

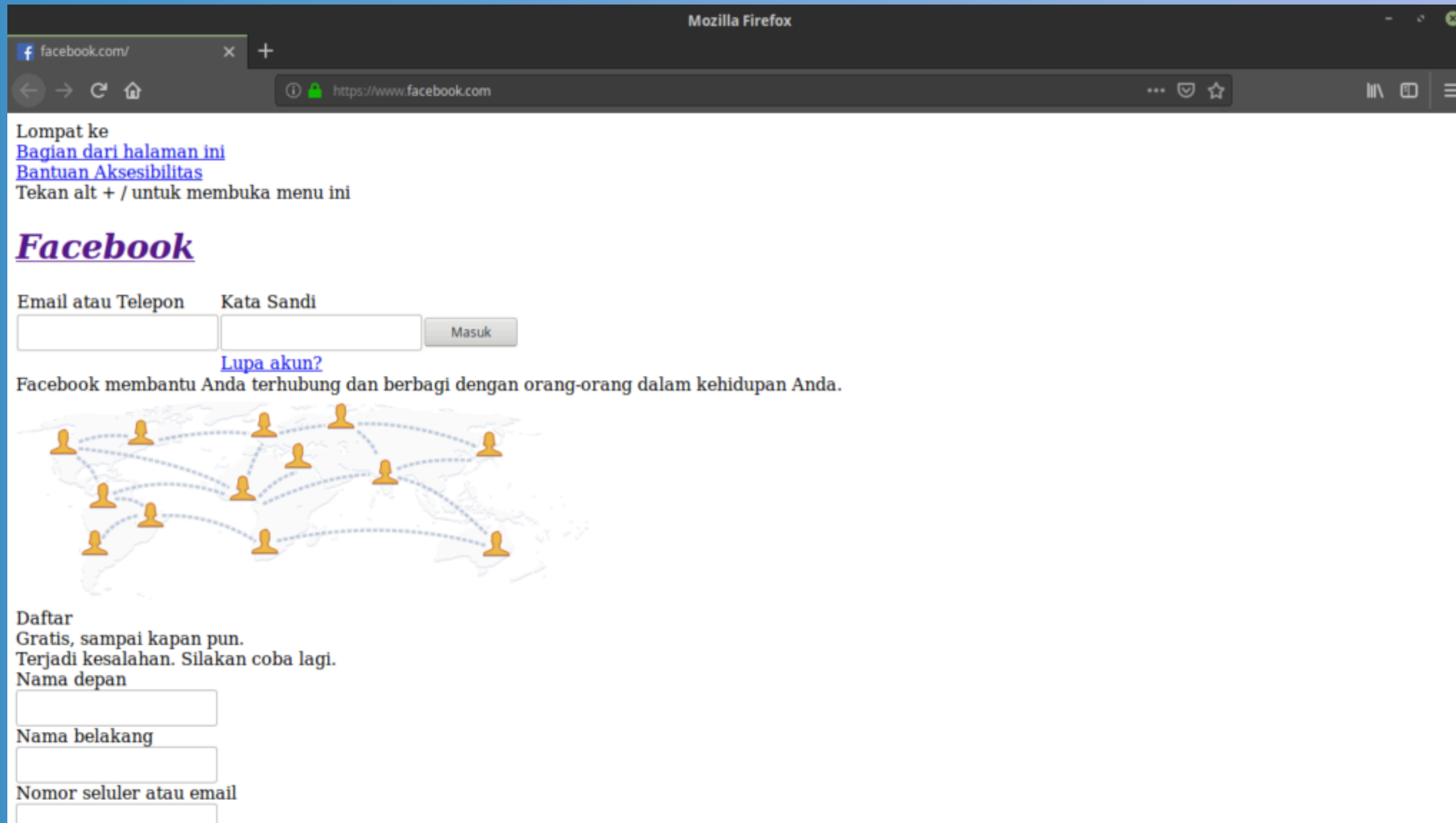
CSS dan Framework CSS

Definisi CSS

CSS merupakan singkatan dari Cascading Style Sheets, CSS digunakan untuk menentukan gaya dari tampilan website yang akan dibuat seperti tata letak halaman, warna , dan font. Semuanya dapat ditentukan oleh CSS, CSS bertujuan agar website terlihat lebih menarik.

Pada dasarnya tanpa menggunakan CSS, tampilan website dapat terlihat menarik hanya dengan bahasa HTML namun penggunaan HTML terbukti tidak efektif jika digunakan untuk design tampilan oleh karena itu CSS dan HTML tidak dapat dipisahkan satu sama lain.

Contoh Web tanpa CSS



Dengan CSS

Facebook - Masuk atau Daftar - Mozilla Firefox

Facebook - Masuk atau Daftar

https://www.facebook.com

facebook

Email atau Telepon

Kata Sandi

Masuk

Lupa akun?

Facebook membantu Anda terhubung dan berbagi dengan orang-orang dalam kehidupan Anda.

Daftar

Gratis, sampai kapan pun.

Nama depan

Nama belakang

Nomor seluler atau email

Kata sandi baru

Tanggal Lahir

13 Jul 1994

Jenis Kelamin

☐ Perempuan ☐ Laki-laki ☐ Khusus

Dengan mengklik Daftar, Anda menyetujui [Ketentuan, Kebijakan Data](#) dan [Kebijakan Cookie](#) kami. Anda akan menerima Notifikasi SMS dari Facebook dan dapat menolaknya kapan saja.

Daftar

Buat Halaman untuk selebriti, grup musik, atau bisnis.

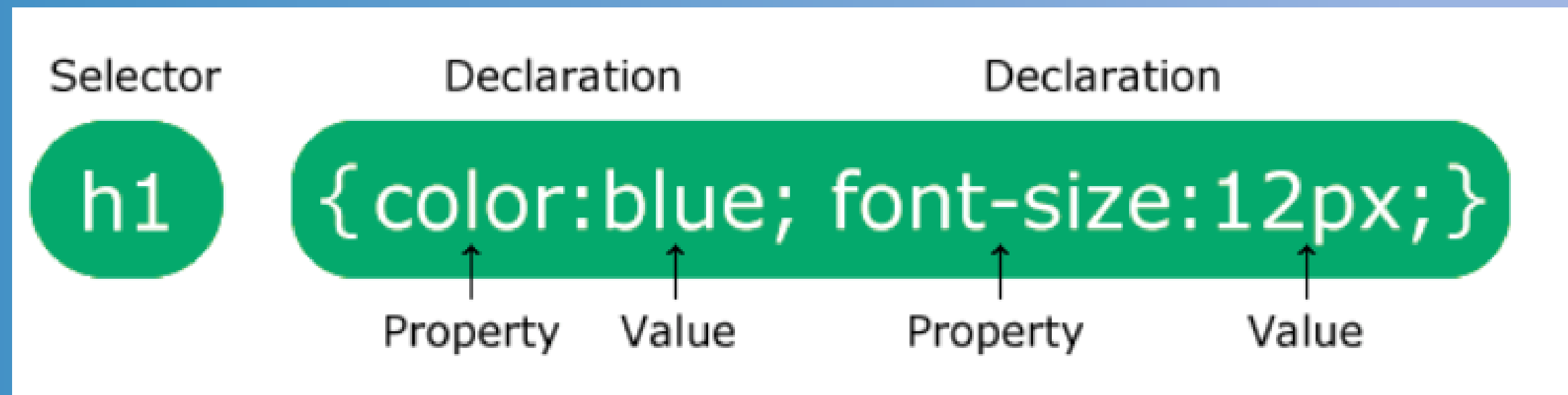
Transferring data from www.gstatic.com...

Integrasi CSS

Ada 3 cara mengintegrasikan CSS ke HTML :

- Internal CSS
- Inline CSS
- External CSS

Basic CSS Syntax



CSS 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-style

mengubah font menjadi bercetak miring

line-height

mengatur spasi antar baris

CSS Text Styling

color

memberi warna pada tulisan

text-align

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

letter-spacing

mengatur spasi antar huruf

word-spacing

mengatur spasi antar kata

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

Pseudo Class

: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 Layouting

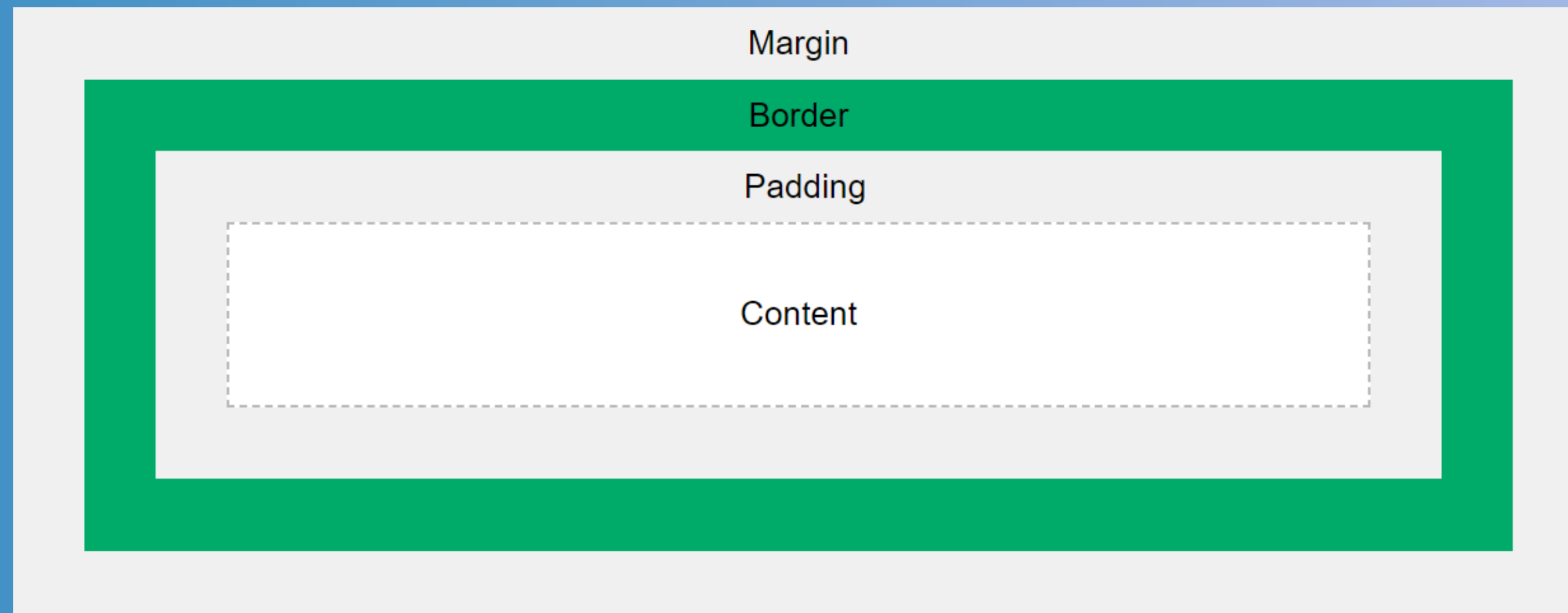
CSS Layouting adalah teknik untuk mengatur tata letak elemen pada halaman web menggunakan CSS.

Ada beberapa hal yang perlu diketahui untuk mengatur layout dengan CSS :

- Konsep Box Model
- Display
- Positioning
- Float
- Flexbox dan Grid

Box Model

Box Model adalah konsep fundamental dalam CSS yang menentukan bagaimana elemen HTML dirender di dalam halaman web. Setiap elemen dianggap sebagai sebuah "kotak" yang terdiri dari beberapa bagian



Display

display digunakan untuk menentukan bagaimana suatu elemen ditampilkan pada halaman web.

Setiap elemen HTML memiliki nilai display default, tergantung pada jenis elemennya. Nilai display default untuk sebagian besar elemen adalah block atau inline.

Examples of inline elements:

- ``
- `<a>`
- ``

Examples of block-level elements:

- `<div>`
- `<h1>` - `<h6>`
- `<p>`
- `<form>`
- `<header>`
- `<footer>`
- `<section>`

Display

Terdapat 4 property display yang umum digunakan :

- Block : Setiap elemennya selalu dimulai pada baris baru dan menempati lebar containernya.
- Inline : Tidak dimulai pada baris baru, dan hanya selebar kontennya.
- Inline-Block : Perilakunya sama seperti inline, namun *height* dan *width* bisa diatur
- None : Menghilangkan elemen

Positioning

Positioning attribute menentukan jenis metode pemosisian yang digunakan untuk suatu elemen.

Elemen kemudian bisa dipindahkan menggunakan properti **top**, **bottom**, **left**, dan **right**. Namun, properti ini tidak akan berfungsi kecuali properti **position** ditetapkan terlebih dahulu. Properti ini juga berfungsi secara berbeda tergantung pada value position.

Positioning

Terdapat 5 position values dalam css :

- **Static** : Default element position, tidak bisa digerakkan, dan masuk ke normal flow
- **Relative** : Dapat digerakkan relatif dari posisi normal flownya, keluar dari normal flow (out of flow)
- **Fixed** : Diposisikan berdasarkan normal flow, dapat digerakkan
- **Absolute** : Dapat digerakkan relatif dari containernya
- **Sticky** : Sama seperti **relative**, namun ketika di titik tertentu dalam scroll, ia akan menempel seperti **fixed**

CSS Responsive

Halaman web dapat dilihat menggunakan berbagai perangkat: desktop, tablet, dan ponsel. Halaman web Anda harus terlihat bagus dan mudah digunakan, apa pun perangkatnya.

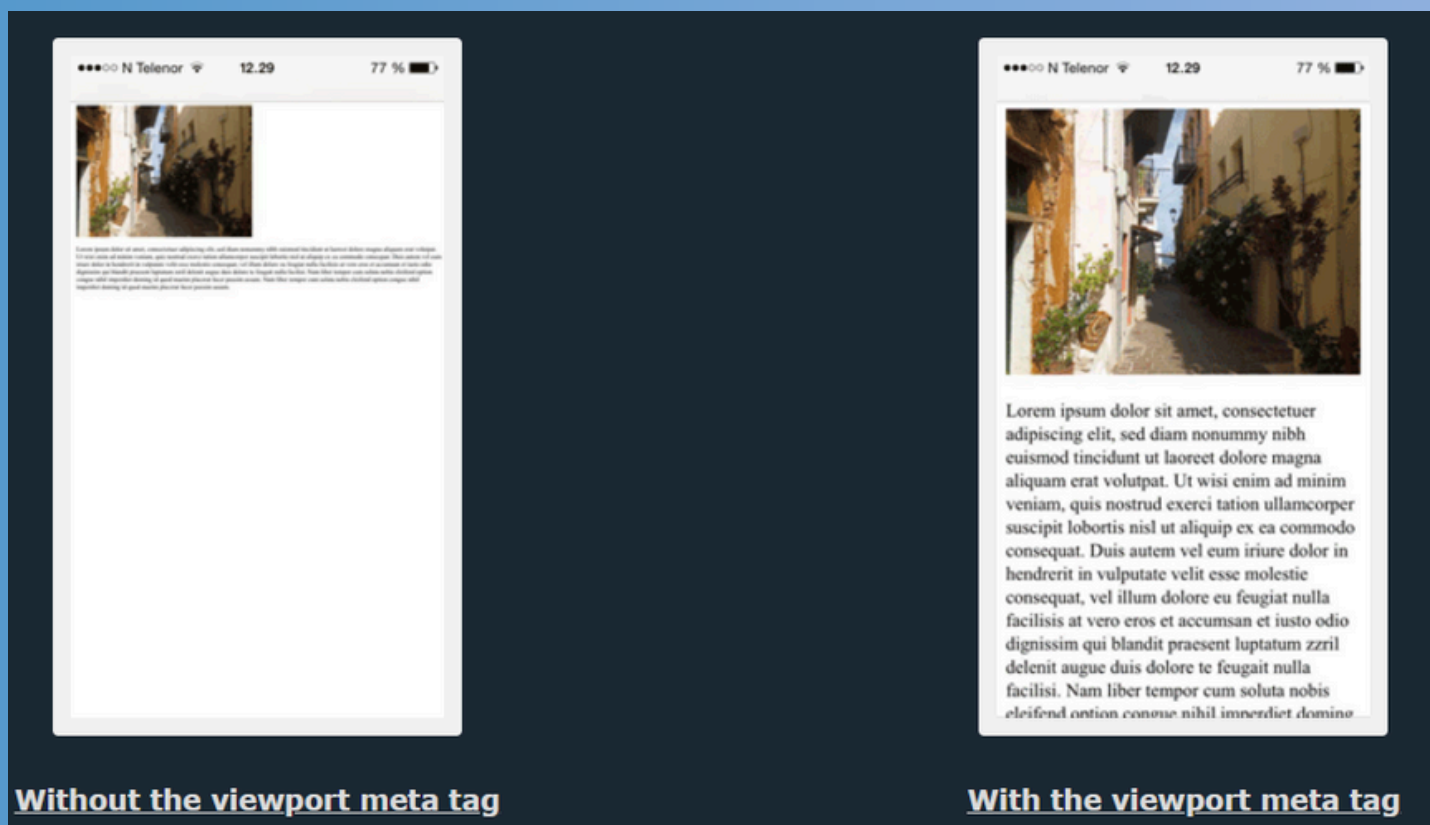
Halaman web tidak boleh menghilangkan informasi agar sesuai dengan perangkat yang lebih kecil, tetapi harus menyesuaikan kontennya agar sesuai dengan perangkat apa pun



Viewport

Viewport adalah area yang terlihat oleh pengguna di halaman web. Viewport tiap device berbeda tergantung ukurannya.

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



Media-Query

Media queries memungkinkan kita menerapkan CSS berdasarkan lebar layar atau karakteristik perangkat lainnya.

Mengatur media query dapat dengan @media.

Contoh :

```
@media (max-width: 768px) {  
  body {  
    background-color: lightgray;  
  }  
}
```

Arti : Ketika layar kurang dari 768px maka background color abu-abu

Float

adalah properti CSS yang digunakan untuk mengatur bagaimana elemen mengalir dalam sebuah halaman web, terutama dalam mengatur tata letak teks dan gambar. Properti ini memungkinkan elemen untuk "mengapung" di sebelah elemen lain, sehingga konten lain dapat mengalir di sekitarnya. Nilai float yang dapat digunakan adalah : **left, right, none, inherit**

FlexBox

Flexbox adalah model tata letak CSS yang dirancang untuk mengatur dan menyusun elemen dalam suatu kontainer secara fleksibel. Dengan Flexbox, elemen dapat dengan mudah disusun dalam satu dimensi (baris atau kolom), serta dapat menyesuaikan ukuran dan posisinya sesuai dengan ruang yang tersedia.

Flexbox memudahkan perancangan struktur tata letak responsif yang fleksibel, tanpa menggunakan float atau positioning.

FlexBox

Cara menggunakan flexbox adalah dengan menjadikan property display container yang ingin kita gunakan sebagai flex.

Property pada container yang dapat digunakan adalah :

- flex-direction : Mengatur arah item pada container
- flex-wrap : Mengatur responsive item pada container
- flex-flow : Sorthand flex-direction dan flex-wrap
- justify-content : mengatur posisi item secara horizontal
- align-items : mengatur posisi item secara vertikal
- align-content : Mengatur posisi item secara vertikal berdasarkan ukuran layar

Grid

CSS Grid adalah sistem tata letak dua dimensi yang memungkinkan kita mengatur elemen dalam baris (rows) dan kolom (columns) dengan lebih fleksibel dibanding Flexbox.

Cara menggunakan CSS grid sama seperti flexbox, dengan cara memberi value pada properti display di container.

Grid

CSS Grid Properties

- `grid-template-columns`
 - `grid-template-rows`
 - `grid-auto-columns`
 - `grid-auto-rows`
 - `grid-auto-flow`
 - `grid-template-areas`
 - `grid-template`
 - `grid-column-gap`
 - `grid-row-gap`
 - `grid-gap`
 - `grid`
- `justify-items`
 - `align-items`
 - `place-items`
 - `justify-content`
 - `align-content`
 - `place-content`

CONTAINER

- `grid-column-start`
- `grid-column-end`
- `grid-row-start`
- `grid-row-end`
- `grid-column`
- `grid-row`
- `grid-area`
- `justify-self`
- `align-self`
- `place-self`

ITEM

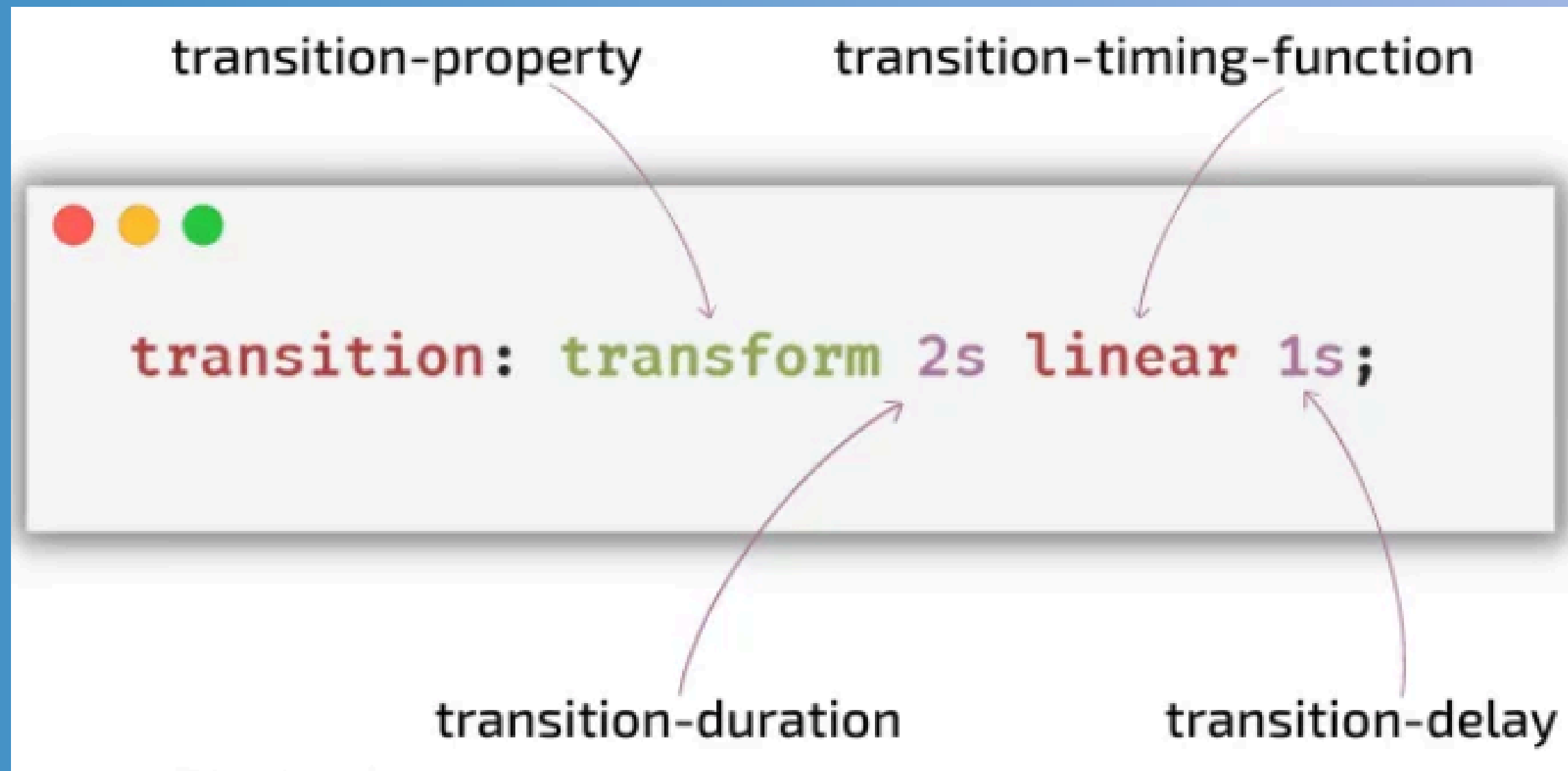
Grid

Special Function & Keywords

- `repeat()`
- `min-content` & `max-content`
- `minmax()`
- `auto-fill` & `auto-fit`

Transition

CSS transition digunakan untuk membuat animasi halus ketika properti CSS berubah, misalnya saat hover atau perubahan state lainnya.



Framework CSS

Framework CSS adalah kumpulan kode CSS yang sudah siap pakai untuk mempercepat dan mempermudah proses pengembangan tampilan website atau aplikasi web. Framework ini menyediakan berbagai komponen, seperti grid system, tombol, formulir, navigasi, dan lainnya, sehingga pengembang tidak perlu menulis CSS dari nol.

Framework yang paling terkenal ada 2, yaitu : bootstrap dan tailwind

Bootstrap

Bootstrap adalah framework CSS berbasis komponen (componen based) yang menyediakan banyak elemen UI siap pakai, seperti tombol, navbar, modal, dan sistem grid responsif. Bootstrap juga dilengkapi dengan JavaScript untuk komponen interaktif.

<https://getbootstrap.com/docs/5.3/getting-started/introduction/>

Kelebihan Bootstrap :

- Cepat dan mudah digunakan.
- Komponen UI sudah siap pakai.
- Dokumentasi lengkap dan komunitas besar.

Kekurangan Bootstrap:

- Kurang fleksibel dalam kustomisasi.
- Bisa menghasilkan tampilan yang seragam

Tailwind CSS

Tailwind CSS adalah framework CSS berbasis utility-first, yang berarti pengembang menggunakan kelas-kelas kecil untuk membangun desain langsung di HTML tanpa memerlukan banyak file CSS tambahan.

<https://tailwindcss.com/docs/installation/tailwind-cli>

Kelebihan Tailwind CSS :

- Sangat fleksibel dan mudah dikustomisasi.
- Tidak menghasilkan CSS berlebihan karena hanya memuat kelas yang digunakan.
- Desain tidak terbatas pada tampilan default seperti Bootstrap.

Kekurangan Tailwind CSS:

- Sedikit lebih sulit dipelajari karena harus memahami kelas utility.
- HTML bisa terlihat penuh dengan banyak kelas.

Terima Kasih