



Website Layout

Silver - Chapter 3 - Topic 3

Selamat datang di **Chapter 3 Topik 3** online
course **Full Stack Web** dari Binar Academy!





Setelah kamu selesai dijelaskan tentang dasar-dasar membuat halaman web dan juga bahasa pemrograman Javascript, **Chapter 3 ini bakalan ngajak kamu buat belajar cara menjalankan kodingan dan cara berkolaborasi dengan developer lain dalam mengembangkan sebuah sistem.** Mulai dari penjelasan terkait terminal, IDE, GIT, membuat web layout dan responsive design.

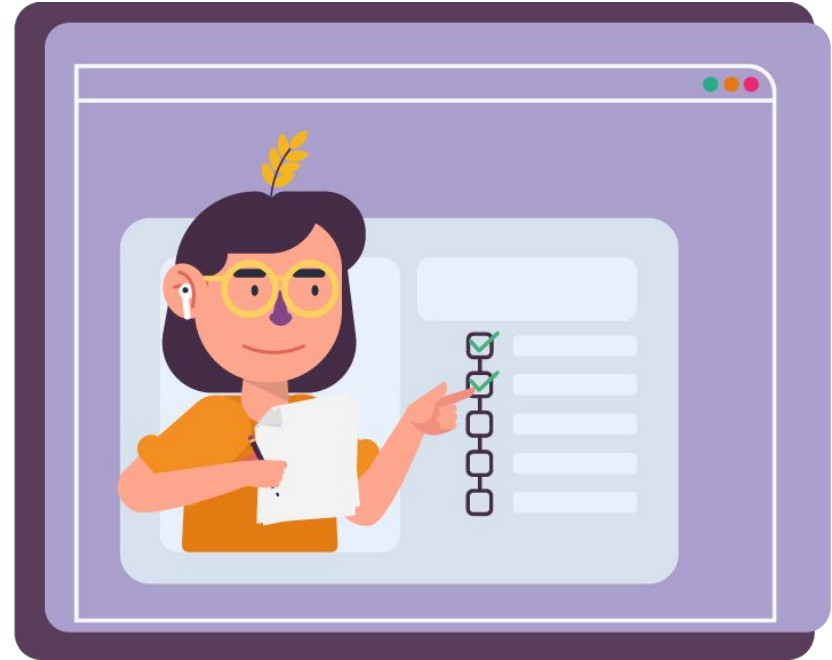
Pada topik pertama ini, kita bakal bahas tentang **web layouting**. Cus langsung aja~





Detailnya, kita bakal bahas dua hal berikut ini:

- Konsep Web Layout
- Teknik Layouting



Tampilan web yang ciamik nggak terlepas dari web layouting, lho!

Emang apa sih web layouting itu?



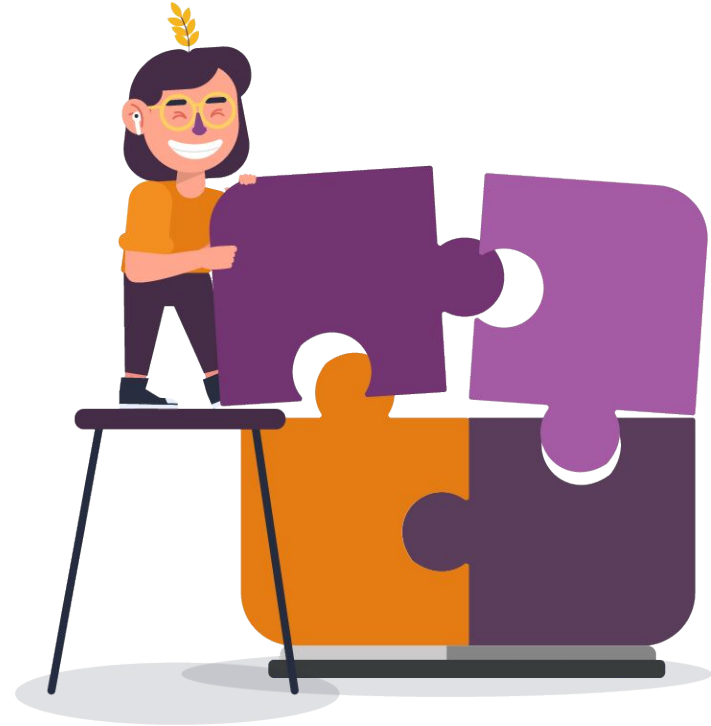


Mari kita mulai dengan pengertian layout.

Coba perhatikan gambar di samping! Apa yang terlintas di benak kamu? Yap~ Sabrina lagi nyusun beberapa puzzle jadi bentuk yang utuh.

Dalam proses penyusunan puzzle, Sabrina nggak bisa ngasal, lho! Ada tata letak yang harus diperhatikan agar puzzle-puzzle dapat membentuk tampilan yang utuh.

Tata letak kayak gitu disebut layout. Yaitu bagaimana suatu elemen diletakkan pada sebuah bidang atau media.

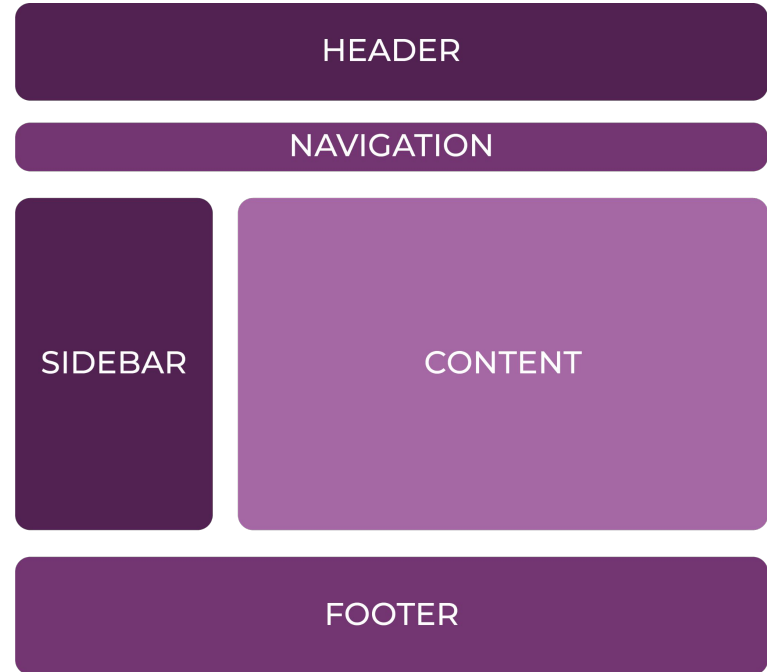




Nah, **web layout** itu adalah tata letak elemen-elemen **website** dalam suatu halaman web.

Contoh elemen-elemen web itu **kayak header, navigasi, content, article, sidebar, dan footer.**

Biar jadi web yang utuh dan enak dilihat, elemen-elemen harus disusun dan di-layout-in dengan baik.

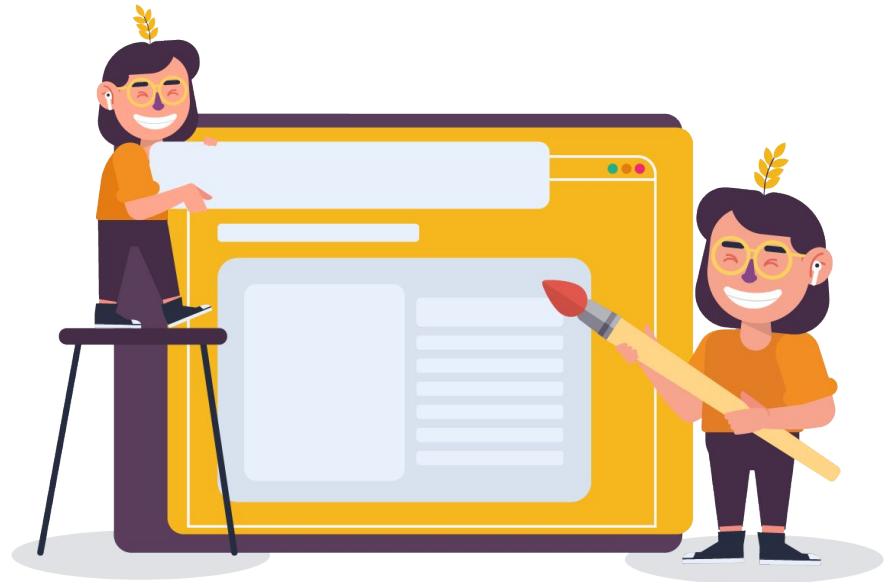




Nah, cara mengatur dan meletakkan elemen-elemen dalam web itu disebut **teknik layouting**.

Ada **tiga teknik layouting** nih, gengs!

- Old school layouting
- Flexbox
- Grid



Kita bedah satu-persatu, yuk!



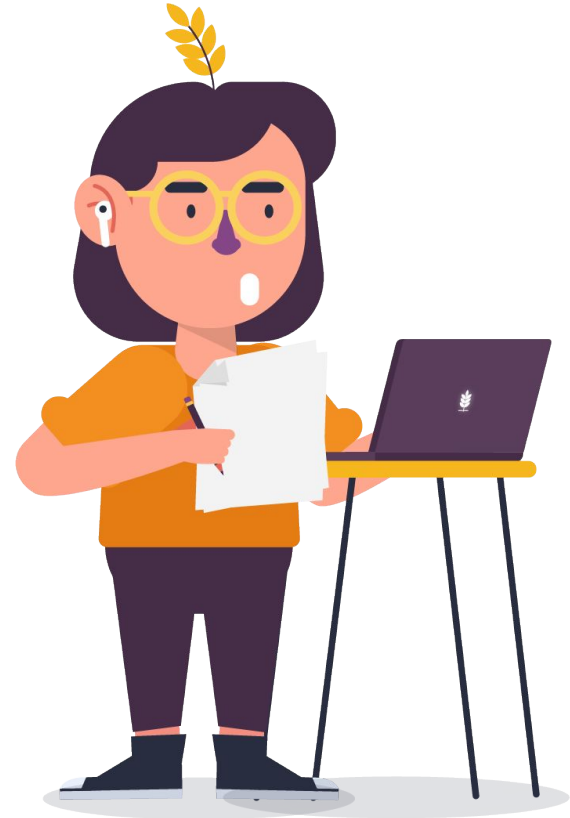


Old school layouting.

Ini bukan tema pensi yaaa xixixi~ 😎

Old school layouting adalah cara lama yang biasa digunakan untuk mengatur layout atau elemen sebuah website. Layout yang diatur menggunakan teknik ini antara lain: **display, dimension, box model, position**, dan **float**.

Biar lebih jelas, yuk kita bahas satu-persatu gimana pengaturan layout ini menggunakan teknik old school layouting~





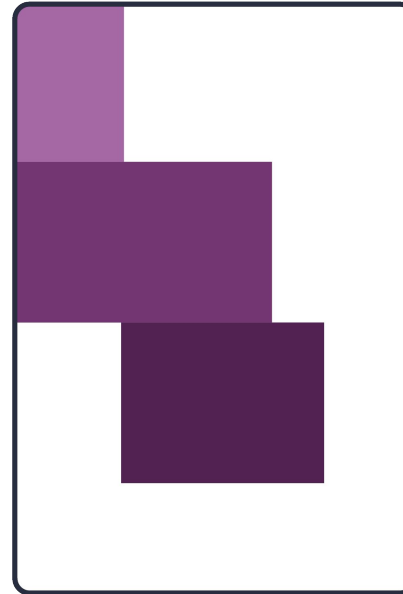
Display

Display adalah properti yang digunakan untuk mengatur perilaku tampilan dari sebuah elemen.

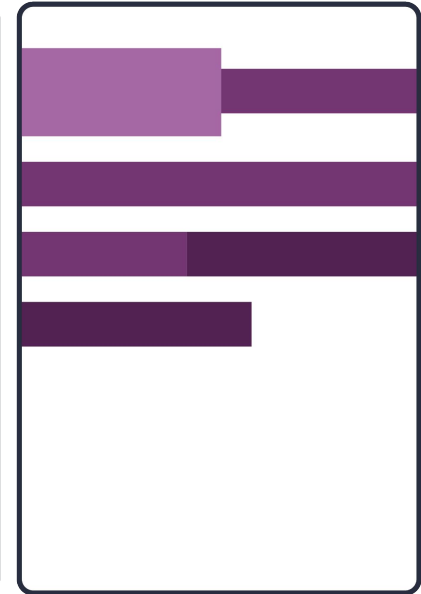
Kayak yang udah kita tahu, bahwa setiap elemen HTML memiliki display bawaan yaitu inline atau block. Contohnya kayak gambar disamping yaa!

Nah, kita bisa mengubah display bawaan ini menggunakan **properti display** yang ada di CSS.

Block



Inline



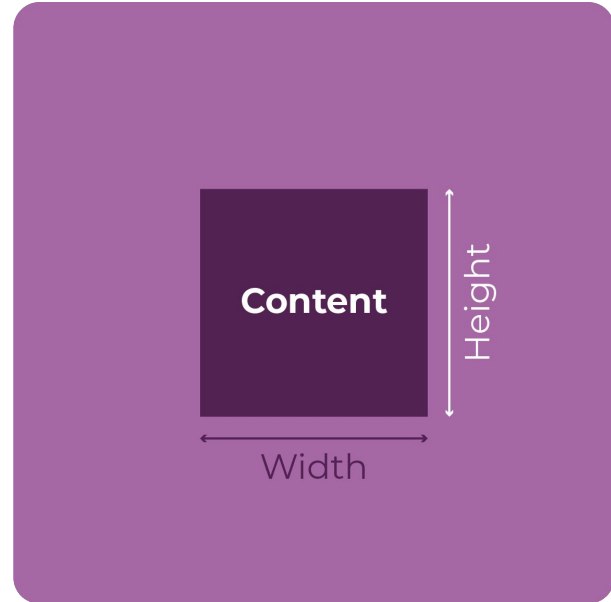


Dimension

Dimension adalah ukuran dari sebuah elemen.

Properti CSS yang digunakan untuk mengaturnya adalah **width** dan **height**. Width digunakan untuk mengatur lebar sebuah elemen. Sedangkan height digunakan untuk mengatur tinggi sebuah elemen.

Untuk valuenya, kita bisa menggunakan absolute unit seperti px atau relative unit seperti rem.



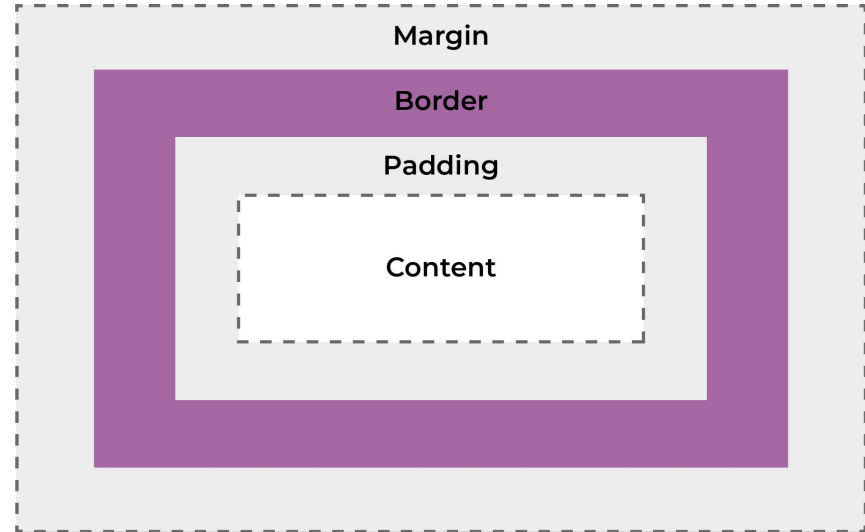


Box Model

Setiap elemen HTML akan di model kan seperti sebuah box.

Box model ini terdiri dari:

- **Content:** berupa teks atau gambar yang memiliki dimensi tinggi dan lebar.
- **Padding:** jarak dari konten ke border, agar halaman lebih mudah dilihat.
- **Border:** pembatas yang mengelilingi padding dan konten.
- **Margin:** area kosong untuk memberi jarak antar elemen.





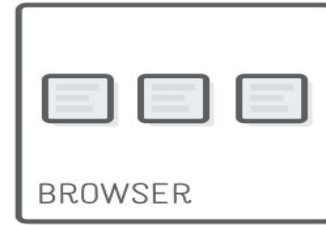
Position

Position adalah properti yang berfungsi mengatur bagaimana elemen diposisikan dalam dokumen.

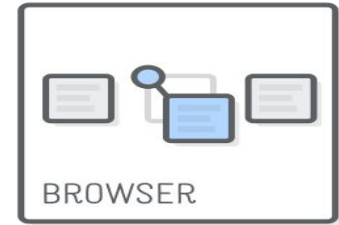
Ada **lima value** untuk properti position, yaitu:

- Static
- Relative
- Absolute
- Fixed
- Sticky

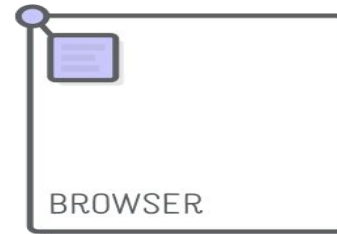
STATIC



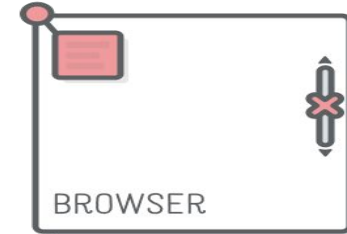
RELATIVE



ABSOLUTE



FIXED



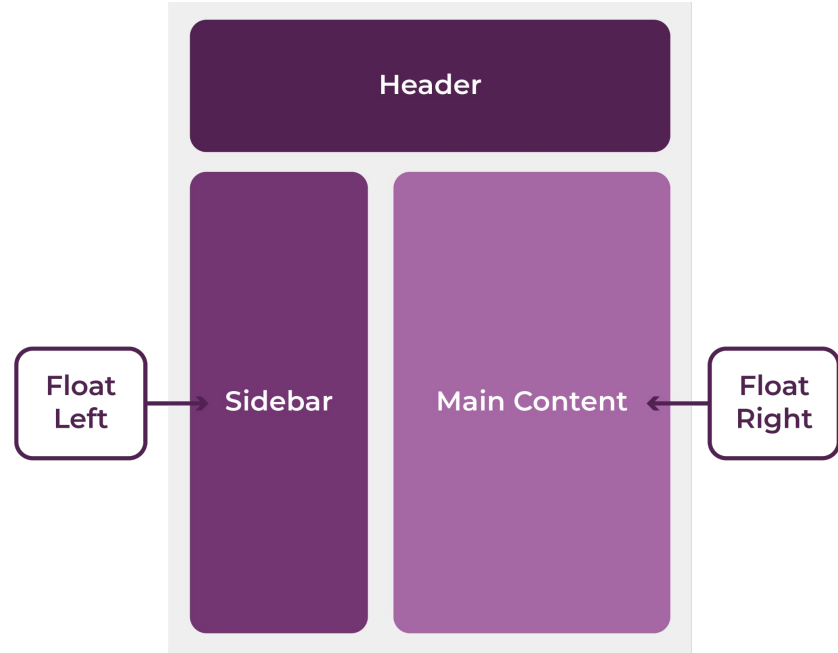


Float

Float adalah properti yang digunakan untuk mengatur letak sebuah elemen.

Ada **empat value** untuk properti float, yaitu:

- Right
- Left
- None
- Inherit





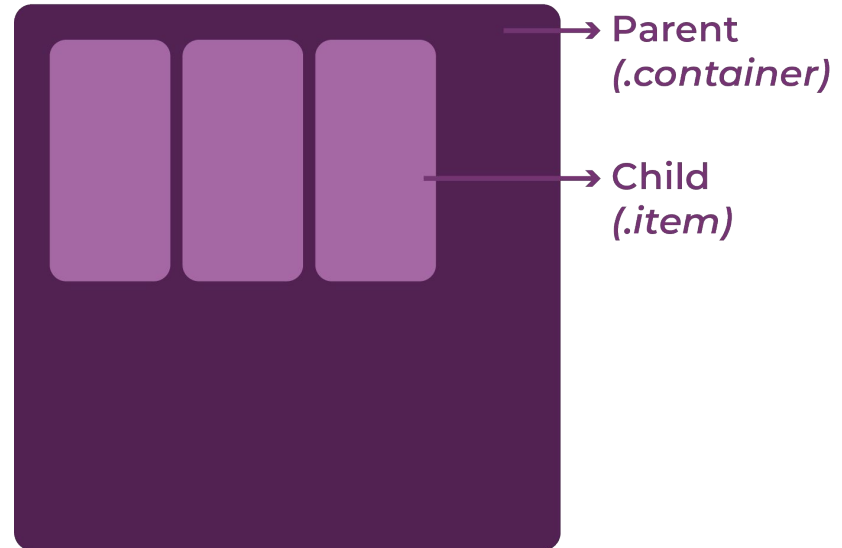
Flexbox.

Flexbox adalah singkatan dari flexible box layout. Merupakan **salah satu modul CSS3, yang berfungsi untuk mengatur layout website dalam satu dimensi atau satu arah.**

Ada dua elemen penting yang perlu diperhatikan saat menggunakan flexbox, yaitu **parent/container** dan **child/item**

Nah, cara menggunakan flexbox adalah dengan mengubah display dari parent menjadi flex.

CSS Flexbox





Berikut daftar properti CSS yang digunakan untuk mengatur parent element di flexbox.

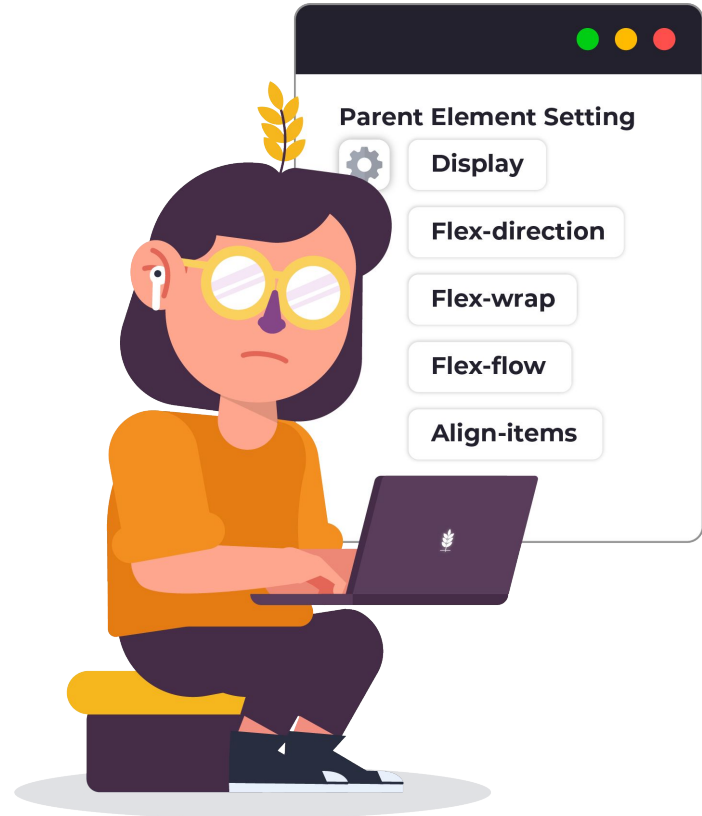
Display: untuk mengubah display menjadi flexbox.

Flex-direction: untuk mengatur arah layout (baris/kolom).

Flex-wrap: untuk mengatur ukuran container.

Flex-flow: shorthand property untuk menggabungkan direction dan wrap.

Align-items: untuk mengatur alignment child secara vertikal.



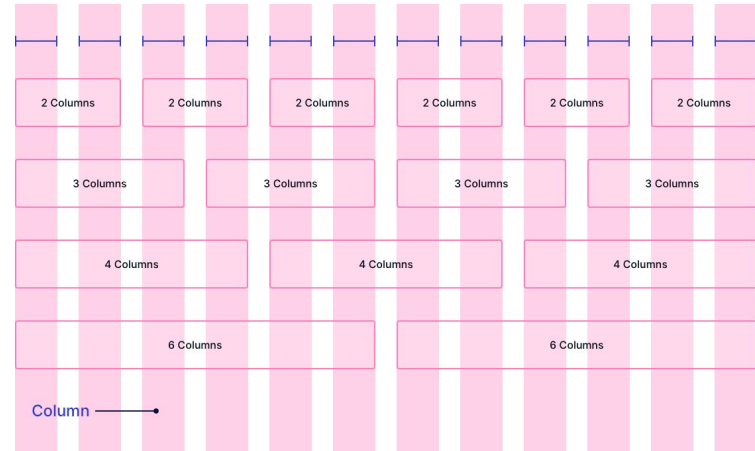


Grid.

Sama seperti flexbox, Grid juga merupakan **salah satu modul CSS3 yang digunakan untuk mengatur layout website.**

Perbedaan antara keduanya adalah, **flexbox hanya dapat mengatur layout dari satu arah**, sedangkan **grid dapat mengatur layout dari dua arah**, baik itu baris maupun kolomnya.

Standar untuk sebuah grid system adalah 12 kolom.



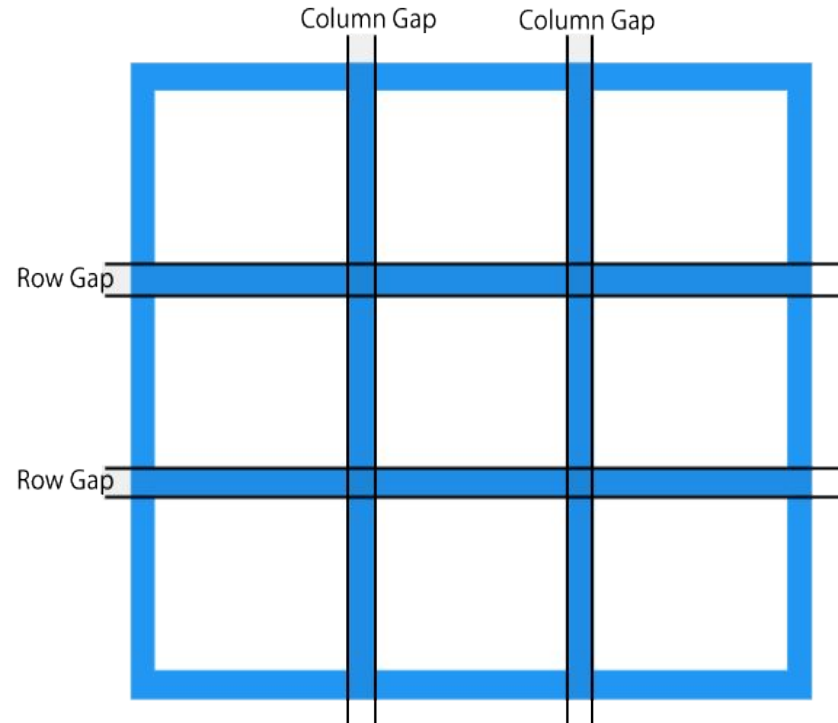


Sama seperti flexbox, grid juga memiliki dua elemen penting, yaitu **parent.container** dan **child/item**

Nah, cara menggunakannya adalah dengan mengubah display dari parent element menjadi grid. Kemudian, tentukan berapa kolom dan baris yang kamu butuhkan.

Penting untuk kamu tahu, nih!

Area dari satu baris dan satu kolom disebut **cell**. Gabungan dari lebih dari beberapa cell disebut **grid area**.





Berikut daftar properti CSS yang digunakan untuk mengatur parent element di grid.

Display: untuk mengubah display menjadi grid.

Grid-template-columns: untuk mengatur jumlah kolom.

Grid-template-rows: untuk mengatur jumlah baris.

Grid-template-areas: untuk menamai sebuah area beberapa item.

Grid-template: shorthand untuk grid-template-columns, rows, dan areas.

Column-gap: untuk mengatur jarak antar kolom.

Row-gap: untuk mengatur jarak antar baris.

Justify-items: untuk mengatur alignment item secara horizontal.

Align-items: untuk mengatur alignment item secara vertikal.

Justify-content: untuk mengatur alignment container secara horizontal.

Align-content: untuk mengatur alignment container secara vertikal.





Berikut daftar properti CSS yang digunakan untuk mengatur **child element** di grid.

Grid-column-start: untuk menentukan posisi awal sebuah item berdasarkan baris kolom.

Grid-column-end: untuk menentukan posisi akhir sebuah item berdasarkan baris kolom.

Grid-row-start: untuk menentukan posisi awal sebuah item berdasarkan baris.

Grid-row-end: untuk menentukan posisi akhir sebuah item berdasarkan baris.



Saatnya kita
Quiz!





1. Yang bukan merupakan cara baru untuk mengatur layout sebuah website adalah...

- A. Display
- B. Flexbox
- C. Grid



1. Yang bukan merupakan cara baru untuk mengatur layout sebuah website adalah...

- A. Display
- B. Flexbox
- C. Grid

Display merupakan salah satu teknik old school layouting yang biasa di gunakan untuk mengatur layout sebuah website.



2. Yang tidak termasuk dalam value sebuah position adalah...

- A. Static
- B. Container
- C. Absolute
- D. Fixed



2. Yang tidak termasuk dalam value sebuah position adalah...

- A. Static
- B. Container
- C. Absolute
- D. Fixed

Value dari position ada 5, yaitu Static, Relative, Absolute, Fixed, Sticky.



3. Flex-wrap merupakan bagian dari teknik layouting flexbox yang memiliki fungsi untuk...

- A. Mengubah display menjadi flexbox
- B. Menggabungkan direction dan wrap
- C. Mengatur ukuran container
- D. Mengatur alignment pada child secara vertikal



3. Flex-wrap merupakan bagian dari teknik layouting flexbox yang memiliki fungsi untuk...

- A. Mengubah display menjadi flexbox
- B. Menggabungkan direction dan wrap
- C. Mengatur ukuran container
- D. Mengatur alignment pada child secara vertikal

Display : untuk mengubah display menjadi flexbox

Flex-wrap : untuk mengatur ukuran container

Flex -flow : Ini adalah shorthand property untuk menggabungkan direction dan wrap

Flex-direction : untuk mengatur alignment child secara horizontal



4. Flex-flow merupakan bagian dari teknik layouting flexbox yang memiliki fungsi untuk...

- A. Mengubah display menjadi flexbox
- B. Menggabungkan direction dan wrap
- C. Mengatur ukuran container
- D. Mengatur alignment pada child secara vertikal



4. Flex-flow merupakan bagian dari teknik layouting flexbox yang memiliki fungsi untuk...

- A. Mengubah display menjadi flexbox
- B. Menggabungkan direction dan wrap
- C. Mengatur ukuran container
- D. Mengatur alignment pada child secara vertikal

Flex-direction : untuk mengatur arah layout(baris/kolom)

Flex-wrap : untuk mengatur ukuran container

Flex -flow : Ini adalah shorthand property untuk menggabungkan direction dan wrap

Flex-direction : untuk mengatur alignment child secara horizontal



5. Manakah pernyataan dibawah ini yang benar?

- A. Flexbox hanya bisa mengatur layout dari satu arah, sedangkan grid bisa mengatur layout dari dua arah.
- B. Flexbox sudah ditentukan dengan pasti jumlah child/ itemnya, sedangkan grid bisa mengatur sendiri.
- C. Flexbox dan grid bisa mengatur layout dari dua arah.
- D. Flexbox bisa mengubah jumlah baris dan kolomnya, sedangkan grid tidak bisa.



5. Manakah pernyataan dibawah ini yang benar?

- A. Flexbox hanya bisa mengatur layout dari satu arah, sedangkan grid bisa mengatur layout dari dua arah.
- B. Flexbox sudah ditentukan dengan pasti jumlah child/ itemnya, sedangkan grid bisa mengatur sendiri.
- C. Flexbox dan grid bisa mengatur layout dari dua arah.
- D. Flexbox bisa mengubah jumlah baris dan kolomnya, sedangkan grid tidak bisa.

Perbedaan grid dan flexbox adalah, flexbox hanya bisa mengatur layout dari satu arah, sedangkan grid bisa mengatur layout dari dua arah, baik itu baris maupun kolomnya.



Referensi dan bacaan lebih lanjut~

1. https://www.w3schools.com/css/css_website_layout.asp
2. <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
3. <https://css-tricks.com/snippets/css/complete-guide-grid/>





Nah, selesai sudah pembahasan kita di Chapter 3 Topic 3 ini.

Selanjutnya, kita bakal bahas tentang **responsive design**.

Penasaran kayak gimana? Cus langsung ke topik selanjutnya~



Terima Kasih!



Next Topic

loading...