**Tugas Praktikum 05 |** **Template CSS**

Nama : Lailatul Izzah

Nim : 4123029

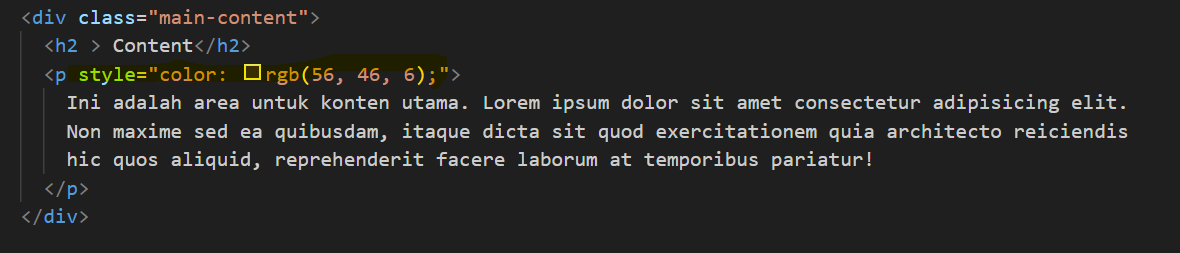
Kelas : B

Semester : 3

Link akun github : [https://github.com/lailIzza/PemrogramanWeb1/tree/main/Tugas04](https://github.com/lailIzza/PemrogramanWeb1/tree/main/Tugas043)

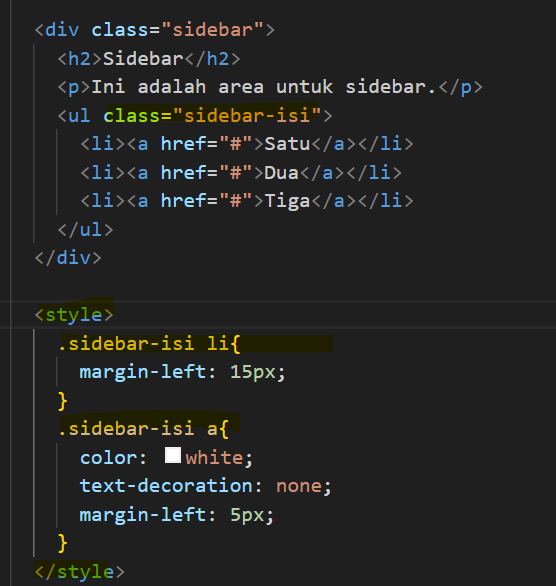
Metode penulisan CSS pada tugas kali ini :

1. Inline



Untuk penulisan dengan metode ini, bisa dilihat pada gambar di atas. Terlihat bahwa taq p terdapat atribut style, dimana style digunakan untuk styling CSS.

1. Internal



Pada metode internal , kita memerlukan menulis selector pada taq yang ingin diberi CSS. Untuk kode ini selector nya yaitu : class = “sidebar-isi’’. Dan penulisan css, diawali dengan memanggil selector dalam style.

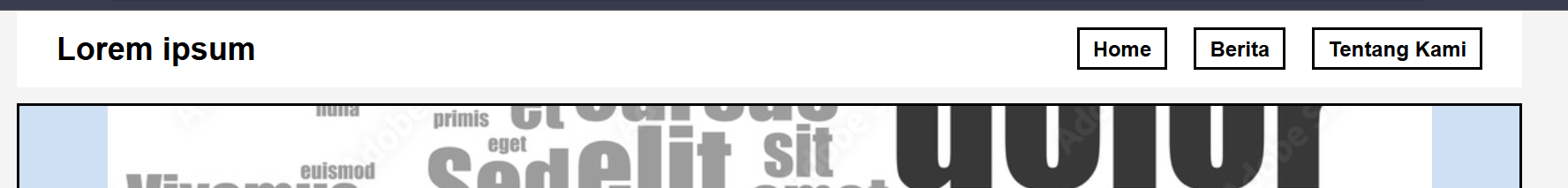
1. External



Pada metode external ini, kita perlu membuat file CSS pada halaman baru. Untuk memanggilnya, kita perlu menuliskan link. Untuk kode ini CSS nya berada di folder CSS dengan nama file template.css

Selanjutnya, kita akan membahas bagian website :

**Navbar**

****

Pada bagian ini, saya menggunakan 3 class untuk styling CSS yaitu : navbar, logo, dan nav-links.

Navbar

* **display: flex;**: Menggunakan *Flexbox* untuk membuat elemen di dalam navbar berada dalam satu baris.
* **justify-content: space-between;**: Mengatur elemen di dalam .navbar agar tersebar secara merata, dengan elemen pertama (.logo) berada di sebelah kiri, dan elemen kedua (.nav-links) di sebelah kanan.
* **align-items: center;**: Mengatur elemen di dalam navbar agar berada di tengah secara vertikal.
* **background-color: white;**: Mengatur warna latar belakang navbar menjadi putih.
* **padding: 15px 30px;**: Memberikan ruang kosong di dalam navbar, yaitu 15px pada bagian atas dan bawah, serta 30px pada bagian kiri dan kanan.
* **position: fixed;**: Menjadikan navbar tetap berada di bagian atas halaman meskipun pengguna menggulir ke bawah.
* **top: 0;**: Mengatur navbar untuk menempel di bagian atas halaman.
* **width: 90%;**: Mengatur lebar navbar agar sesuai dengan lebar .container dan berada di tengah halaman.
* **z-index: 1000;**: Memastikan navbar berada di lapisan teratas agar tidak tertutup elemen lain.

Logo

* **font-size: 1.5em;**: Mengatur ukuran font logo menjadi lebih besar agar lebih menonjol.
* **color: black;**: Mengatur warna teks logo menjadi hitam.

Nav-link

**.nav-links**:

* **list-style-type: none;**: Menghapus tanda titik pada setiap item daftar (<li>).
* **display: flex;**: Menampilkan tautan navigasi dalam satu baris secara horizontal.

**.nav-links li**:

* **margin-left: 20px;**: Memberikan jarak antar elemen daftar dalam navbar agar tidak terlalu berdekatan.

**.nav-links a**:

* **text-decoration: none;**: Menghilangkan garis bawah pada tautan.
* **color: black;**: Mengatur warna teks tautan menjadi hitam.
* **font-weight: bold;**: Menampilkan teks tautan dengan ketebalan lebih tinggi agar lebih jelas.
* **padding: 5px 10px;**: Memberikan ruang di dalam tombol tautan agar tampak lebih rapi.
* **border: 2px solid black;**: Menambahkan border hitam di sekeliling tombol tautan untuk memberikan tampilan seperti tombol.

**.nav-links a:hover**:

* **background-color: lightgray;**: Mengatur warna latar belakang tombol tautan menjadi abu-abu muda saat pengguna mengarahkan mouse (hover) ke atasnya, memberikan efek visual yang menarik saat di-hover.

**Header**

****

Pada bagian header, saya menggunakan elemen <header> dan class header-img untuk styling CSS.

Header

* **position: relative;**: Mengatur posisi elemen header sebagai relatif terhadap posisi standar di dokumen, memungkinkan elemen anak (seperti gambar) diatur di dalamnya.
* **width: 100%;**: Membuat elemen header memenuhi lebar layar penuh.
* **height: 300px;**: Mengatur tinggi header menjadi 300px, cukup besar untuk membuat gambar tampil menonjol.
* **background-color: #CFE0F5;**: Menambahkan warna latar belakang biru terang (#CFE0F5) untuk header sebagai cadangan jika gambar tidak dimuat.
* **border: 2px solid black;**: Memberikan border hitam di sekeliling header agar terlihat terpisah dari bagian lain.
* **overflow: hidden;**: Menyembunyikan bagian gambar yang melampaui area header.
* **display: flex; justify-content: center; align-items: center;**: Menggunakan Flexbox untuk menempatkan konten secara horizontal dan vertikal di tengah header.

Header-img

* **width: 100%; height: 100%;**: Menyesuaikan gambar agar memenuhi seluruh lebar dan tinggi header.
* **object-fit: cover;**: Mengatur gambar agar menyesuaikan dengan ukuran header tanpa distorsi. Bagian yang berlebih akan terpotong di tepi.
* **display: block;**: Menghapus ruang kosong default di sekitar gambar, membuatnya tampil seperti blok penuh.

**Main-content dan Sidebar**

****

Pada bagian ini saya memberikan elemen <div class="content"> adalah pembungkus utama untuk bagian main content dan sidebar. Elemen ini diformat dengan display: flex di CSS untuk menempatkan kedua elemen tersebut secara horizontal berdampingan. Selanjutnya class content, main-content, sidebae, dan sidebar-isi untuk styling CSS.

Content

* **display: flex;:** Menyusun main-content dan sidebar dalam satu baris horizontal**.**
* **margin: 20px 0;:** Memberikan ruang di atas dan bawah konten.
* **height: 500px;:** Mengatur tinggi konten utama.

Main-content

* **flex: 4;**: Mengatur agar area main-content memiliki porsi lebih besar dari sidebar. Dengan rasio 4:1, main-content mengambil 4 bagian dari 5, sedangkan sidebar mengambil 1 bagian.
* **padding: 20px;**: Memberikan ruang di dalam main-content agar teks tidak menempel ke tepi elemen.
* **background-color: rgb(191, 240, 13);**: Mengatur warna latar belakang hijau cerah untuk main-content.
* **margin-right: 10px;**: Memberikan jarak antara main-content dan sidebar.
* **border: 2px solid #111111;**: Memberikan border hitam pada main-content agar lebih jelas terlihat.

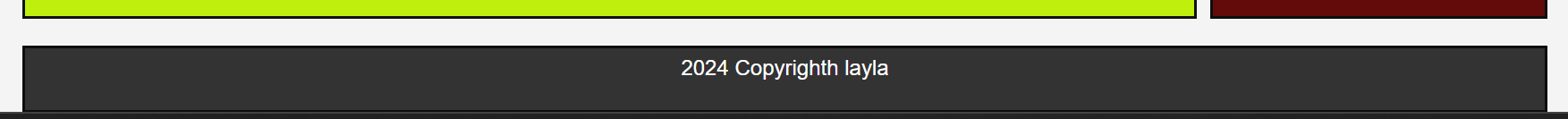
Sidebar

* **flex: 1;**: Mengatur sidebar agar mengambil porsi lebih kecil, yaitu 1 dari total 5 bagian.
* **padding: 20px;**: Memberikan ruang di dalam sidebar agar konten tidak menempel ke tepi.
* **background-color: rgb(99, 11, 11);**: Mengatur warna latar belakang sidebar menjadi merah gelap.
* **border: 2px solid #0d0d0d;**: Memberikan border hitam pada sidebar untuk memperjelas batasannya.
* **color: white;**: Mengatur warna teks dalam sidebar menjadi putih agar kontras dengan latar belakang.

Sidebar-isi

* **color: white;**: Mengatur warna tautan menjadi putih agar kontras dengan warna latar belakang sidebar.
* **text-decoration: none;**: Menghilangkan garis bawah pada tautan.
* **margin-left: 5px;**: Memberikan jarak tambahan antara tautan satu dengan lainnya untuk tampilan yang lebih bersih.

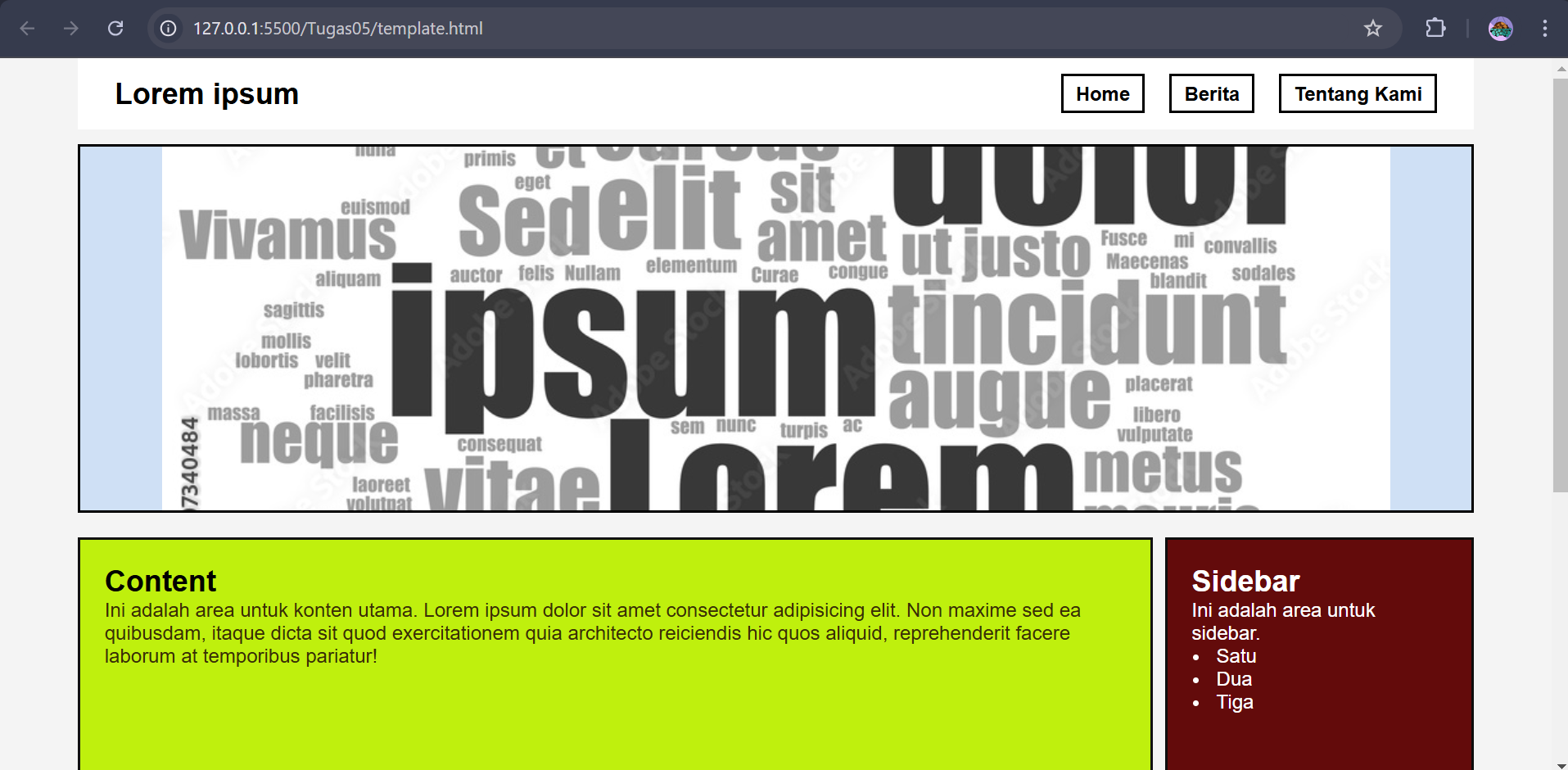
**Footer**

****

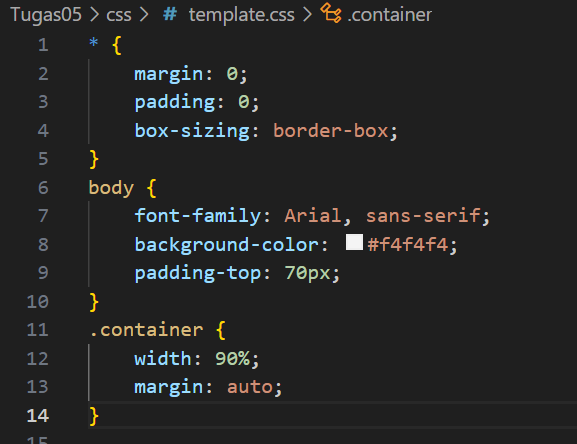
Pada bagian ini menggunakan class footer untuk styling CSS.

* **height: 50px;**: Mengatur tinggi footer menjadi 50px, cukup untuk menampung teks dengan nyaman.
* **text-align: center;**: Memposisikan teks di tengah secara horizontal.
* **padding: 5px;**: Memberikan ruang dalam footer agar teks tidak terlalu menempel ke tepi atas dan bawah.
* **background-color: #333;**: Memberikan warna latar belakang abu-abu gelap (#333) untuk kontras dengan bagian atas halaman.
* **color: white;**: Mengatur warna teks menjadi putih agar mudah terbaca di atas latar belakang gelap.
* **margin-top: 20px;**: Memberikan jarak antara footer dengan konten di atasnya.
* **border: 2px solid #0d0d0d;**: Memberikan border hitam di sekitar footer untuk mempertegas batasannya.

Sehingga hasil penuh dari template CSS adalah seperti ini :



Selain itu berikut kode yang mengatur dan memastikan tampilan halaman lebih seragam, rapi, dan responsif terhadap perubahan ukuran layar.



Global CSS Reset (\*) : Ini adalah *universal selector* yang menerapkan aturan CSS untuk semua elemen pada halaman.

* **margin: 0; padding: 0;**: Secara default, browser memberikan margin dan padding pada elemen tertentu (seperti <h1>, <p>, <ul>). Aturan ini mengatur margin dan padding semua elemen menjadi nol agar kita bisa mengatur layout dari nol dan memastikan elemen-elemen berada di posisi yang tepat tanpa terpengaruh oleh margin dan padding default.
* **box-sizing: border-box;**: Properti ini mengubah cara penghitungan lebar dan tinggi elemen. Dengan border-box, padding dan border termasuk dalam penghitungan width dan height, sehingga elemen tidak akan meluas ke luar dari lebar yang sudah ditentukan. Ini memudahkan dalam mendesain layout karena padding tidak menambah ukuran total elemen.

Body

* **font-family: Arial, sans-serif;**: Mengatur font yang digunakan di seluruh halaman menjadi Arial. Jika Arial tidak tersedia, font sans-serif lainnya akan dipakai.
* **background-color: #f4f4f4;**: Mengatur warna latar belakang pada seluruh halaman menjadi warna abu-abu terang (#f4f4f4). Ini memberikan kesan yang lebih lembut dibandingkan warna putih polos.
* **padding-top: 70px;**: Memberikan ruang kosong di bagian atas halaman. Ini dilakukan untuk menghindari bagian atas halaman tertutup oleh navbar yang menggunakan position: fixed. Dengan padding-top, isi halaman tetap bisa terlihat jelas tanpa tertutup navbar.

Container

* **width: 90%;**: Mengatur lebar elemen .container menjadi 90% dari lebar halaman browser. Ini berarti .container tidak memenuhi layar penuh, memberikan ruang di kiri dan kanan agar tampilan lebih rapi dan mudah dibaca.
* **margin: auto;**: Mengatur margin secara otomatis pada elemen .container. Dengan pengaturan ini, elemen .container akan berada di tengah secara horizontal dalam halaman, karena margin kiri dan kanan dibagi sama rata.

**Contoh penggunaan template :**