**TAG HTML** (yang paling umum di gunakan)

1. **div**

Tag yang paling sering digunakan untuk membungkus atau memblok-blokan & mengkolom-kolimkan suatu elemen html.

1. **ul**

Tag html yang berfungsi sama dengan div, namun biasanya di gunakan untuk membungkus web yang mempunyai konten yang sama lebih dari 1

Contoh:

<ul>

<li> //konten 1 </li>

<li> //konten 2 </li>

<li> //konten 3 </li>

<li> //konten 4 </li>

</ul>

1. **ol**

Tag html yang berfungsi sama dengan div dan ul, namun mempunyai default angka di setiap li-nya, biasanya di gunakan untuk navigasi pagination atau pager .

Contoh:

<ol>

<li> //konten 1 </li>

<li> //konten 2 </li>

<li> //konten 3 </li>

<li> //konten 4 </li>

</ol>

1. **span**

tag html yang berfungsi sama dengan div, namun elemenn ini lebih sering digunakan untuk suatu elemen satuan (styling text) yang bukan berbentuk grid atau kolom, biasanya digunakan agar suatu elemen satuaan tersebut bisa di styling.

contoh:

<p>lorem <span class=”bold-text”>**Ipsum**</span> dolor sit amet </p>

1. **a**

tag yang berfungsi sebagai link dalam suatu element web, untukmenambahkan link di gunakan atribut href dalam tag tersebut.

Tag ini dapat membungkus tag apapun yng membutuhkan link

Contoh:

<a href=”http://www.tempo.co” > klik disini </a>

<a href=”http://www.tempo.co”><img src=”images/logo.jpg”></a>

1. **img**

Tag yang digunakan untuk memanggil image dalam elemen web, disertai dengan atribut src. Tidak memiliki tag penutup.

Contoh :

<img src=”images/logo.jpg”>

1. **h1,h2,h3,h4,h5**

Tag yang biasanya digunakan untuk mendefinisikan judul dalam elemen html.

Contoh :

<h1>Judul</h1>

<h2>Judul</h2>

<h3>Judul</h3>

<h4>Judul</h4>

<h5>Judul</h5>

1. **input**

Tag yang digunakan untuk membuat formulir dalam html, mempunyai atribut yang berbeda-beda fungsinya, dan harus diawalin dengan elemen form.

Contoh:

<form>

<input type=”text” />

<input type=”password” />

<input type=”file” />

<input type=”submit” />

</form>

1. **p**

Merupakan Tag yang membungkus tulisan atau paragraph pada umumnya di html.

Contoh :

<p> Lorem ipsum dolor sit amet </p>

1. **strong**

Merupakan tag untuk membuat tulisan bold di dalam html

Contoh :

<p> Lorem **<strong>ipsum</strong>** dolor sit amet </p>

**Comment dalam html**

Kita dapat membuat komen dalam elemen html dengan cara membungkus seperti <!—komen 🡪, dapat juga digunakan untuk mendisable html yang tidak terpakai

**CSS**

Merupakan kode-kode styling yang **PALING PENTING DIKUASAI** oleh seorang front-end developer, karena css ini merupakan dasar-dasar styling struktur web sebelum kita ingin membuat web responsive, atau membuat animasi di web, pengetahuan css ini harus kuat, karena dengan css ini kita dapat membuat semua model desain web apapun desainnya.

Cara panggil css dari html:

<link rel=”stylesheet” type=”text/css” media=”all” href=”nama-file.css” >

disarankan:

Sebelum menuliskan css tambahkan dulu di paling atas kode “@charset "utf-8"; ”Untuk mendefinisikan css (sama kasusnya dengan mendefifinisikan html dengan doctype pertama kali)

cara penulisan css:

jika diawali titik(.) maka itu adalah perintah memanggil class

contoh:

**file.html**

<div class=”nama-class”>blablabla</div>

**file.css**

.nama-class{

/\* styling \*/

}

jika diawali titik(#) maka itu adalah perintah memanggil id

contoh:

**file.html**

<div id=”nama-id”>blablabla</div>

**file.css**

#nama-id{

/\* styling \*/

}

dapat pula langsung styling elemen tanpa class dan id

contoh:

**file.html**

<div> blablabla</div>

**file.css**

div{

/\* styling \*/

}

dalam latihan pembuatan code CSS mau tidak mau, suka tidak suka, tidak akan ada jalan lain selain mencoba sendiri styling code css, karena setiap front-end developer bisa saja mempunyai gaya yang berbeda dalam styling-nya.

selain itu styling ini menurut saya lebih bersifat tricky, dalam arti bisa saja jika di lihat secara tampilan code kita benar sesuai dengan desain, namu jika kita lihat code sorce css-nya akan terlihat jika ada pengembangan konten web css tidak efisien sehingga akan menyulitkan kita di kedepannya.

Beberapa hal tersebut biasanya terjadi karena banyak factor, beberapa yang paling umum yaitu : -banyaknya penambahan class baru, -menggunakan nilai (-). Namun masalah-masalah lainnya juga hanya bisa ditemui saat kita mencoba langsung daripada hanya menghafal tag, karena dalam hal css ini menurut saya, memahami cara kerja dan logikanya lebih penting daripada hanya menghafal tag.

Jika sudah bisa mengusai css, maka selanjutnya mempelajari javascript atau apapun di dalam front end developer akan semakin mudah, selain itu para front end juga dapat menggunakan framework css yang terkenal(grid 960gs, bootstrap, foundation) sehingga prosses pembuatan web menjadi lebih singkat dan mudah, alasan terbesar saya tidak langsung mengajarkan framework karena menurut saya hanya akan percuma dan buang-buang waktu jika mempelajari framework tapi tidak mengerti cara kerja, saya sendiri termasuk front end yang tidak suka menggunakan framework, karena jika kita mengerti kita juga bisa membuat framework sendiri, dan itu sangat-sangat tidak menyulitkan, selain itu jika tidak menggunakan framework, tidak ada keterbatasan desain dalam styling css.

**Tag-tag dalam css**

Berikut Berberapa code style css.

Untuk beberapa css yang tricky akan saya labeli dengan tulisan **TRICKY**

Untuk penjelasan lengkapnya dapat bertanya langsung.

**Margin**

Jarak luar antar elemen

Margin-top:nilai;

Margin:right:nilai;

Margin-bottom:nilai;

Margin-left:nilai;

Penulisan singkat:

Margin: top right bottom left ; // contoh: margin: 2px 3px 7px 9px;

Atau

Margin: atas& bawah kanan&kiri ; // contoh: margin: 10px 15px;

Atau

Margin: atas kanan&kiri bawah ; // contoh: margin: 10px 15px 12xx;

**Padding(TRICKY)**

Jarak dalam suatu elemen

padding-top:nilai;

padding:right:nilai;

padding-bottom:nilai;

padding-left:nilai;

Penulisan singkat:

padding: top right bottom left ; // contoh:Padding: 2px 3px 7px 9px;

Atau

Padding: atas& bawah kanan&kiri ; // contoh: Padding: 10px 15px;

Atau

Padding: atas kanan&kiri bawah ; // contoh: Padding: 10px 15px 12px;

**Float (TRICKY)**

Menggeser suatu elemen html

Float: left;

Float: right;

Float:none;

**Display**

Status display sebuah elemen html

Display:block;

Display:none; // elemen html tidak ditampilkan

**Width**

Mengatur lebar elemen

Width: nilai;

Min-width: nilai; //lebar minimal

Max-width: nili; //lebar maksimal

**Height**

Mengatur tinggi elemen

height: nilai;

Min-height: nilai; //tinggi minimal

Max-height: nili; //tinggi maksimal