

# CSS - Cafe Menu (11<sup>th</sup> of Jan 2024).

① Basic structure needed to start building your web page.

ex : <!DOCTYPE html>  
<html lang = "en"></html>

② Adding a **head** element, a **title** element.

ex : <head> → it is not visible on the Web page, but it is useful  
    <title>Cafe Menu</title> for search engine or how the  
    </head> page gets displayed.

③ Adding <meta charset = "utf-8">, nested inside the **head** element.

④ Adding **body** element below the **head** element.

ex : <head>  
    :  
    </head>  
    <body></body>

⑤ Adding a **main** element within the **body** element.

ex : <body>  
    <main></main>  
    </body>

⑥ Adding an **h1** element within the **main** element, and don't forget to add the name too.

ex : <h1> You ARE PRETTY </h1>

⑦ Adding a **p** element below the **h1** element with a text.

ex : <p> Est. 2000 </p>

⑧ Adding a **section** element within the **main** element.

ex : <section></section>

⑨ Creating an **h2** element in the **section** element.

⑩ Adding a **style** element within the **head** element.

ex : <head>  
    <style>  
    <meta>  
    <title></title>

    </style>  
    </head>



11 Adding style to an element by specifying it in the `style` element and setting a property for it like this :

```
element {  
    declaration: property: value;  
}
```

In this  
case =

selector  
h1 {  
 text-align: center;  
}

Note: the sign " " means there's a space.

12 Adding a new type selector for each `h2` & `p` element to the existing `style` element.

ex. `<style>`  
h1 {  
 text-align: center;  
}  
h2 {  
 text-align: center;  
}  
`</style>`

13 Atau kalau emang formatnya sama semua (property & value), bisa digabung :

`<style>` SPACE !!  
h1, h2, p {  
 text-align: center;  
}  
`</style>`

14 It will be easier to separate the styles in a new file. Consider it as a `styles.css` file. Rewrite the styles you have created into the `styles.css` file and exclude the opening and closing `style` tags.

15 Removing `style` element (and ofc the contents too). [Just delete it, bro].

16 Linking the `styles.css` by nesting a self-closing `link` element in the `head` element.

ex: `<link rel="stylesheet" href="styles.css"></link>`

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

18 Changing the background color for the `body` element in `styles.css`

```
body {  
    background-color: brown;  
}
```

(19) Changing the body element's background color to be burlywood so we are still be able to read the text.

ex: `body { background-color: burlywood; }`

(20) Adding a div element inside the body element.

Inside the opening div tag, add the id attribute, with a value of menu.

ex: `<div id="menu">  
 <h1></h1>  
 <p></p>  
</div>`

(21) Make the div not take up the entire width of the page.

You can use the id selector to target a specific element with an id attribute. An id selector is defined by placing the hash symbol (#) directly in front of the element's id value. For example, if an element has the id of cat then you would target that element like this:

`#cat {  
 width: 250px;  
}`

`#menu {  
 width: 300px;  
}`

(22) Comments in CSS look like this:

`/* comment here */`

# changing the → `background-color: burlywood;`

into ↴

`/* background-color: burlywood */`

so you can see the effect of only styling the `#menu` element.

(23) Setting the background color of the div element to be burlywood.

`#menu {  
 width: 300px;  
 background-color: burlywood;  
}`

24) Changing the width property's value to be 80%.

```
#menu {  
    width: 300px;  
    background-color: burlywood;  
    width: 80%;  
}
```

25) Setting the #menu element. You can do this by setting its margin-left and margin-right properties to auto.

```
#menu {  
    margin-left: auto;  
    margin-right: auto;  
}
```

26) [THIS IS WEIRD]

changing #menu into .menu

→ It is a dot.

] READ THE INSTRUCTION AGAIN !

27) changing the `<div id="menu">` into `<div class="menu">`

↳ to apply the class's styling to the div element.  
↳ I don't get it

28) Deleting the comment and its content.

Adding a background-image property and set its value to `url (https://abc.org/image.jpg)`

↳ body {  
 background-image: url (https....jpg); → tadinya ini yg  
} burlywood.

29) Adding `<article></article>` under `<h2>coffee</h2>`

30) Article elements → commonly contain multiple elements that have related info. # Nest 2 p elements inside article element.

```
<article>  
    <p>French Vanilla</p>  
    <p>3.00 </p>  
</article>
```

] it's the menu of the cafe ^

- Start adding the menu / flavor of the coffee & each price.  
setiap menu harus ada article element - nya. contoh:

<article>  
<p> A </p> → 1<sup>st</sup> menu  
<p> 2.15 </p> → its price  
</article>

<article>  
<p> B </p> → 2<sup>nd</sup> menu  
<p> 1.75 </p> → its price  
</article>

article 1  
article 2

- (32) Adding the class name flavor to the French Vanilla p element  
<p class = "flavor"> French Vanilla </p> ⇒ INGETIN FORMAT  
PENULISAN KAYAK GINI !!!!. :-

- (33) Using your new flavor class as a selector, set the align text-align property's value to left.

• flavor {  
    text-align: left;  
}

- (34) Adding a class named price to p element that has 3.00 as its text. <p class = "price"> 3.00 </p>

- (35) Align the text to the right for the elements with the price class.  
• price {  
    text-align: right;  
}

- (36) [READ CAREFULLY !!!] It would be nice if the flavor and Price were on the same line. p elements are block-level elements, so they take up the entire width of their parent element. To get them on the same line, you need to apply some styling to the p element so they behave more like inline elements,

<h2> Coffee </h2>  
<article class = "Item">  
    <p class = "flavor"> French Vanilla </p>  
    <p class = "price"> 3.00 </p>  
</article>

③ The `p` elements are nested in an `article` element with the `class` attribute of `item`. You can style all the `p` elements nested anywhere in elements with a class named `item` like this:

```
• item p { }
```

# add a `display` property with value `inline-block` so the `p` elements behave more like inline elements.

```
• item p {  
    display: inline-block;  
}
```

④ Inline-block elements only take up the width of their content. To spread them out, add a width property to the flavor and price class selectors that have a value of 50%.

```
• flavor {  
    text-align: left;  
    width: 50%;  
}
```

```
• price {  
    text-align: right;  
    width: 50%;  
}
```

TERNYATA INI  
MENIPU :/  
HARUSNYA 49%  
EACH (dijelaskan  
di step 39)

⑤ 50% → 49%.

⑥ Moving the `p` element with the class `price` next to the `p` element with the class `flavor`.

```
<p class="flavor"> French Vanilla </p><p class="price"> 3.00 </p>
```

⑦ Changing the percentage of the width only.

⑧ Changing the entire article elements by adding `class item` to the article elements

```
<article class="item">  
    <p> Caramel </p>  
    <p> 1.60 </p>  
</article>
```

REPEAT WITH THE SAME FORMAT  
ORDERS.

⑨ Positioning the other `p` elements to be on the same line with no space between them. [ SEE THE 40TH STEP ].

④ Adding class names flavor & price to all the remaining p elements.

```
<article class = "item">  
  <p class = "flavor">Caramel</p><p class = "price">3.00</p>  
</article>
```

⑤ Changing the flavor class width value → 75%.  
 price class width value → 25%.

⑥ Adding a section element <section></section>

⑦ Adding <h2>Desserts</h2>

⑧ Adding <article class = "item"></article>

⑨ Adding <p>Donut</p><p>1.50</p>

⑩ Changing into <p class = "dessert">Donut</p><p class = "price">1.50</p>

⑪ Adding dessert class as an additional selector beside flavor (next to the flavor)

```
.flavor, .dessert {  
  text-align: left;  
  width: 75%;}  
}
```

⑫ Adding more desserts & prices using 3 more article elements (see step 50). → the format is same.

⑬ Giving the menu class a padding-left and a padding-right with the same value 20px.

```
.menu {  
  padding-left: 20px;  
  padding-right: 20px;  
}
```

to give some space between the content and the sides.

⑭ Adding padding-top: 20px  
 padding-bottom: 20px

- ⑤5 Kan value 4 padding (right, left, top, & bottom) sama, yaitu 20px. Jadi dimerge aja jadi:
- menu {  
padding: 20px;  
}
- ⑤6 The current width of the menu will take up 80% of the body's width (see step 24). On a very wide screen, the coffee and dessert appear far apart from their prices.  
# add a max-width property to the menu class with a value of 500px. max-width: 500px;
- ⑤7 Adding a font-family property with the value sans-serif.  
font-family: sans-serif;
- ⑤8 Styling both the h1 & h2 elements to use Impact font.  
h1, h2 {  
font-family: Impact;  
}  
→ kac dotnya ilang ya... T-T  
font-family: Impact, serif;
- ⑤9 Adding a fallback value for the font-family.
- ⑥0 Creating an established class selector and giving it the font-style property with the value italic.
- established {  
font-style: italic;  
}
- ⑥1 Applying the established class to the Est. 2020 text.  
<h1> CAMPER CAFE </h1>  
<p class="established"> Est. 2020 </p>
- ⑥2 Adding 2 new type selectors (h1 & h2). Use the font-size property, & the value = 40px for h1 & 30px for h2.
- h1 {  
font-size: 40px;  
}
  - h2 {  
font-size: 30px;  
}

⑥3 Adding a footer element below the main element.

<footer></footer>

⑥4 Inside the footer, add a p element. Then, nest an anchor a element in the p that links to ... and has the text Visit our website.

<footer>

<p><a href="www.abc.com">Visit our Web</a></p>

</footer>

→ to display a divider.

⑥5 Adding another p element

⑥6 Adding an hr element between the p element with the class established and the first section element. hr elements are self-closing.

<p class="established">Est. 2020 </p>

<hr>

<section></section>

⑥7 Setting the height property for hr element.

hr {

height: 3px;

⑥8 background-color: brown;

border-color: brown;

}

⑥9 The default value of a property named border-width is 1px for all edges of hr elements.  
# changing the height into 2px.

⑦0 Between the main element and the footer element, add hr element. → <hr> after closing the main & before opening the footer.

⑦1 Adding padding: 20px inside the body element

⑦2 Adding margin-top & margin-bottom: 5px inside .item p { }

- 74) Make the font size of the items and prices larger by using a value of 18px.
- item p {  
font-size: 18px;  
}
- 75) Adding some space by creating a class named bottom-line using 25px for the margin-top property.
- bottom-line {  
margin-top: 25px;  
}
- 76) Adding the bottom-line class to the second hr element so the styling is applied.
- <hr class = "bottom-line">
- 77) Adding a comment with the text FOOTER.
- /\*FOOTER\*/
- 78) (style.css) after adding a footer, make all the text have a value of 14px for the font size.
- footer {  
font-size: 14px;  
}
- 79) (style.css) use a type selector for the anchor element a and use the value black for the color property.
- a {  
color: black;  
}
- 80) (style.css) you can change properties of a link when the link has actually been visited by using a pseudo-selector that looks like a:visited {propertyName: propertyValue; } # change the color of the footer visit our website link to be grey.
- a:visited { color: grey; }

- 81 a : hover { color: brown; } → style.css  
(When the mouse hovers over them by using a pseudo-selector that looks like `a : hover { propertyValue; }`)
- # change the color of the footer `visit our website` link to be brown <sup>14</sup> when a user hovers over it.
- 82 a : active { color: white; } → kalau "visit our website" diklik, warnanya putih. → style.css
- 83 (style.css) changing the color for when the link is visited to black and use brown for when the link is actually clicked.
- 84 (style.css) changing the top margin of the h1 element to 0px to remove all the top margin.  
h1 {  
 font-size: 40px;  
 margin-top: 0px;  
}
- 85 (style.css) changing the bottom margin of h1 to 15px.
- 86 (style.css) to decrease the default margin space below the address p element, create a class selector named address and the bottom margin is set to 5px.  
.address {  
 margin-bottom: 5px;  
}
- 87 (index.html) Applying the address class to the p element containing the street address 123 Free Code Camp Drive.  
`<p class="address">123 Free Code Camp Drive</p>`
- 88 (index.html) Adding `` under the coffee heading.
- 89 (style.css) creating an img type selector, and use the value block for the display property and, set the margin-left and margin-right values to center it horizontally using auto value.

```
img {  
    display: block;  
    margin-left: auto;  
    margin-right: auto;  
}
```

90 Adding one last image under the **Desserts** heading using the url and giving the image an alt value of pie icon

```
<img src = "https://abc.org/pie.jpg" alt = "pie icon">
```

91 (style.css)

