```

```
<body>
  <h1>Ini Title</h1>
  <h2>Ini Sub Title</h2>
  <p>Ini Isi Artikel</p>
</body>
</html>
```

HEX, RGB DAN HSL

- Selain menggunakan Color Name, CSS juga mendukung color menggunakan format HEX, RGB dan HSL
- <https://colorpicker.me/>

KODE : COLOR

```
<style>
  h1 {
    color:  rgb(175, 34, 53);
  }
  h2 {
    color:  hsl(225, 82%, 44%);
  }
  p {
    color:  #10981d;
  }
</style>
```

```
<body>
  <h1>Ini Title</h1>
  <h2>Ini Sub Title</h2>
  <p>Ini Isi Artikel</p>
</body>
</html>
```



TEXT



TEXT

- CSS bisa digunakan untuk mengubah properties atau format untuk text
- Contoh sebelumnya kita sudah menggunakan color untuk mengubah warna text
- Selain color, masih banyak yang bisa kita ubah dari properties atau format untuk text

TEXT ALIGNMENT

- Text alignment digunakan untuk mengubah rata tulisan, kita bisa menggunakan rule text-align, dimana memiliki beberapa nilai seperti
- <https://developer.mozilla.org/en-US/docs/Web/CSS/text-align>
- left untuk rata kiri
- right untuk rata kanan
- center untuk rata tengah
- justify untuk rata kanan dan kiri

TEXT DECORATION

- Text Decoration digunakan untuk menambah garis dekorasi ke text.
- Kita bisa menggunakan aturan text-decoration
- <https://developer.mozilla.org/en-US/docs/Web/CSS/text-decoration>



TEXT TRANSFORMATION

- Text Transformation digunakan untuk mengubah uppercase dan lowercase untuk text
- Kita bisa menggunakan aturan text-transform
- <https://developer.mozilla.org/en-US/docs/Web/CSS/text-transform>

TEXT SPACING

- Text Spacing digunakan untuk mengatur jarak dalam text
- Aturan text-indent digunakan untuk mengatur jarak di awal text
- <https://developer.mozilla.org/en-US/docs/Web/CSS/text-indent>
- Aturan letter-spacing digunakan untuk mengatur jarak antar huruf
- <https://developer.mozilla.org/en-US/docs/Web/CSS/letter-spacing>
- Aturan line-height digunakan untuk mengatur jarak antar baris
- <https://developer.mozilla.org/en-US/docs/Web/CSS/line-height>
- Aturan word-spacing digunakan untuk mengatur jarak antar kata
- <https://developer.mozilla.org/en-US/docs/Web/CSS/word-spacing>
- Aturan white-space digunakan untuk mengatur bagaimana whitespace ditampilkan
- <https://developer.mozilla.org/en-US/docs/Web/CSS/white-space>

TEXT SHADOW

- Text Shadow digunakan untuk menambahkan efek bayangan pada text
- Untuk menambahkan Text Shadow, kita bisa menggunakan aturan text-shadow
- <https://developer.mozilla.org/en-US/docs/Web/CSS/text-shadow>

REFERENSI

- Untuk melihat lebih detail tentang text, kita bisa lihat referensi nya dihalaman ini :
- https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_text
- https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_text_decoration



FONT

FONT

- Saat membuat tulisan, biasanya Web Browser akan menggunakan default font nya, tiap Web Browser memiliki default Font masing-masing
- Saat membuat web, baiknya kita menggunakan font yang sama sehingga tampilan web kita konsisten
- Kita bisa mengatur Font di halaman HTML menggunakan CSS dengan property font
- <https://developer.mozilla.org/en-US/docs/Web/CSS/font>

FONT FAMILY

- Untuk mengubah jenis font, kita bisa menggunakan property font-family
- <https://developer.mozilla.org/en-US/docs/Web/CSS/font-family>
- Menggunakan font-family sangat tergantung dengan sistem operasi yang digunakan, jika font nya tidak ada di sistem operasi yang digunakan, maka hasilnya tidak akan sesuai dengan yang kita mau
- Kita bisa menggunakan generic font family yang sudah menjadi standar untuk CSS
- Kita bisa melihat daftar generic family name disini :
<https://www.w3.org/TR/css-fonts-3/#generic-font-families>

FONT STYLE

- Untuk mengubah style dari font, kita bisa menggunakan property font-style
- <https://developer.mozilla.org/en-US/docs/Web/CSS/font-style>
- Untuk mengubah ketebalan dari font, kita bisa menggunakan property font-weight
- <https://developer.mozilla.org/en-US/docs/Web/CSS/font-weight>



FONT SIZE

- Untuk mengubah ukuran dari font, kita bisa menggunakan property font-size
- <https://developer.mozilla.org/en-US/docs/Web/CSS/font-size>

GOOGLE FONT

- Salah satu penyedia font gratis yang bisa kita gunakan adalah Google Font
- Kita bisa mencari font di Google Font, lalu menggunakan CSS untuk menambahkan font yang kita mau di halaman Web yang kita mau
- <https://fonts.google.com/>

KODE : GOOGLE FONT

```
<title>Google Font</title>
<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Agbalumo">
<style>
  .title {
    font-family: 'Agbalumo';
  }
</style>
</head>
<body>
  <h1 class="title">Selamat belajar css font</h1>
</body>
```



BACKGROUND

BACKGROUND

- CSS bisa digunakan untuk mengubah latar belakang / background dari tiap element di HTML
- Kita bisa menggunakan property background
- <https://developer.mozilla.org/en-US/docs/Web/CSS/background>

BACKGROUND COLOR

- Background yang paling sederhana Background Color, yaitu mengubah background element menjadi color
- Sama seperti property color, background color juga mendukung format color HEX, RGB dan HSL
- Kita bisa menggunakan property background-color untuk mengubah background menggunakan color
- <https://developer.mozilla.org/en-US/docs/Web/CSS/background-color>



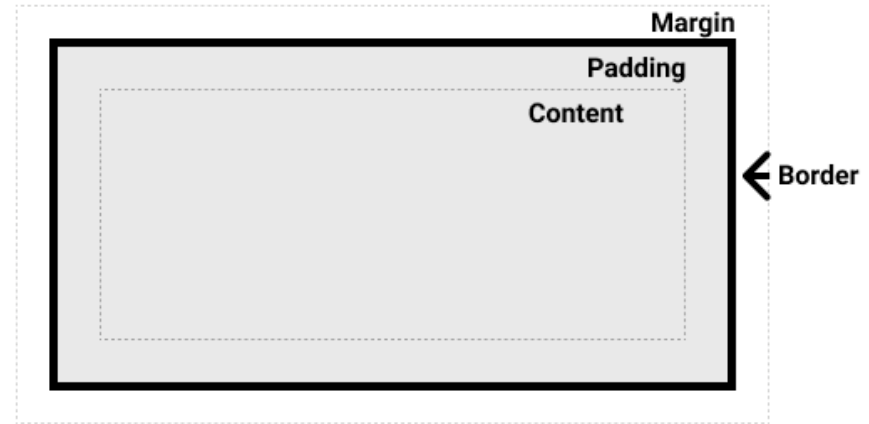
BOX MODEL

BOX MODEL

- Saat menggunakan CSS, terdapat konsep bernama Box Model
- Biasanya digunakan ketika kita ingin mengatur tata letak / layout pada element
- Setiap element di HTML, memiliki Box yang terdiri dari content, padding, border dan margin

GAMBARAN BOX MODEL

- Content adalah isi dari element
- Padding adalah arena transparan antara content dan border
- Border adalah kotak yang mengelilingi content
- Margin adalah arena transparan paling luar setelah border



SIZE

- Saat kita membuat element HTML, kita tahu bahwa beberapa element ditampilkan dalam block, dan beberapa element ditampilkan dalam inline
- Dengan menggunakan CSS, kita bisa mengubah ukuran dari tiap element menggunakan height dan width
- <https://developer.mozilla.org/en-US/docs/Web/CSS/height>
- <https://developer.mozilla.org/en-US/docs/Web/CSS/width>

PADDING

- Untuk mengatur area Padding, kita bisa menggunakan property padding dengan CSS
- <https://developer.mozilla.org/en-US/docs/Web/CSS/padding>



MARGIN

- Untuk mengatur area Margin, kita bisa menggunakan property margin dengan CSS
- <https://developer.mozilla.org/en-US/docs/Web/CSS/margin>



DEBUG DENGAN BROWSER

DEBUG DENGAN BROWSER

- Salah satu cara saat kita ingin melihat Box Model pada halaman HTML yaitu dengan menggunakan Web Browser
- Tiap Web Browser biasanya ada fitur untuk melihat detail dari informasi halaman HTML yang sedang dibuka
- Contohnya di Google Chrome, kita bisa klik kanan element yang ingin kita lihat, lalu pilih menu Inspect Element



MIN DAN MAX SIZE



MIN DAN MAX SIZE

- Saat menambahkan size pada element, kadang kita ingin menentukan minimal atau maksimal dari size element, hal ini untuk menjaga element tidak terlalu kecil atau tidak terlalu besar
- Kita juga bisa mengaturnya menggunakan property di CSS

MIN SIZE

- Untuk mengatur minimal kita bisa menggunakan property
- min-height untuk minimal tinggi
- <https://developer.mozilla.org/en-US/docs/Web/CSS/min-height>
- min-width untuk minimal lebar
- <https://developer.mozilla.org/en-US/docs/Web/CSS/min-width>

MAX SIZE

- Untuk mengatur maksimal kita bisa menggunakan property
- max-height untuk minimal tinggi
- <https://developer.mozilla.org/en-US/docs/Web/CSS/max-height>
- max-width untuk minimal lebar
- <https://developer.mozilla.org/en-US/docs/Web/CSS/max-width>



BORDER

BORDER

- Dalam Box Model, terdapat bagian Border
- Untuk mengubah Border, kita bisa menggunakan property border
- <https://developer.mozilla.org/en-US/docs/Web/CSS/border>

BORDER DETAIL

- Kadang, kita ingin mengubah border tiap sisi berbeda, untuk itu kita bisa menggunakan property border-right, border-left, border-top dan border-bottom
- <https://developer.mozilla.org/en-US/docs/Web/CSS/border-left>
- <https://developer.mozilla.org/en-US/docs/Web/CSS/border-right>
- <https://developer.mozilla.org/en-US/docs/Web/CSS/border-bottom>
- <https://developer.mozilla.org/en-US/docs/Web/CSS/border-top>

BORDER RADIUS

- Border Radius adalah untuk mengubah putaran dari ujung border
- Secara otomatis isi padding pun akan mengikuti putaran dari border
- Kita bisa menggunakan property border-radius
- <https://developer.mozilla.org/en-US/docs/Web/CSS/border-radius>



OUTLINE

OUTLINE

- Outline mirip seperti Border, lokasinya berada setelah Border
- Yang membedakan dari Border, Outline tidak mengambil area dari Box, sehingga tidak mengganggu ukuran layout / tata letak
- <https://developer.mozilla.org/en-US/docs/Web/CSS/outline>



BACKGROUND IMAGE

BACKGROUND IMAGE

- Selain menggunakan Color, kita juga bisa menggunakan Image sebagai background
- Untuk menggunakan image sebagai background kita bisa menggunakan property background-image
- <https://developer.mozilla.org/en-US/docs/Web/CSS/background-image>

GRADIENT

- Saat menggunakan background-image, selain menggunakan gambar, kita juga bisa menggunakan warna gradient
- Terdapat banyak sekali jenis warna gradient yang didukung oleh CSS
- <https://developer.mozilla.org/en-US/docs/Web/CSS/gradient>
- Untuk mencoba warna-warna gradient, kita bisa menggunakan website : <https://cssgradient.io/>



OPACITY

OPACITY

- Element di HTML bisa diatur transparansinya menggunakan property opacity
- <https://developer.mozilla.org/en-US/docs/Web/CSS/opacity>
- Perlu diperhatikan, saat menggunakan opacity, semua element baik itu background dan content nya, akan berubah menjadi transparan



VALUE DAN UNIT

VALUE DAN UNIT

- Saat kita menggunakan CSS, kita sering sekali menggunakan berbagai jenis value, dari mulai number, text, color name, rgb, image, dan lain-lain
- Semua standarisasi dari CSS Value dan Unit sudah ditetapkan di spesifikasi berikut :
- <https://drafts.csswg.org/css-values/>

LENGTH

- Salah satu value yang penting kita mengerti adalah Value Length
- Sebelumnya kita sering menggunakan px untuk pixel, selain px masih banyak type yang bisa kita gunakan
- <https://developer.mozilla.org/en-US/docs/Web/CSS/length>



WRITING MODE

WRITING MODE

- Writing mode digunakan untuk menentukan cara membaringkan text, apakah ingin horizontal, atau vertical
- Kita bisa mengubahnya menggunakan property writing-mode
- <https://developer.mozilla.org/en-US/docs/Web/CSS/writing-mode>



OVERFLOW

OVERFLOW

- Overflow adalah kejadian dimana konten text melebihi ukuran dari Box element
- Kita bisa menentukan bagaimana cara menampilkan ketika terjadi Overflow menggunakan property overflow
- <https://developer.mozilla.org/en-US/docs/Web/CSS/overflow>



PSEUDO-ELEMENTS SELECTOR

PSEUDO-ELEMENTS SELECTOR

- Pseudo-elements Selector adalah kata kunci untuk menambah selector ke bagian tertentu dari element yang terseleksi
- Saat memilih pseudo-elements, kita harus gunakan pemisah tanda :: (titik dua sebanyak dua kali)
- Ada banyak pseudo-elements, kita bisa lihat detailnya di halaman dokumentasinya :
- <https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-elements>



PSEUDO-CLASSES SELECTOR

PSEUDO-CLASSES SELECTOR

- Pseudo-classes Selector adalah kata kunci yang ditambahkan ke selector yang merepresentasikan state / keadaan tertentu dari element yang diseleksi
- Untuk menambahkan pseudo-classes, kita bisa gunakan : (titik dua)
- Kita bisa lihat daftar pseudo-classes yang tersedia di halaman dokumentasinya :
- <https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes>



LINK

LINK

- Saat kita membuat web, kita akan sering sekali menggunakan Link menggunakan tag a
- Link memiliki banyak sekali pseudo-classes yang bisa kita gunakan untuk mengubah tampilan Link pada state tertentu
- https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes#user_action_pseudo-classes
- https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes#location_pseudo-classes



LIST



LIST

- Saat membuat daftar, kita sering menggunakan element list
- CSS memiliki beberapa property yang bisa digunakan untuk mengubah List

CSS LIST

- Property list-style-type bisa kita gunakan untuk mengubah jenis list yang ditampilkan di tiap item
- <https://developer.mozilla.org/en-US/docs/Web/CSS/list-style-type>
- Property list-style-image bisa kita gunakan untuk mengubah item list dalam bentuk gambar
- <https://developer.mozilla.org/en-US/docs/Web/CSS/list-style-image>
- Property list-style-position bisa kita gunakan untuk mengubah posisi item dalam list
- <https://developer.mozilla.org/en-US/docs/Web/CSS/list-style-image>
- Atau kita bisa menggunakan shortcut menggunakan list-style
- <https://developer.mozilla.org/en-US/docs/Web/CSS/list-style>



TABLE



TABLE

- Saat kita membuat element Table di HTML, kita akan banyak menggunakan Box Model

TABLE BORDER

- Jika kita ingin mengatur border pada tabel, kita bisa menggunakan Border pada table, tr, th, td, dan lain-lain
- <https://developer.mozilla.org/en-US/docs/Web/CSS/border>
- Saat menggunakan border, kadang tiap kolom akan ada jarak, jika kita ingin menghilangkannya, kita bisa menggunakan border-collapse :
- <https://developer.mozilla.org/en-US/docs/Web/CSS/border-collapse>
- Atau kita bisa juga tambahkan padding pada bagian-bagian tabel

TABLE SIZE

- Untuk mengatur ukuran table, kita bisa menggunakan property width dan height
- <https://developer.mozilla.org/en-US/docs/Web/CSS/height>
- <https://developer.mozilla.org/en-US/docs/Web/CSS/width>

TABLE ALIGNMENT

- Untuk mengatur posisi text pada tabel, kita bisa menggunakan property text-align atau writing-mode
- <https://developer.mozilla.org/en-US/docs/Web/CSS/text-align>
- <https://developer.mozilla.org/en-US/docs/Web/CSS/writing-mode>



FORM

FORM

- Saat kita membuat Form, element input yang kita buat, semua bisa diatur menggunakan Box Model di CSS
- Selain itu, khusus Form, terdapat pseudo-classes yang dikhususkan untuk input form :
- https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes#input_pseudo-classes



CONTENT

CONTENT

- Content adalah property yang bisa digunakan untuk mengubah isi dari element
- <https://developer.mozilla.org/en-US/docs/Web/CSS/content>
- Untuk mengganti isi Element menggunakan property content, kita hanya bisa menggunakan content Image, sedangkan untuk text, bisa kita lakukan di pseudo-elements
- https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_pseudo-elements



COUNTER

COUNTER

- CSS memiliki kemampuan untuk membuat Counter, ini sangat cocok untuk membuat penomoran tanpa kita harus lakukan secara manual
- Kita bisa gunakan property counter-set untuk membuat counter
- <https://developer.mozilla.org/en-US/docs/Web/CSS/counter-set>
- Property counter-increment untuk menaikkan counter
- <https://developer.mozilla.org/en-US/docs/Web/CSS/counter-increment>
- Dan property counter-reset untuk melakukan reset counter
- <https://developer.mozilla.org/en-US/docs/Web/CSS/counter-reset>
- Untuk mengambil nilai counter, bisa menggunakan function counter()
- <https://developer.mozilla.org/en-US/docs/Web/CSS/counter>



FILTER

FILTER

- CSS bisa digunakan untuk menambahkan filter di element yang kita inginkan
- Kita bisa menggunakan property filter
- <https://developer.mozilla.org/en-US/docs/Web/CSS/filter>
- Terdapat banyak sekali jenis filter yang bisa kita gunakan



IMPORT

IMPORT

- Sebelumnya, kita sudah tahu untuk menambahkan kode CSS dari external file, kita bisa menggunakan tag link
- CSS juga memiliki kemampuan untuk mengambil kode CSS dari luar menggunakan At-Rule, dimana harus ditempatkan dibagian atas kode CSS
- Kita bisa mengambil kode CSS dari external menggunakan rule @import
- <https://developer.mozilla.org/en-US/docs/Web/CSS/@import>



CASCADE

CASCADE

- Saat kita menggunakan CSS, kita harus tahu konsep bernama Cascade
- Cascade adalah logika bagaimana web browser mengumpulkan property CSS dari berbagai sumber sebelum diterapkan ke element yang dipilih
- Web browser akan mengambil sumber property CSS dari tiga sumber, yaitu User Agent, Author dan User
- <https://developer.mozilla.org/en-US/docs/Web/CSS/Cascade>

USER AGENT STYLESHEET

- User Agent atau browser, biasanya memiliki nilai awal untuk stylesheet
- Tiap User Agent biasanya berbeda, oleh karena itu saat membuat CSS kita harus hati-hati dengan nilai awal User Agent, karena bisa berbeda untuk tiap browser
- Salah satu contoh yang banyak dilakukan oleh programmer web adalah, melakukan reset ke nilai kosong
- Contohnya seperti ini :
<https://github.com/necolas/normalize.css/blob/master/normalize.css>

AUTHOR STYLESHEET

- Saat membuat web, kadang biasanya kita tidak langsung membuat file CSS, biasanya kita akan menggunakan stylesheet buatan orang lain, contoh yang populer adalah Bootstrap atau TailwindCSS
- <https://getbootstrap.com/>
- <https://tailwindcss.com/>
- Biasanya User akan melakukan import atau link untuk Author Stylesheet



USER STYLESHEET

- User stylesheet adalah yang kita buat sendiri, biasanya dibuat untuk mengubah stylesheet yang sudah digunakan baik itu dari User Agent atau Author

CASCADE ORDER

- Saat kita menggunakan sumber-sumber tadi untuk membuat CSS, Web Browser akan menggabungkan semua property CSS untuk element menggunakan urutan sebagai berikut
 - ← User Agent Stylesheet
 - ← Author Stylesheet
 - ← User Stylesheet
- Jika ternyata kita membuat aturan yang sama di CSS, maka urutan posisi yang paling akhir yang akan digunakan

!IMPORTANT

- Saat kita membuat aturan CSS, semua aturan akan mendapatkan prioritas normal
- Jika kita ingin membuat sebuah aturan menjadi sangat penting, sehingga tidak boleh digantikan setelahnya oleh aturan lain, maka kita bisa menambahkan !important pada aturan tersebut
- <https://developer.mozilla.org/en-US/docs/Web/CSS/important>



LAYER



MASALAH DENGAN !IMPORTANT

- Menggunakan !important sangat tidak flexible, karena artinya kita tidak bisa mengubah propertinya lagi
- Alternatif lain yang lebih flexible adalah menggunakan @layer

@LAYER

- @layer adalah fitur di dalam CSS Cascade, dimana kita bisa membuat layer (seperti tumpukan), dimana posisi layer bisa diurutkan sesuai dengan yang kita mau
- Dengan menggunakan @layer, kita bisa menentukan mana yang lebih penting dan mana yang tidak begitu penting dengan mengubah posisi layer
- Dengan begitu, kita bisa mengubah-ubah posisi, tanpa harus menggunakan !important lagi
- <https://developer.mozilla.org/en-US/docs/Web/CSS/@layer>



MATERI SELANJUTNYA



MATERI SELANJUTNYA

- CSS Layout
- JavaScript