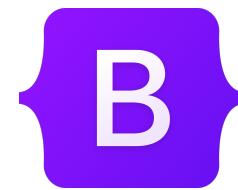




Pemrograman Sisi Cleint Dasar CSS

Muhammad Syaifur Rohman, S.Kom, M.CS



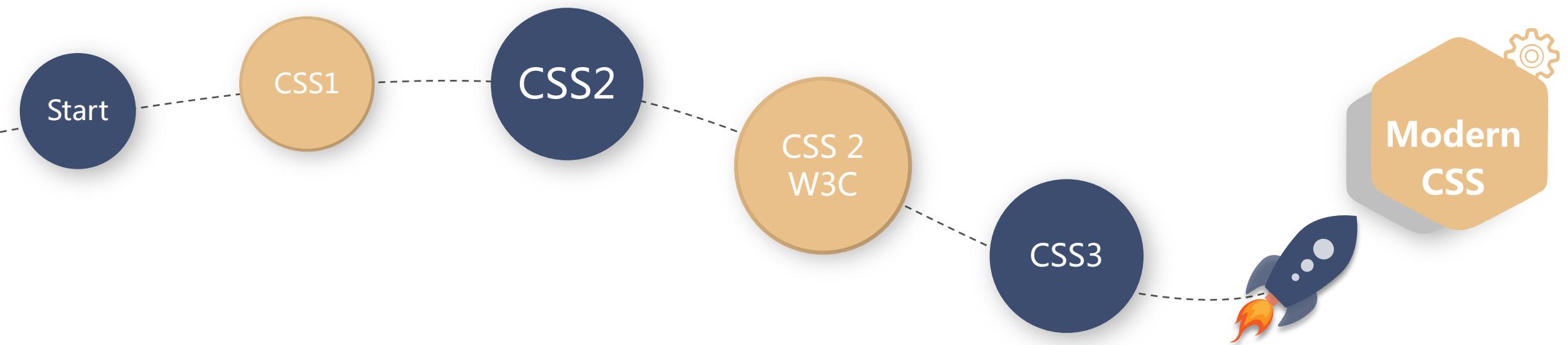
Sejarah HTML



Sejarah

CSS, yang merupakan singkatan dari Cascading Style Sheets, adalah teknologi yang digunakan untuk mengatur tampilan dan format dokumen yang ditulis dalam bahasa markup seperti HTML. CSS memungkinkan pengembang web dan desainer untuk memisahkan konten dari desain, sehingga memudahkan untuk mengubah tampilan situs web tanpa harus mengubah konten dasarnya.

Timeline CSS

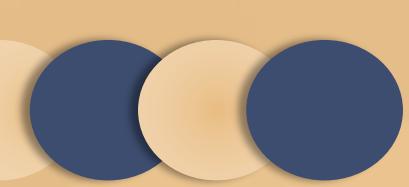


Framework CSS



tailwindcss

Framework CSS adalah kumpulan file CSS yang telah dirancang sebelumnya, yang bertujuan untuk membantu pengembang web dan desainer dalam membangun antarmuka web yang cepat dan konsisten. Framework ini biasanya menyertakan sistem grid untuk layout halaman, komponen antarmuka pengguna (UI) yang dapat digunakan kembali, serta utility classes untuk memanipulasi tampilan dan jarak antar elemen.



Bootstrap Framework

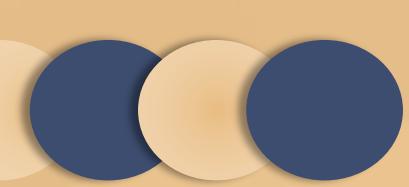
Bootstrap adalah salah satu framework HTML, CSS, dan JS yang paling populer untuk pengembangan web responsif dan mobile-first. Dirilis pertama kali pada tahun 2011 oleh Twitter, Bootstrap telah berkembang secara signifikan sejak itu. Versi terkini pada saat penulisan ini adalah Bootstrap 5, yang menghadirkan perbaikan pada grid system, komponen, dan utilities, serta menghilangkan jQuery sebagai ketergantungan.

Kelebihan

- **Kemudahan Penggunaan:** Dengan kelas siap pakai, Bootstrap memudahkan pembuatan layout dan UI.
- **Responsif:** Sistem grid memudahkan pembuatan desain yang responsif.
- **Dokumentasi yang Luas:** Memiliki dokumentasi yang lengkap, membuatnya mudah dipelajari.
- **Komunitas yang Besar:** Mudah mendapatkan dukungan dan sumber belajar.

Kekurangan

- **Ukuran File:** Dapat menambah ukuran file akhir proyek.
- **Kustomisasi:** Meskipun dapat dikustomisasi, prosesnya bisa menjadi rumit untuk desain yang sangat spesifik.
- **Tampilan Seragam:** Aplikasi yang menggunakan Bootstrap dapat memiliki tampilan yang serupa.



Bootstrap – Sistem Grid

Cara Kerja: container, row, dan column

Bootstrap menggunakan sistem grid 12 kolom yang responsif, yang memungkinkan pengembang untuk dengan mudah mengatur konten dalam layout yang kompleks.

Container: Wrapper untuk konten grid. Bisa container untuk fixed-width atau container-fluid untuk full width.

Row: Mengelompokkan kolom secara horizontal.

Column: Konten sebenarnya yang akan diatur. Menggunakan kelas seperti col-md-4 (mengambil 4 dari 12 kolom di medium devices).

Responsive Layout dengan Class Breakpoint

Bootstrap memiliki breakpoint responsif yang memungkinkan kolom untuk tumpang tindih atau merespons berdasarkan ukuran layar. Misalnya, col-sm-, col-md-, col-lg-, col-xl-, berarti kolom akan beradaptasi mulai dari small, medium, large, dan extra large devices.

Struktur Dasar dan Komponen Dasar Bootstrap

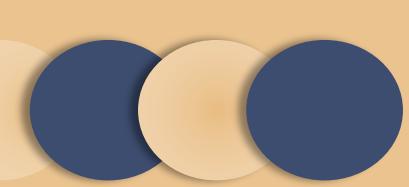


```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bootstrap Example</title>
  <link href="https://stackpath.bootstrapcdn.com/bootstrap/5.1.0/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>

<div class="container mt-5">
  <!-- Sistem Grid -->
  <div class="row">
    <div class="col-md-8">
      <div class="card">
        <div class="card-body">
          Ini adalah Card.
        </div>
      </div>
    </div>
    <div class="col-md-4">
      <button type="button" class="btn btn-primary">Tombol Bootstrap</button>
    </div>
  </div>
</div>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/5.1.0/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

- Menggunakan sistem grid dengan row dan col-md-8 serta col-md-4 untuk membagi layout menjadi dua bagian, di mana satu bagian lebih besar.
- Mencakup card sebagai contoh komponen UI.
- Menambahkan button dengan kelas btn dan btn-primary untuk menunjukkan penggunaan komponen tombol.



Tailwind Framework

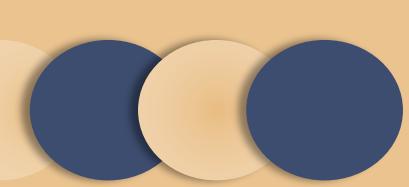
Tailwind CSS mengadopsi pendekatan 'utility-first' untuk styling, yang berarti menggunakan kelas utilitas tunggal yang melakukan satu pekerjaan secara spesifik. Hal ini bertujuan untuk memudahkan pembuatan desain yang responsif dan custom dengan cepat tanpa perlu keluar dari HTML.

Kelebihan

- **Efisiensi:** Membuat prototipe cepat tanpa perlu menulis banyak CSS custom.
- **Responsif dan Mobile-first:** Desain responsif lebih mudah diimplementasikan dengan kelas utilitas yang tersedia.
- **Kustomisasi Tinggi:** Konfigurasi yang sangat kustomisasi melalui file konfigurasi Tailwind.
- **Kurva Belajar yang Rendah:** Mudah dipelajari, terutama bagi pemula.

Kekurangan

- **Markup yang 'Berat':** Bisa menghasilkan HTML dengan banyak kelas utilitas, yang bagi sebagian orang mungkin terlihat berantakan.
- **Keunikan:** Mungkin memerlukan waktu untuk beradaptasi bagi pengembang yang terbiasa dengan pendekatan tradisional CSS.



Tailwind – Utility Class

Cara Menggunakan Utility Classes untuk Styling

Dengan Tailwind, Anda mengontrol tampilan elemen melalui kumpulan kelas utilitas yang menentukan properti CSS tertentu. Misalnya, text-center untuk teks terpusat, bg-blue-500 untuk warna latar belakang biru, dan p-4 untuk padding.

Responsiveness dan Mobile-first Design

Tailwind memudahkan implementasi desain responsif dengan prefix seperti sm:, md:, lg:, dan xl: sebelum kelas utilitas. Misalnya, md:text-center akan menerapkan text-center hanya pada medium devices dan lebih besar.

Pendekatan Komponen vs. Utility Class

Walaupun Tailwind mengutamakan kelas utilitas, Anda dapat membangun komponen kustom dengan menggabungkan kelas-kelas tersebut. Banyak pengembang menggunakan pendekatan ini untuk membuat komponen yang dapat digunakan kembali dengan @apply dalam CSS atau melalui komponen JavaScript.

Struktur Dasar dan Komponen Dasar Tailwind



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tailwind CSS Example</title>
  <link href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.19/dist/tailwind.min.css" rel="stylesheet">
</head>
<body>

<div class="max-w-md mx-auto bg-white rounded-xl shadow-md overflow-hidden md:max-w-2xl m-5">
  <div class="md:flex">
    <div class="md:flex-shrink-0">
      
    </div>
    <div class="p-8">
      <div class="uppercase tracking-wide text-sm text-indigo-500 font-semibold">Category</div>
      <a href="#" class="block mt-1 text-lg leading-tight font-medium text-black hover:underline">The Quick Brown Fox Jumps Over the Lazy Dog</a>
      <p class="mt-2 text-gray-500">The quick brown fox jumps over the lazy dog. A quick movement of the enemy will jeopardize six gunboats.</p>
    </div>
  </div>
</div>

</body>
</html>
```

- Sebetulnya Tailwind menyediakan kelas utilitas untuk hampir setiap kebutuhan styling, banyak UI dapat dibangun tanpa menulis CSS tambahan.
- Membuat layout card dengan shadow (shadow-md), border radius (rounded-xl), dan margin (m-5).
- Mengatur tampilan gambar (object-cover, w-full).
- Menggunakan typography untuk teks (text-sm, text-indigo-500, font-semibold, dll.). Membuat card responsif dengan md:flex dan md:flex-shrink-0.

Slicing Design

Untuk latihan slicing design yang membandingkan penggunaan Bootstrap dan Tailwind CSS, kita akan membuat sebuah card sederhana yang menampilkan gambar, judul, deskripsi, dan tombol. Desain ini akan menunjukkan bagaimana kedua framework tersebut digunakan untuk mencapai hasil yang serupa dengan pendekatan yang berbeda.



Colorful Card

This is a more colorful example card with Bootstrap that includes images, titles, descriptions, and buttons.

[Learn More](#)



Desain Target:

- Card dengan:
 - Gambar di bagian atas.
 - Judul di bawah gambar.
 - Deskripsi singkat di bawah judul.
 - Tombol di bagian bawah card.

Bootstrap vs Tailwind in Slicing



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bootstrap Colorful Card Example</title>
  <link href="https://stackpath.bootstrapcdn.com/boot
rel="stylesheet">
</head>
<body>

<div class="container py-5">
  <div class="card" style="width: 18rem;">
    
    <div class="card-body bg-light">
      <h5 class="card-title">Colorful Card</h5>
      <p class="card-text">This is a more colorfu
images, titles, descriptions, and buttons.</p>
      <a href="#" class="btn btn-info">Learn More
      </a>
    </div>
  </div>
</div>

</body>
</html>
```



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tailwind CSS Colorful Card Example</title>
  <link href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.19/dist/tailwind.min.css"
rel="stylesheet">
</head>
<body>

<div class="flex justify-center items-center h-screen">
  <div class="max-w-sm rounded overflow-hidden shadow-lg bg-white">
    
    <div class="px-6 py-4">
      <div class="font-bold text-xl mb-2 text-gray-800">Colorful Card</div>
      <p class="text-gray-600 text-base">
        Enjoy a more colorful example with Tailwind CSS, featuring an image, title,
        description, and a button with hover effects.
      </p>
    </div>
    <div class="px-6 pt-4 pb-2">
      <button class="bg-purple-500 hover:bg-purple-700 text-white font-bold py-2 px-4 rounded">
        Learn More
      </button>
    </div>
  </div>
</div>

</body>
</html>
```

Enjoy a more colorful example with Tailwind CSS, featuring an image, title, description, and a button with hover effects.



Thanks

Muhammad Syaifur Rohman, S.Kom, M.CS

