**FILE INI MERUPAKAN RANGKUMAN YANG DIBUAT OLEH THIRAFI ILMAM UNTUK DAPAT MENGINGAT DAN MEMPELAJARI FRONT END WEBSITE DEV**

**Update : 7/7/2023**

**Struktur Dasar Dokumen HTML:**

<!DOCTYPE html>

<html>

<head>

<title>Judul Halaman</title>

</head>

<body>

Konten halaman web...

</body>

</html>

**Tag Dasar:**

<html>: Tag utama yang menandakan awal dan akhir dari dokumen HTML.

<head>: Bagian kepala dokumen yang berisi informasi metadata dan tautan ke file eksternal.

<title>: Menentukan judul halaman yang akan ditampilkan di bilah judul browser.

<body>: Bagian tubuh dokumen yang berisi konten yang akan ditampilkan di halaman web.

**Tag Teks:**

<h1> hingga <h6>: Menandai judul dengan ukuran dan tingkatan yang berbeda.

<p>: Paragraf teks.

<strong>: Menebalkan teks.

<em>: Miringkan teks.

<u>: Menggarisbawahi teks.

**Tag Tautan dan Gambar:**

<a href="URL">Teks Tautan</a>: Membuat tautan ke URL eksternal atau internal.

<img src="nama-gambar.jpg" alt="Deskripsi Gambar">: Menampilkan gambar dengan sumber dan deskripsi yang ditentukan.

**Tag Daftar:**

<ul>: Membuat daftar tak terurut.

<ol>: Membuat daftar terurut.

<li>: Item dalam daftar.

**Tag Tabel:**

<table>

<tr>

<th>Kolom 1</th>

<th>Kolom 2</th>

</tr>

<tr>

<td>Baris 1, Kolom 1</td>

<td>Baris 1, Kolom 2</td>

</tr>

</table>

**Tag Form :**

<form action="proses.php" method="POST">

<label for="nama">Nama:</label>

<input type="text" id="nama" name="nama" required><br>

<label for="email">Email:</label>

<input type="email" id="email" name="email" required><br>

<input type="submit" value="Kirim">

</form>

Contoh ini menunjukkan form dengan input teks dan email, serta tombol submit.

<input>: Menambahkan berbagai jenis input seperti teks, password, checkbox, radio, dll.

<textarea>: Menampilkan area input teks yang lebih besar.

<select>: Membuat dropdown atau daftar pilihan.

<option>: Menentukan opsi dalam elemen <select>.

<label>: Menandai label untuk elemen formulir.

**Tag Lainnya:**

<div> :Mengelompokkan elemen menjadi sebuah blok.

<span> : Menandai sebagian teks untuk diberikan gaya atau perilaku khusus.

<br> : Membuat baris baru (break).

**Atribut Lain**

class: Menentukan kelas CSS untuk elemen.

id: Menentukan ID unik untuk elemen.

style: Menentukan gaya CSS inline untuk elemen.

src: Menentukan sumber file untuk elemen seperti gambar, audio, video, dll.

href: Menentukan URL tujuan untuk tautan.

target: Menentukan di mana dokumen terkait akan ditampilkan (misalnya, di jendela baru atau di frame).

**Selektor:**

Nama elemen: Menggambarkan elemen yang ingin diubah gayanya. Contoh: p, h1, div.

ID: Menggunakan ID yang unik untuk elemen tertentu. Contoh: #id-elemen.

Kelas: Menggunakan kelas yang sama untuk beberapa elemen. Contoh: .kelas-elemen.

Properti dan Nilai:

Properti: Menentukan aspek tertentu yang ingin diubah. Contoh: color, font-size, background.

Nilai: Menentukan nilai untuk properti tertentu. Contoh: red, 14px, url(gambar.jpg).

p {

color: blue;

}

h1 {

font-size: 24px;

}

body {

background-image: url(background.jpg);

}

#id-elemen {

font-weight: bold;

}

.kelas-elemen {

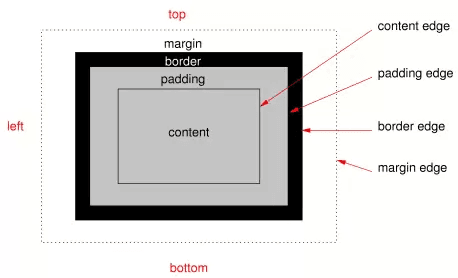
text-align: center;

}

**CSS**

**Box Model:**

Box Model merupakan konsep dalam CSS yang menggambarkan elemen HTML sebagai sebuah kotak yang terdiri dari beberapa bagian: konten (content), padding, border, dan margin.



Konten (Content): Bagian dalam kotak yang menampung isi elemen seperti teks, gambar, atau elemen lainnya.

Padding: Ruang antara konten dan tepi elemen. Padding dapat diberi nilai dalam piksel (px), persen (%), atau unit lainnya.

Border: Garis yang mengelilingi konten dan padding elemen. Border memiliki properti seperti tebal, warna, dan jenis garis (solid, dashed, dll.).

Margin: Ruang di luar border elemen. Margin dapat memberikan jarak antara elemen dengan elemen lainnya.

.box {

width: 200px;

height: 100px;

padding: 20px;

border: 1px solid black;

margin: 10px;

}

**Display:**

Display adalah properti CSS yang mengontrol bagaimana elemen HTML ditampilkan di halaman web. Properti ini mempengaruhi perilaku tata letak dan penempatan elemen.

**Flex**

Elemen dengan display flex mengaktifkan tata letak fleksibel (flexbox) yang memungkinkan penyesuaian dan penempatan elemen secara responsif. Elemen-elemen dengan display flex dapat diatur dalam satu baris atau satu kolom, serta memiliki kemampuan mengatur ruang kosong dan penyebaran ruang yang fleksibel.

Contoh:

.table-element {

display: table;

}

**Block**

Elemen block (misalnya <div>, <p>, <h1>) secara default mengambil ruang penuh secara horizontal dan akan memulai di baris baru. Anda dapat mengatur lebar dan tinggi elemen block, serta menerapkan margin dan padding pada semua sisi.

Contoh:

.block-element {

display: block;

width: 300px;

height: 200px;

}

**Table-cell**

Elemen dengan display table-cell akan ditampilkan sebagai sel dalam tabel. Elemen-elemen dengan display ini akan mengikuti aturan tata letak tabel dan dapat diatur lebar dan tingginya.

Contoh:

.table-cell-element {

display: table-cell;

width: 100px;

height: 50px;

}

**Flex**

Elemen dengan display flex mengaktifkan tata letak fleksibel (flexbox) yang memungkinkan penyesuaian dan penempatan elemen secara responsif. Elemen-elemen dengan display flex dapat diatur dalam satu baris atau satu kolom, serta memiliki kemampuan mengatur ruang kosong dan penyebaran ruang yang fleksibel.

Contoh:

.flex-container {

display: flex;

flex-direction: row;

justify-content: space-between;

}

**Grid**

Elemen dengan display grid mengaktifkan tata letak grid yang memungkinkan penempatan elemen dalam baris dan kolom yang terstruktur. Elemen-elemen dengan display grid dapat diatur dalam grid yang kompleks dengan kontrol posisi yang lebih granular.

Contoh:

.grid-container {

display: grid;

grid-template-columns: 1fr 1fr 1fr;

grid-gap: 10px;

}

**Inline**

Elemen inline (misalnya <span>, <a>, <img>) tidak memulai di baris baru dan hanya menggunakan ruang yang diperlukan oleh konten di dalamnya. Anda tidak dapat mengatur lebar dan tinggi elemen inline, namun dapat menerapkan margin dan padding hanya pada sisi horizontal.

Contoh:

.inline-element {

display: inline;

}

**Inline-Block**

Elemen inline-block memiliki sifat inline dan dapat diatur lebar dan tingginya serta menerapkan margin dan padding pada semua sisi.

Contoh:

.inline-block-element {

display: inline-block;

width: 150px;

height: 100px;

}

**None**

Elemen dengan display none akan sepenuhnya disembunyikan dari halaman web. Elemen tersebut tidak akan memengaruhi tata letak dan tidak akan dilihat oleh pengguna.

Contoh:

.hidden-element {

display: none;

}

**Combinator:**

Combinator dalam CSS adalah sebuah konsep yang digunakan untuk menggabungkan selektor dan memilih elemen spesifik berdasarkan hubungan atau posisi relatif mereka terhadap elemen lain. Terdapat empat jenis combinator yang umum digunakan:

**Selector Descendant (Space)**

Selector descendant memilih elemen-elemen yang berada di dalam elemen lain (elemen anak, elemen cucu, dan seterusnya).

Contoh:

div p {

/\* Styling untuk elemen <p> yang berada di dalam elemen <div> \*/

}

**Selector Child (>)**

Selector child memilih elemen-elemen yang langsung berada sebagai anak dari elemen lainnya.

Contoh:

div > p {

/\* Styling untuk elemen <p> yang menjadi anak langsung dari elemen <div> \*/

}

**Selector Adjacent Sibling (+)**

Selector adjacent sibling memilih elemen yang berada tepat setelah elemen lain, dengan asumsi keduanya memiliki elemen yang sama sebagai elemen induk.

Contoh:

h1 + p {

/\* Styling untuk elemen <p> yang berada tepat setelah elemen <h1> \*/

}

**Selector General Sibling (~)**

Selector general sibling memilih elemen-elemen yang berada setelah elemen lain, dengan asumsi keduanya memiliki elemen yang sama sebagai elemen induk.

Contoh:

h1 ~ p {

/\* Styling untuk elemen-elemen <p> yang berada setelah elemen <h1> \*/

}