

{CSS}

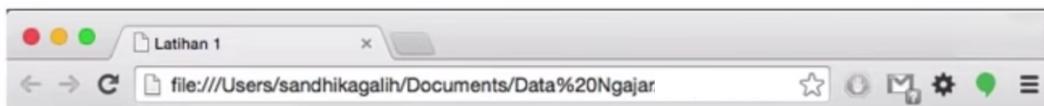
Cascading Style Sheet

“

mekanisme sederhana yang mengatur gaya / *style* (cth: warna, ukuran, posisi, dll) pada halaman web.

sebelum **CSS**

sebelum CSS



Hello World

```
<font size="+6" color="lightblue" face="arial"><center><h1>Hello World</h1></center></font>
```

sebelum CSS

```
<font size="+6" color="lightblue" face="arial">
  <center>
    <h1>Hello World</h1>
  </center>
</font>
```

dengan **CSS**

dengan CSS

HTML

```
<h1>Hello World</h1>
```



Hello World

CSS

```
h1 {  
    font-size: 120px;  
    font-family: arial;  
    color: lightblue;  
    text-align: center;  
}
```

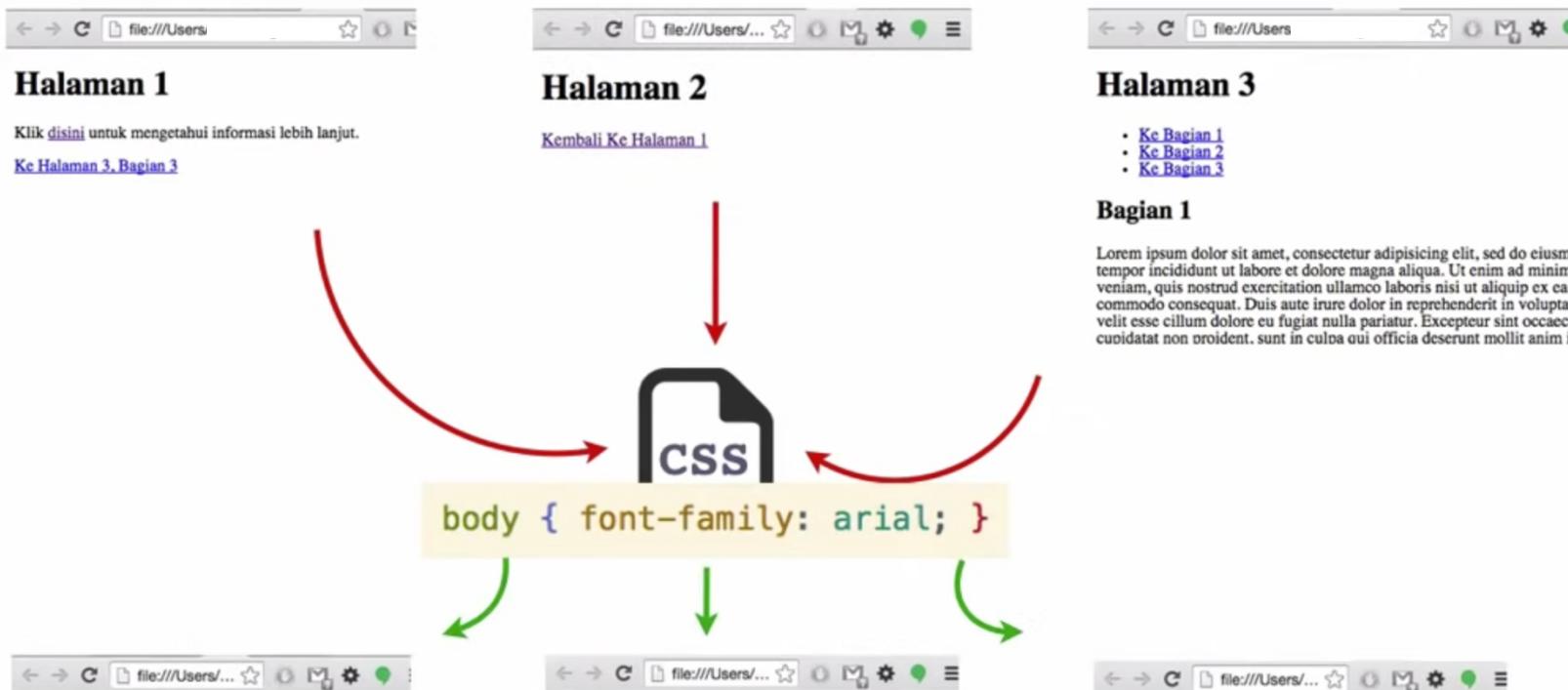
{CSS}

Cascading Style Sheet

Cascading Style Sheet

- Aturan yang digunakan untuk menampilkan elemen / tag HTML
- Dibuat terpisah dengan HTML
- Bertujuan untuk memisahkan konten dan style
- 1 CSS dapat digunakan untuk banyak halaman HTML
- 1 halaman HTML dapat terlihat berbeda jika menggunakan CSS yang berbeda pula

1 CSS dapat digunakan di banyak halaman



{CSS}
anatomy

```
selector { property: value; }
```

```
selector { property: value; }
```

selector

A diagram illustrating the structure of a CSS rule. The selector 'h1' is highlighted in purple. The opening brace '{' is highlighted in green. The property 'color' is highlighted in green. The value 'blue' is highlighted in olive green. A red curly arrow labeled 'selector' points to the purple 'h1'. A red curly arrow labeled 'value' points to the green 'blue'. A red curly arrow labeled 'property' points to the green 'color'.

```
h1 { color: blue; }
```

```
h1 {  
    font-family: "Courier New";  
    text-align: center;  
    font-size: 30px;  
    color: blue;  
}
```

Selector

- Digunakan untuk memilih dan memanipulasi elemen spesifik pada HTML.
- Elemen HTML dipilih berdasarkan tag, id, class bahkan pola / pattern.
- Semakin kompleks struktur HTML, selector juga bisa semakin kompleks / spesifik

Selector

```
h1 { color: blue; }
```

CSS, pilih seluruh elemen h1, lalu
ubah teks didalamnya menjadi
berwarna biru

Property & Value

background-attachment	elevation
background-color	empty-cells
background-image	float
background-position	font
background-repeat	font-family
border	font-size
border-collapse	font-size-adjust
border-color	font-stretch
border-spacing	font-style
border-style	font-variant
border-top	font-weight
border-right	height
border-bottom	left

Property & Value

background-attachment	elevation
background-color	empty-cells
background-image	float
background-position	font
background-repeat	font-family
border	font-size
border-collapse	font-size-adjust
border-color	font-stretch
border-spacing	font-style
border-style	font-variant
border-top	font-weight
border-right	height
border-bottom	left

350++

menempatkan
{CSS}

- embed

```
<style></style>
```

- inline

```
<p style="color: lightblue;"> ... </p>
```

- external

```
<link rel="stylesheet" href="style.css">
```

Embed

```
<!DOCTYPE html>
<html>
<head>
    <title>Coba CSS</title>
    <style>
        body { font-family: arial; }
        h1 { color: lightblue; }
        p {
            line-height: 1.6em;
            color: grey;
        }
    </style>
</head>
<body>

    <h1>Hello World</h1>

    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non
proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>

</body>
</html>
```

Inline

```
<!DOCTYPE html>
<html>
<head>
    <title>Coba CSS</title>
</head>
<body>

    <h1 style="text-align: center; font-size: 60px;">Hello World</h1>

    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non
proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>

</body>
</html>
```

external



external.html



style.css

font
styling

Font

font-family

mengatur jenis font yang akan digunakan

font-size

mengatur ukuran font

font-weight

mengatur ketebalan font

font-variant

mengubah font menjadi *small caps*

Font

font-style

mengubah font menjadi bercetak miring

line-height

mengatur spasi antar baris

font-family

nama font

generic family

font-family

nama font

generic family

```
body {  
    font-family: arial, verdana, sans-serif;  
}  
  
p {  
    font-family: "Times New Roman", serif;  
}
```



The diagram shows a block of CSS code with two annotations. A red bracket labeled 'nama font' covers the font-family declaration in the body rule, which lists 'arial', 'verdana', and 'sans-serif'. Another red bracket labeled 'generic family' covers the font-family declaration in the p rule, which lists '\"Times New Roman\"' and 'serif'.

Sans-serif

Serif

font-size

I

px

%

em

font-weight

lighter

I
normal

100-900

bold

bolder

font-variant

normal

small-caps

```
<h1>Hello World</h1>
```

Hello World
normal

HELLO WORLD
small-caps

font-style

normal

italic

oblique

font-style

oblique



italic



line-height

normal

px

em

CSS font

shorthand / *singkatan*

```
body {  
    font-style: italic;  
    font-variant: normal;  
    font-weight: bold;  
    font-size: 16px;  
    line-height: 18px;  
    font-family: helvetica, arial, sans-serif;  
}
```

```
body {  
    font: italic normal bold 16px/18px helvetica,arial, sans-serif;  
}
```

The diagram illustrates the structure of a CSS font declaration. It consists of several tokens connected by vertical lines. The tokens are color-coded: blue for 'body' and '{', green for 'font:', 'italic', 'normal', 'bold', '16px/18px', 'helvetica', 'arial', 'sans-serif', and red for the semicolon. The word 'optional' is written above the first four tokens ('font:', 'italic', 'normal', 'bold') in green. The word 'wajib' is written below the last two tokens ('16px/18px' and 'sans-serif') in red. A horizontal red line connects the '16px/18px' and 'sans-serif' tokens.

text
styling

Text

color

memberi warna pada tulisan

text-align^I

mengatur format paragraf / teks

text-indent

memberi indentasi pada paragraf / teks

text-decoration

mengatur dekorasi pada teks

text-transform

mengubah jenis huruf menjadi huruf besar, kecil / kapital

Text

letter-spacing

mengatur spasi ^I antar huruf

word-spacing

mengatur spasi antar kata

color

nama warna

`red, lightseagreen, royalblue, ...`

hexadecimal

`#ff0000, #20b2aa, #4169e1, ...`

rgb

`rgb(255,0,0), rgb(32,178,170), rgb(65,105,225), ...`

text-align

left

right

center

justify

text-indent

px

%

text-decoration

none

underline

overline

line-through

text-transform

none

lowercase

uppercase

capitalize

letter-spacing

normal

px

word-spacing

normal

px

CSS
background

Background

background-color

mengatur warna pada background

background-image

mengatur gambar yang akan digunakan pada background

background-position

mengatur posisi gambar pada background

background-repeat

mengatur jenis pengulangan gambar pada background

background-position

- top left
- top center
- top right
- center left
- center center
- center right
- bottom left
- bottom center
- bottom right
- x% y%
- x-pos y-pos

CSS background

shorthand / *singkatan*

```
background: color url() position repeat;
```

```
background: lightgreen url(bg.png) top left no-repeat;
```

CSS
selector basic

“

digunakan pada css untuk
mengenali sebuah elemen HTML
yang akan diberi style

selectors

elemen HTML

id

class

complex selector

id

- Sebuah elemen HTML hanya boleh memiliki 1 id
- Setiap halaman hanya boleh memiliki 1 elemen dengan id tersebut
- Dapat digunakan sebagai penanda halaman untuk link
- Digunakan juga untuk javascript
- Sebaiknya tidak digunakan untuk CSS (lebih baik gunakan class)

CSS
pseudo-class

“

kelas semu yang dimiliki oleh sebuah elemen HTML, yang membuat kita dapat mendefinisikan style pada “keadaan tertentu” dari elemen tersebut.

pseudo-class yang berhubungan dengan link

:link

style default pada sebuah link (a yang memiliki href)

:hover

style ketika kursor mouse berada diatas sebuah link / elemen

:active

style ketika sebuah link di-klik (keadaan aktif)

:visited

style ketika sebuah link sudah pernah dikunjungi sebelumnya
(menggunakan browser yang sama)

CSS

inheritance (pewarisan)

“

sebuah elemen mewarisi
beberapa nilai dari properti
yang dimiliki oleh elemen
parent-nya

properti yang diwariskan

color

font

letter-spacing

line-height

list-style

text-align

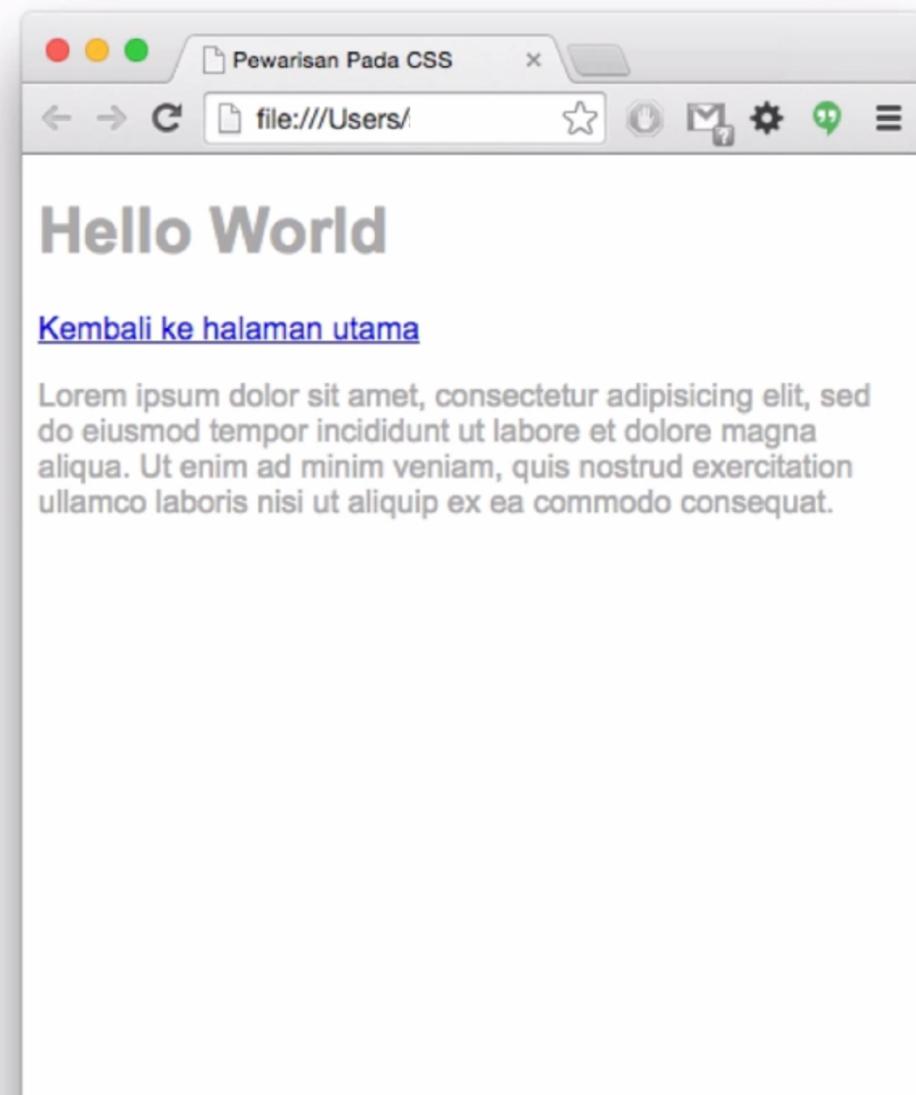
text-indent

text-transform

visibility

white-space

elemen **< a >** tidak mewarisi
nilai dari properti **color**



```
body {  
    font-family: arial;  
    color: darkgrey;  
}  
  
a { color: inherit; }
```

CSS specificity

“

setiap deklarasi css (selector)
memiliki **berat** yang berbeda. Berat
tersebut menentukan seberapa
spesifik sebuah elemen dapat dipilih
oleh selector.

⚙ HTML

```
1 <p>Lorem ipsum dolor sit amet, consectetur adipisicing  
elit. Consequuntur, pariatur necessitatibus quia,  
magnum placeat, blanditiis asperiores quo voluptatibus  
ullam, eius aperiam! Adipisci ex odit neque sint,  
aperiam magni, sed aspernatur!</p>
```

⚙ CSS

```
9  
10 p { color: red; }  
11  
12 p { color: green; }  
13
```

⚙ HTML

```
1 <p>Lorem ipsum dolor sit amet, consectetur adipisicing  
elit. Consequuntur, pariatur necessitatibus quia,  
magnam placeat, blanditiis asperiores quo voluptatibus  
ullam, eius aperiam! Adipisci ex odit neque sint,  
aperiam magni, sed aspernatur!</p>
```

⚙ CSS

```
9  
10 p { color: red; }  
11  
12 p { color: green; }  
13
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Consequuntur,
 pariatur necessitatibus quia, magnam placeat, blanditiis asperiores quo
 voluptatibus ullam, eius aperiam! Adipisci ex odit neque sint, aperiam
 magni, sed aspernatur!

⚙ HTML

```
1 <p id="p1">Lorem ipsum dolor sit amet, consectetur  
adipisicing elit. Consequuntur, pariatur  
necessitatibus quia, magnam placeat, blanditiis  
asperiores quo voluptatibus ullam, eius aperiam!  
Adipisci ex odit neque sint, aperiam magni, sed  
aspernatur!</p>
```

⚙ CSS

```
10 #p1 { color: red; }  
11  
12 p { color: green; }
```

HTML

```
1 <p id="p1">Lorem ipsum dolor sit amet, consectetur  
adipisicing elit. Consequuntur, pariatur  
necessitatibus quia, magnam placeat, blanditiis  
asperiores quo voluptatibus ullam, eius aperiam!  
Adipisci ex odit neque sint, aperiam magni, sed  
aspernatur!</p>
```

css

```
9  
10 #p1 { color: red; }  
11  
12 p { color: green; }
```

Consequuntur, pariatur necessitatibus quia, magnam placeat, blanditiis asperiores quo voluptatibus ullam, eius aperiam! Adipisci ex odit neque sint, aperiam magni, sed aspernatur!

#p1 lebih spesifik dari **p**

CSS LAYOUTING

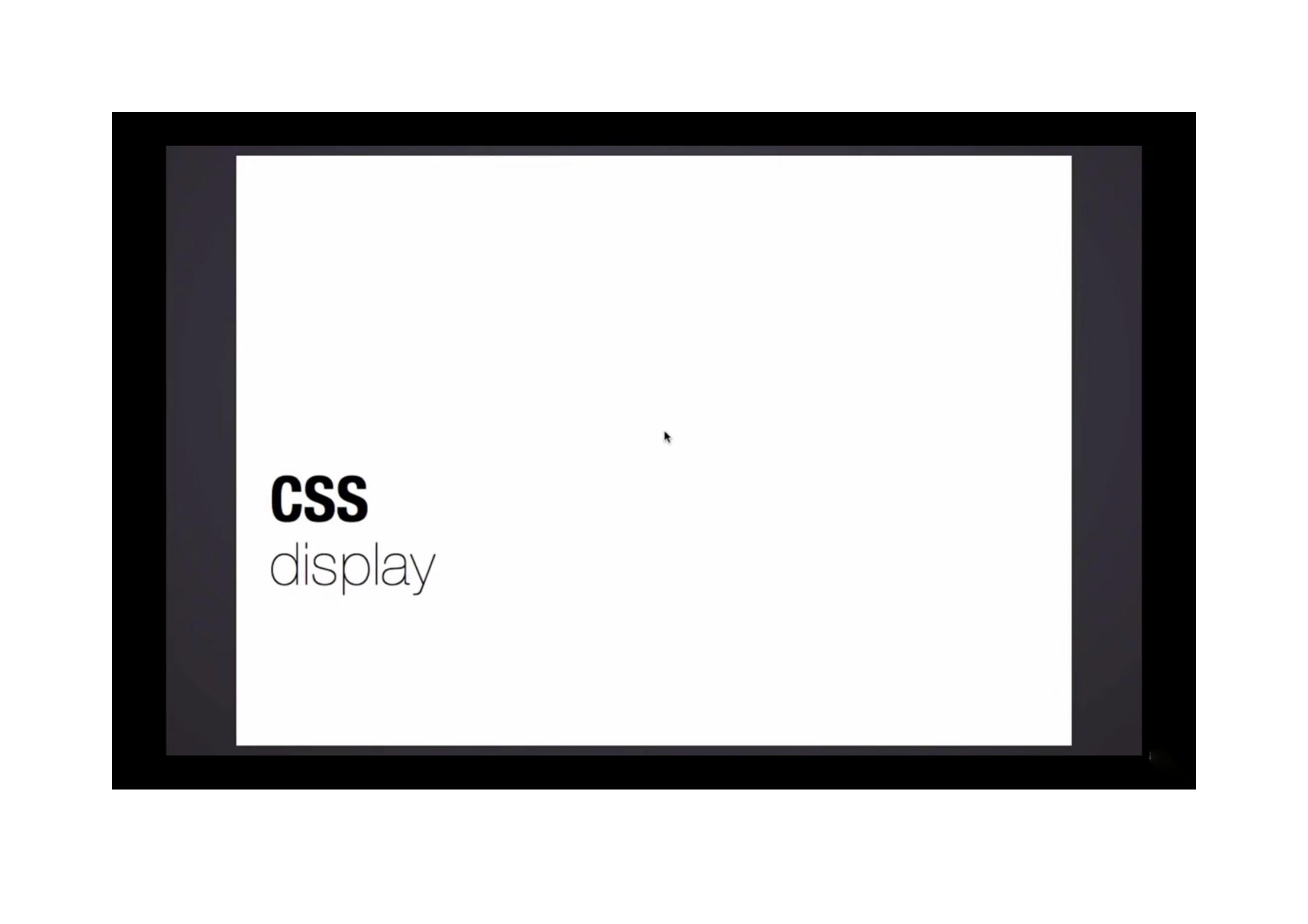
PENGATURAN TATA LETAK MENGGUNAKAN CSS



DISPLAY

MEMAHAMI PROPERTY DISPLAY PADA CSS





CSS
display

**<div> & **

“

tag pada HTML digunakan untuk memberikan 'maksud' / 'arti' pada sebuah konten (**p** untuk paragraf, **h1** untuk heading utama, dll).

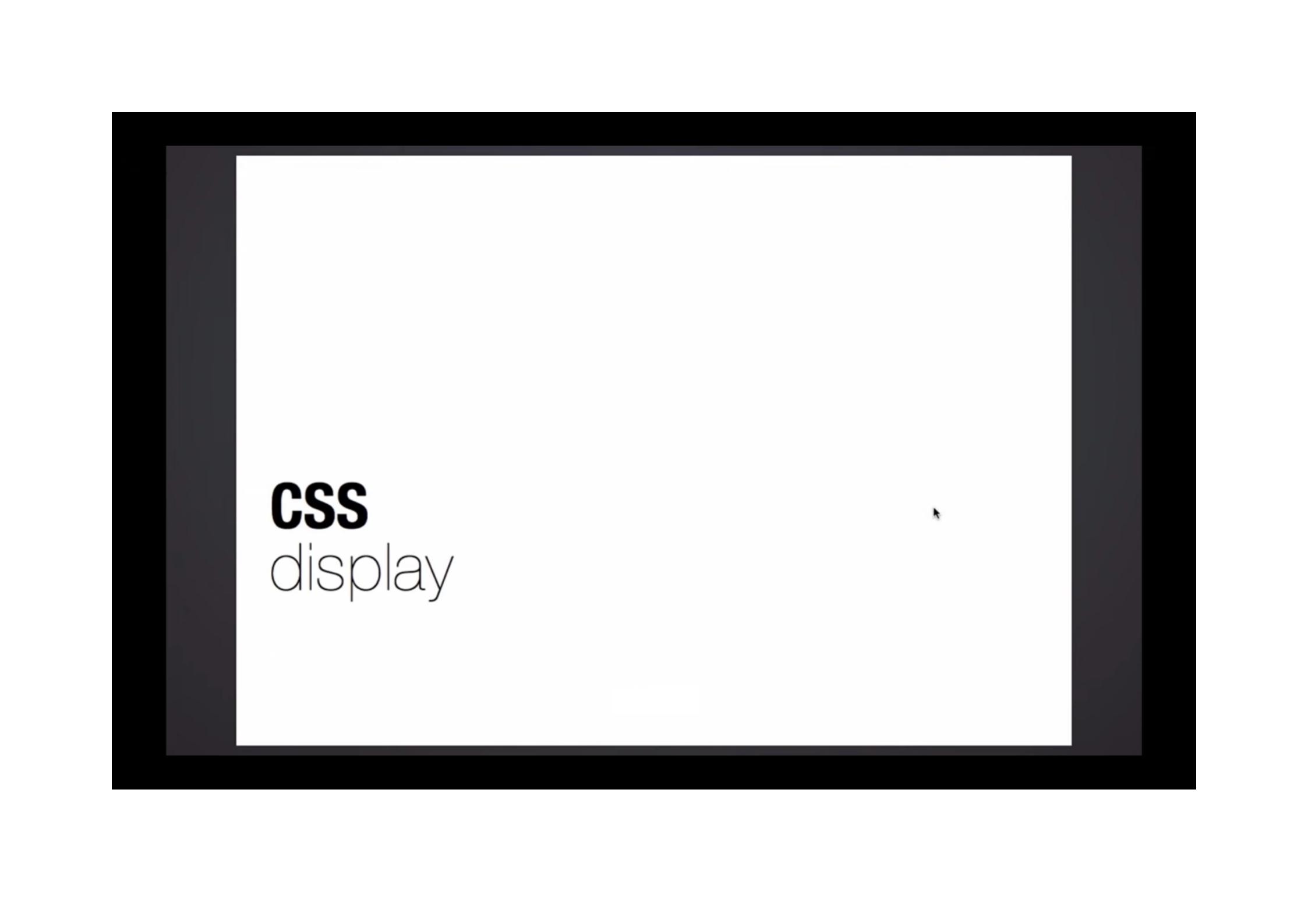
Tag **<div>** dan **** tidak memiliki 'arti' apapun, keduanya digunakan untuk mengelompokkan tag-tag HTML dan memberikan informasi terhadap tag-tag tersebut.

```
<div></div>
```

```
div {  
    display: block;  
}
```

```
<span></span>
```

```
span {  
    display: inline;  
}
```



CSS
display

“

setiap tag pada HTML berada di dalam sebuah **kotak**. properti **display** pada CSS mengatur perilaku dari kotak tersebut.

r

“

setiap tag pada HTML memiliki nilai default untuk properti **display**. Tetapi kita juga dapat mengubah perilaku dari tag tersebut dengan mengganti value-nya.

display

I

inline

inline-block

block

none

inline

- Elemen HTML yang secara default tidak menambahkan baris baru ketika dibuat
- Lebar dan tinggi elemennya sebesar konten yang ada di dalamnya
- Kita tidak dapat mengatur tinggi dan lebar dari elemen inline
- *Margin dan padding hanya mempengaruhi elemen secara horizontal, tidak vertikal*

elemen inline

b, strong, i, em,
a, span, sub, sup,
button, input, label,
select, textarea,

inline-block

- Tidak ada elemen yang secara default memiliki properti `display: inline-block;`
- Kita harus ubah secara manual properti tersebut
- Perilaku dasarnya sama dengan elemen inline
- Perbedaanya, elemen inline-block dapat kita atur tinggi dan lebar-nya

elemen block

h1-h6, p, ol, ul, li

form, hr, div,

...

I

block

- Elemen HTML yang secara default menambahkan baris baru ketika dibuat
- Jika tidak diatur lebar-nya, maka lebar default dari elemen block akan memenuhi lebar dari browser / parent-nya
- Kita dapat mengatur tinggi dan lebar dari elemen block
- Di dalam elemen block, kita dapat menyimpan tag dengan elemen inline, inline-block, atau bahkan elemen block lagi

none

Digunakan untuk menghilangkan sebuah elemen ⁱ

mengubah **display**

CSS LAYOUTING

#2 DIMENSI

CSS

Dimensi & Overflow

CSS
dimensi

width & height

I

px

%

in, cm, mm, pt, pc

CSS
overflow

overflow

I

visible

auto

hidden

scroll

CSS LAYOUTING BOX MODEL

“

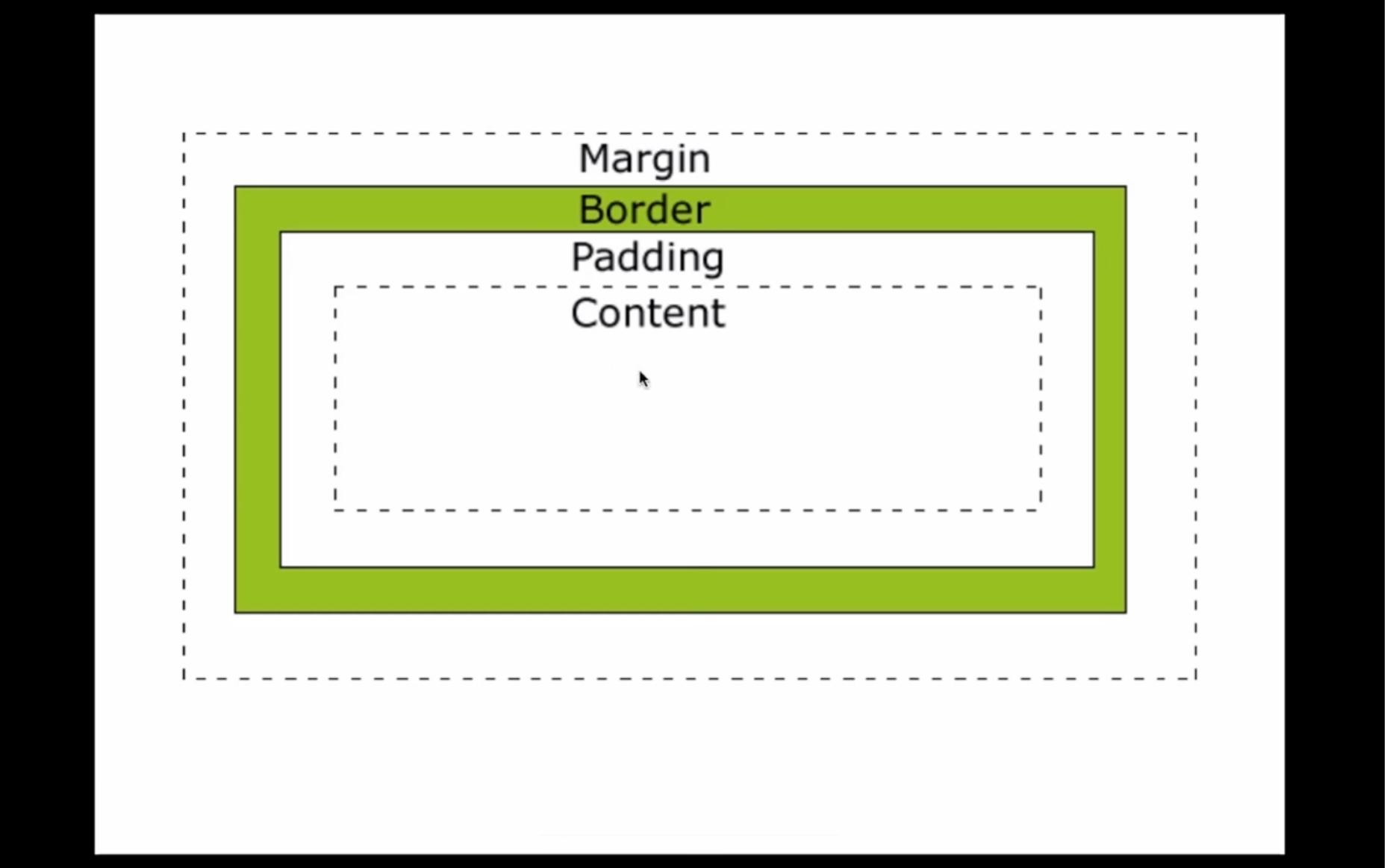
- setiap elemen di halaman berada di dalam sebuah box (kotak)
- kita bisa mengatur ukuran dan posisi kotak tersebut
- kita bisa memberi warna / gambar sebagai background kotak tersebut

“

CSS box model mendefinisikan ‘kotak’ yang dihasilkan oleh sebuah elemen, lalu menampilkannya sesuai dengan format visualnya.

“

CSS box model terdiri dari **margin**, **border**,
padding dan **content**.



The diagram illustrates the CSS box model, which defines the structure of an element's content area. It consists of four concentric rectangular layers: **Margin**, **Border**, **Padding**, and **Content**. The **Content** layer is the innermost white rectangle. The **Padding** layer is the thin white border around the content. The **Border** layer is the thick green border around the padding. The **Margin** layer is the outermost dashed black border. A small black cursor icon is positioned in the center of the content area.

Margin

Border

Padding

Content

CSS Box Model

margin

area transparan di sekitar kotak (di luar border)

padding

area transparan di dalam kotak (antara content dan border)

border

batas disekeliling content dan padding

content

konten sebenarnya di dalam box, bisa berupa teks atau gambar

properti untuk margin, padding & border

margin-top

padding-top

border-top

margin-right

padding-right

border-right

margin-bottom

padding-bottom

border-bottom

margin-left

padding-left

border-left

margin

padding

border

margin

margin

cara pakai

overlapping margin

negatif margin

auto

shorthand

margin-top

margin-right

margin-bottom

margin-left



margin

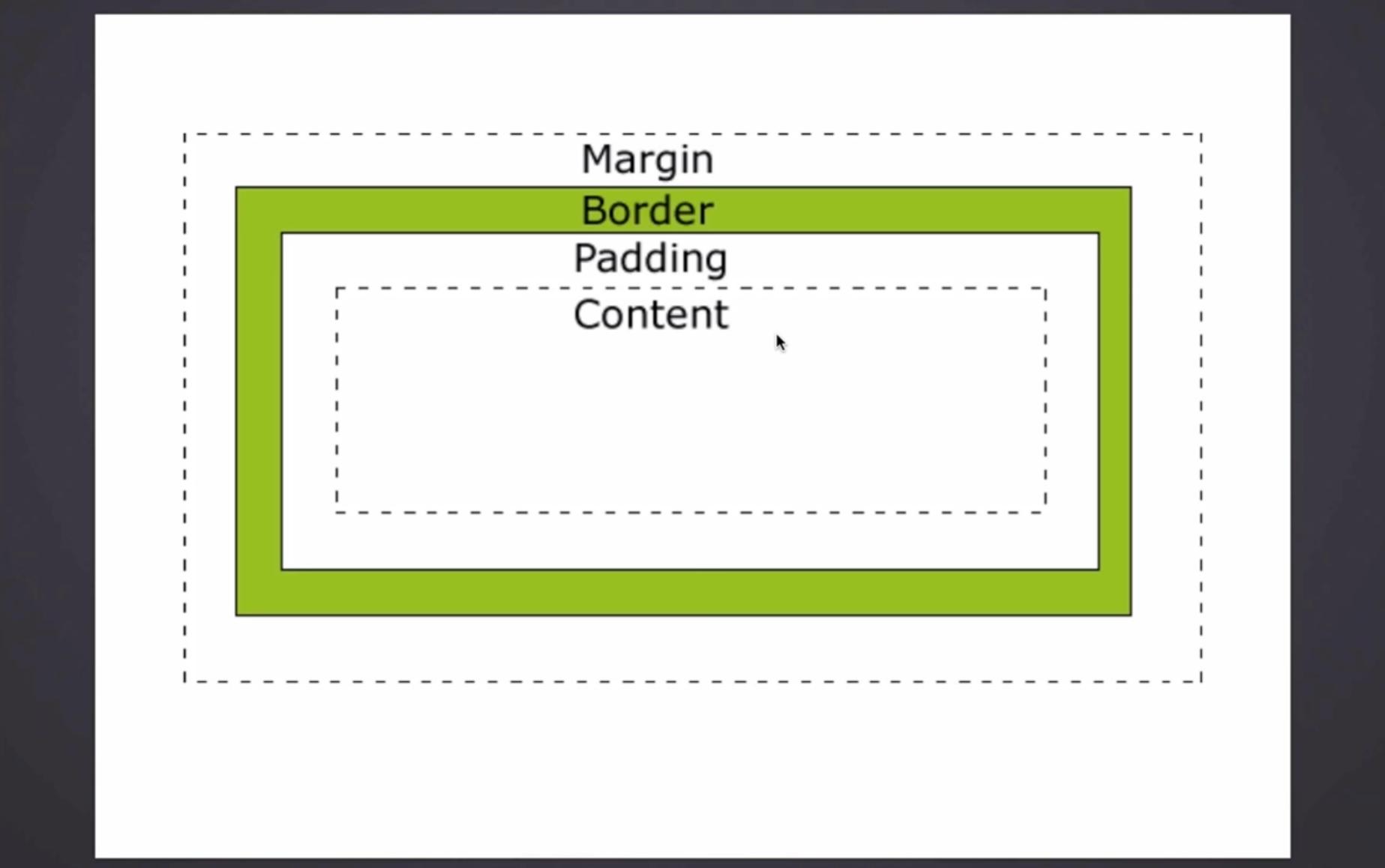
cara pakai

overlapping margin

negatif margin

auto

shorthand



The diagram illustrates the CSS box model, which defines the structure of an element's content area. It consists of four concentric rectangular layers: **Margin**, **Border**, **Padding**, and **Content**. The **Content** layer is the innermost white rectangle. The **Padding** layer is the thin white border around the content. The **Border** layer is the thick green border around the padding. The **Margin** layer is the outermost dashed black border.

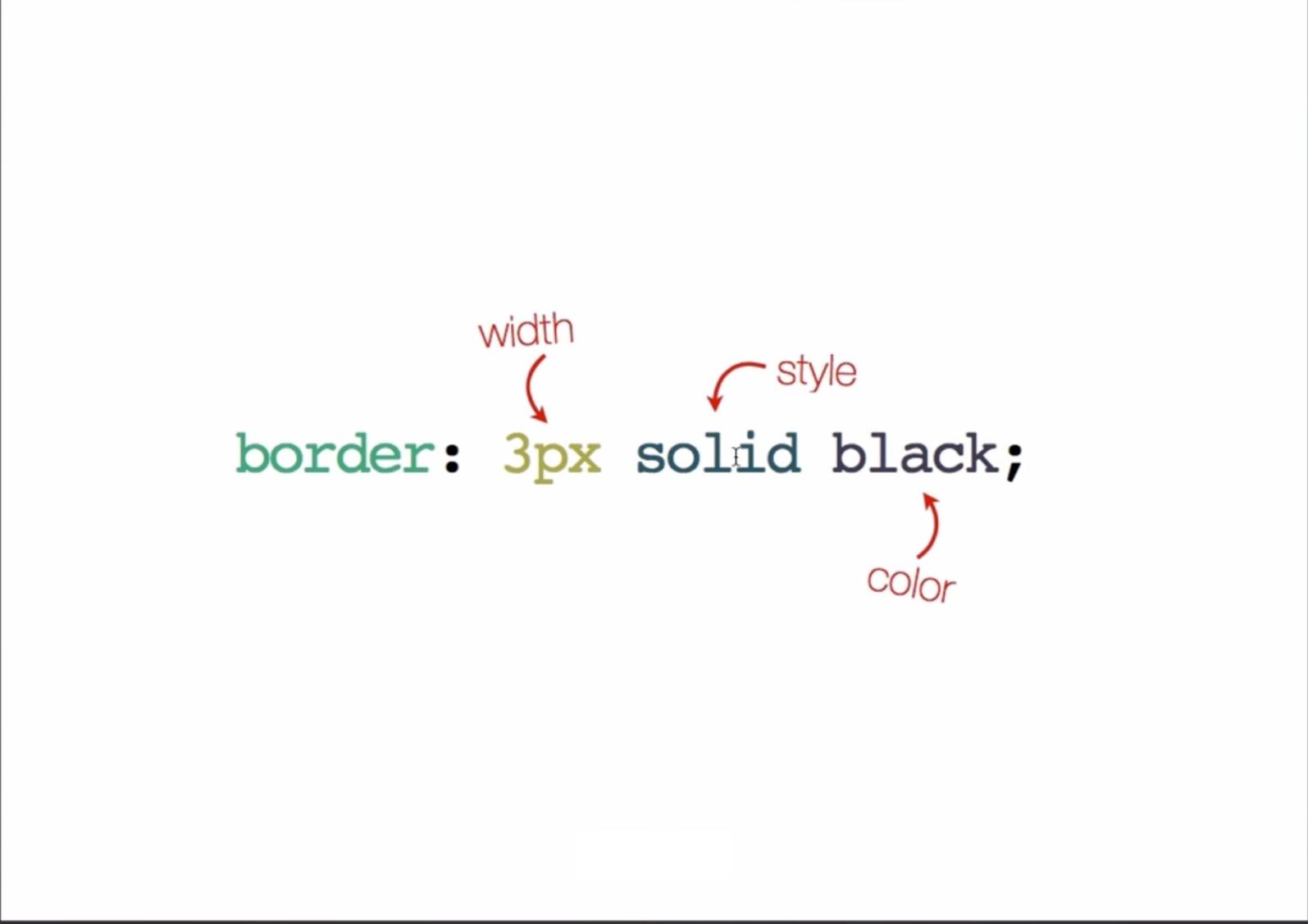
Margin

Border

Padding

Content

```
border: width style color;
```



```
border: 3px solid black;
```

width

style

)

color

The image shows a white rectangular box centered on a dark gray background. Above the box, there is a green line of CSS code: "border: 3px solid black;". Four red arrows point from the words "width", "style", ")", and "color" to the corresponding parts of the code: the "3px", the "solid", the closing parenthesis ")", and the "black" respectively.

style

solid

dotted

dashed

double

css reset

css float

“

Float merupakan properti pada CSS untuk mengatur posisi sebuah elemen. Sebuah elemen dapat dipaksa untuk berada di sebelah kiri atau kanan dari parent/pembungkusnya dengan menggunakan properti ini.

float

I
none

left

right



CSS position

position

static

relative

absolute

fixed

CSS Position

- **static** adalah nilai default dari tiap-tiap elemen ketika tidak diberi properti position
- Menggunakan position selain **static** (non-static), akan membuat sebuah elemen menjadi seolah-olah **berbeda dimensi** dari elemen lainnya
- Elemen yang diberi position selain **static** dapat menggunakan properti **top**, **left**, **bottom** dan **right** untuk mengatur posisinya

absolute

- Ketika kita menggerakkan elemen dengan posisi **absolute** (menggunakan properti **top**, **left**, **bottom** dan **right**), ruang yang ditempati oleh elemen tersebut dianggap **tidak ada**
- Ketika kita menggerakkan elemen dengan posisi absolute, elemen akan bergerak relatif terhadap posisi dari elemen parent-nya, **selama elemen parent-nya memiliki posisi yang juga non-static**
- Jika kita memberi properti **top: 0;** dan **left: 0;** maka elemen akan berada di ujung kiri atas dari elemen parent-nya

position: **fixed**;