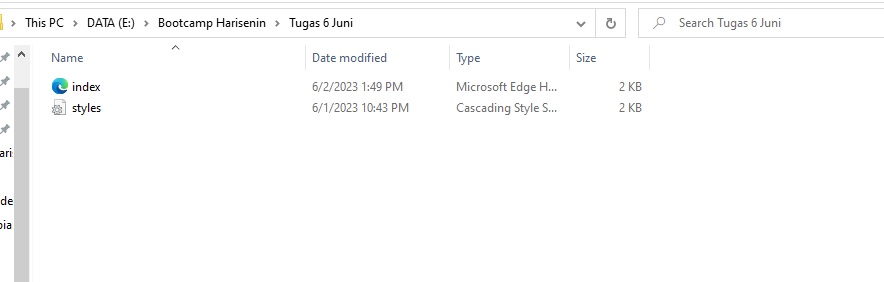
Tugas 6 Juni 2023

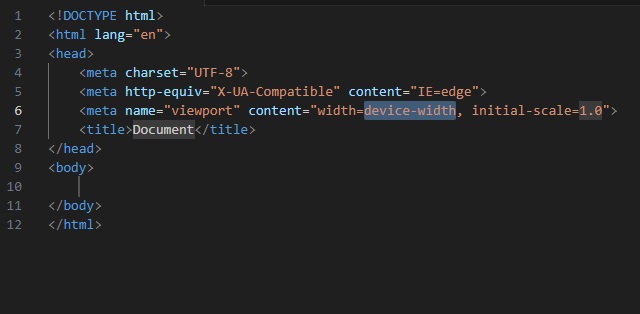
**Halaman Landing Page (section atas)**

Oleh : Guido Gusthi Abadi

1. Siapkan file html dan css nya terlebih dahulu, dalam satu folder.

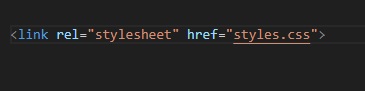


Ini membantu agar kita mudah mengatur lokasi file ketika ngoding, tidak perlu import-import dari eksternal.

1. Tulis kode html mulai dari DOCTYPE hingga closing tag </html>

Ini adalah format dasar untuk coding, yang mengandung <html lang=”en”> berarti bahasa yang digunakan adalah English.

1. Tautkan dengan file css di dalam folder kita



Gunanya adalah agar kode HTML kita bisa berinteraksi atau nyambung dengan style di CSS kita. <…href=”styles.css”> adalah sumber yang ingin ditautkan, namanya disesuaikan dengan nama file CSS yang kita buat. Seandainya kita membuat file CSS dengan nama “styling.css”, maka di href tersebut kita tulis juga demikian.

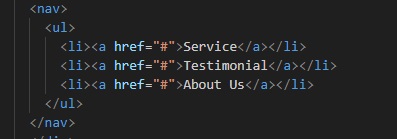
1. Membuat kelas untuk bagian atas serta logo, yang nanti akan ditempatkan pada posisi kiri atas dari halaman.



Untuk menambahkan gambar di HTML, kita bisa menggunakan tag <img> yang sifatnya self-closing tag, atau tag yang tidak membutuhkan tag penutup. Untuk memasukkan gambar dari link luar, kita bisa menggunakan properti “src” yang berarti source, lalu diisi dengan link referensi gambar. Sementara, properti “alt” pada gambar di atas berarti alternatif kata yang muncul apabila gambar gagal dimuat. Misalnya, jika gambar dari link yang diberikan tidak keluar, maka pada kasus ini yang keluar adalah kata “logopsi”.

Sementara, properti “width” berarti lebar dari gambar yang kita import. Biasanya, gambar berukuran sangat besar apabila kita mengambil dari link luar, sehingga harus disesuaikan terlebih dahulu. Terakhir, properti “class” adalah properti yang digunakan untuk memudahkan kita memberi style di CSS nanti.

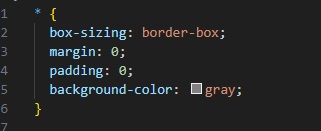
1. Membuat navigation bar untuk halaman



Sekarang kita akan membuat bagian untuk navigasi di pojok kanan atas halaman. Tag <nav> berguna untuk memudahkan kita mengatur dan memanipulasi bagian navbar nantinya. Sementara, tag <ul> adalah unordered list, yang artinya kita membuat list tanpa ada arti seperti peringkat atau urutan, hanya sekedar list saja. Di dalam list, saya memberikan tag <li> sebanyak tiga kali, yang berarti saya ingin membuat tiga list untuk bagian ini.

Sementara, tag <a> adalah tag yang digunakan agar konten di dalamnya dapat diklik dan dapat mengarah ke sumbernya. Fungsi properti “href=” adalah untuk memasukkan sumber yang ingin diarahkan, ketika konten diklik. Semua tag disini, baik tag <nav>, <ul>, <li> dan <a> merupakan tag yang harus diberikan tag penutup, misalnya </nav> dan </ul>.

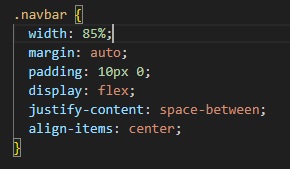
1. Mulai mengatur CSS, dimulai dari untuk halaman secara universal



Selector “\*” berarti kita styling keseluruhan elemen dalam halaman, atau secara universal. Ini membantu kita untuk membuat halaman lebih rapi dan sesuai dengan keinginan. Styling “box-sizing: border-box;” berarti kita menyesuaikan halaman sehingga padding dan border sudah include di dalam width dan height.

Setelah itu, styling “margin” dan “padding” yang sama-sama 0 berarti kita menghilangkan semua padding default (bawaan dari web) sehingga kita bisa lebih mudah mengaturnya nanti, tidak menjadi ada kurang-tambah karena bawaan webnya. Terakhir, styling “background-color: gray” adalah warna background untuk seluruh halaman, yang berarti abu-abu.

1. Mengatur bagian navbar

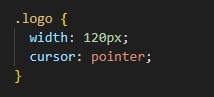


Disini kita mulai mengatur navbar pada CSS, karena “navbar” adalah properti, dan bukan tag, maka kita menggunakan titik (.) pada selector. Sehingga seluruh elemen pada class navbar di HTML akan diatur. Styling “width” adalah ukuran tulisan-tulisannya secara horizontal (memanjang). Kemudian, “margin: auto” akan membuat seluruh bagian navbar ini berada di tengah elemennya secara horizontal.

Kemudian, “padding: 10px 0;” membuat padding bagian atas dan bawah menjadi 10 pixels, dan kiri-kanannya adalah 0. Lalu, styling “display:flex” akan membuat elemen navbar menjadi flex container, sehingga kita bisa mengaturnya dengan aturan flexbox. Styling “justify-content: space-between;” memberikan jarak antara text content di navbar, karena pada awalnya semuanya rapat.

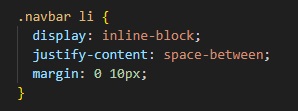
Terakhir, kita akan membuat konten-konten di navbar menjadi tengah secara vertical, sehingga digunakan styling “align-items: center”.

1. Mengatur Logo



Bagian ini simple saja, saya hanya mengatur lebar logo sebesar 120 pixels agar sedikit lebih besar daripada tulisan di navbar, ini dilakukan dengan styling “width: 120px”. Lalu, “cursor:ponter” maksudnya adalah ketika kita mengarahkan mouse ke logo, mouse yang tadinya berbentuk tanda panah akan berubah menjadi pointer (jari telunjuk).

1. Mengatur list di navbar

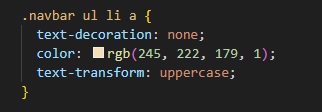


Sebelumnya, kita sudah mengatur bagian .navbar secara keseluruhan. Kali ini, saya ingin mengatur hanya bagian pada list saja, sehingga kita men-select dengan “.navbar li” agar tidak keseluruhan item di dalam class “navbar” terpengaruh oleh styling ini.

Bagian styling “display: inline-block;” gunanya adalah untuk membuat keseluruhan item pada <li> itu sejajar, karena secara *default* tampilan list ini berurutan ke bawah. Lalu, karena secara *default* juga jarak antara <li> satu dengan lainnya itu rapat, maka kita beri styling “space-between” pada bagian “justify-content” sehingga antara list yang satu dengan list lainnya itu memiliki jarak atau spasi.

Terakhir, kita mengatur margin menjadi 0 secara atas dan bawah, namun secara menyamping ada sedikit jarak yaitu 10pixels. Agar list-list ini dapat semakin terlihat dalam bentuknya masing-masing.

1. Mengatur bagian a, li, ul dan .navbar

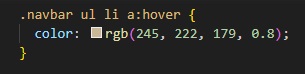


Mungkin sekilas tampak mirip, apa perbedaan antara memilih “.navbar ul li a”, dengan “.navbar” saja dan dengan “.navbar li” ? Perbedaannya adalah, jika kita memilih hanya “.navbar”, maka itu mencakup keseluruhan bagian di dalamnya. Kemudian, “.navbar li” hanya memilih bagian <li> saja untuk dimodifikasi.

Sementara, selector “.navbar ul li a” ini mencakup hingga bagian link yang ada pada <a href=””> sehingga bagian tersebut juga ikut dimodifikasi. Dalam hal ini, saya menggunakan styling “text-decoration: none;” untuk menghilangkan semua dekorasi pada teks, misalnya *underline*. Pada warna, saya menggunakan “rgb” sehingga saya bisa mengatur opasitas dari warna tersebut untuk keperluan hover nanti.

Styling “text-transform: uppercase;” bertujuan untuk mengubah semua teks yang ada menjadi huruf capital semua. Ini lebih nyaman dan enak dilihat untuk bagian navigasi.

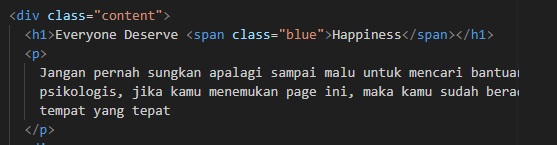
1. Mengatur bagian navigasi ketika di-hover



Seperti yang sudah saya sebutkan pada bagian 10 tadi, saya menggunakan “rgb” pada styling warna atau color untuk mengatur opasitas ketika di-hover. Selector “.navbar ul li a” diberi tambahan “:hover” untuk memodifikasi bagian tersebut ketika kita hover. Hover adalah tindakan mengarahkan kursor ke elemen.

Dan disini, saya menurunkan opasitas dari warna dari 1 menjadi 0.8, sehingga ketika kita mengarahkan kursor ke bagian navigasi, warnanya akan sedikit berubah.

1. Mengisi konten halaman



Baik, kali ini saya membuat <div> baru untuk bagian konten, sehingga saya mengisi class dari <div> tersebut dengan nama “content”. Di dalam konten, bisa dilihat saya menggunakan tag <h1> agar pada bagian ini, tulisan terlihat besar. Sementara tag <span class=”blue”> ditujukan agar saya bisa memodifikasi konten yang ada di dalam (nested) tag ini, yaitu tulisan **Happiness**.

Di bawah tag <h1>, saya menambahkan teks konten lain dengan ukuran yang lebih kecil, saya menggunakan tag <p> untuk teks ini.

1. Memodifikasi konten halaman



Tadi kita sudah mengisi konten yang berada pada <div> dengan class “content”. Sehingga di CSS, saya memanggilnya dengan “.content” untuk dimodifikasi. Saya mengatur width yaitu lebar konten sebesar 100% agar sesuai dengan element parentnya. Sementara, styling “position: absolute;” untuk mengatur posisi dari konten agar sesuai dengan elemen terdekat dari konten ini.

Kemudian, styling “top: 50%;” mengatur jarak atas dari konten terhadap elemen terdekat menjadi sebesar 50%, atau setengah dari keseluruhan tinggi konten. Disusul dengan styling “translateY(-50%);” untuk mengatur posisi konten sebesar 50% dari tingginya sendiri pada garis sumbu Y (vertikal).

Seperti yang sebelumnya sudah dilakukan, styling “text-align” menjadi “center” adalah agar posisi konten berada di tengah secara horizontal. Dan terakhir, warna yang saya pilih adalah warna khaki.

1. Mengatur warna salah satu tulisan di <h1>

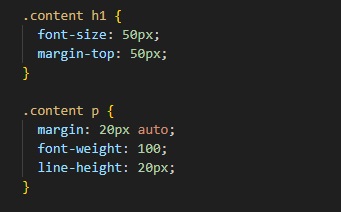


Seperti yang saya katakan tadi, saya memberi tag <span> dengan class “blue” pada tulisan “Happiness” sehingga tulisan tersebut berbeda dari tulisan lain yang ada di <h1>. Dalam hal ini, saya memberi warna yang tidak jauh berbeda, yaitu wheat. Kalau tidak melihat langsung, mungkin sulit mengetahui perbedaannya. Ini hanya untuk membuat tampilan teks konten lebih menarik saja, tidak ada tujuan lain dari segi fungsional atau dari segi teknis.



Nah, bagaimana? Terlihat kan perbedaannya? Inilah tujuan dari memberikan tulisan “Happiness” ini tag <span> dengan class “blue”. Kok “blue”? Awalnya, saya ingin memberi warna biru, tetapi ternyata tidak cocok, jadi menggunakan warna lain saja hehe.

1. Mengatur bagian <h1> dan <p>



Pada bagian ini, saya mengatur bagian h1 dan p pada “content”. Untuk h1, saya memberi font sebesar 50 pixels dengan menggunakan styling “font-size: 50px;”, sementara tujuan dari styling “margin: top” adalah agar konten dari <h1> memiliki margin sebesar 50 pixels di bagian atasnya.

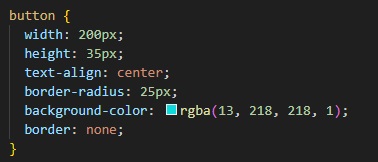
Sementara, margin pada bagian <p> saya beri sebesar 20 pixels pada bagian atas dan bawah, sementara kiri dan kanannya saya atur menjadi *auto* agar isi konten dari <p> berada pada tengah-tengah secara horizontal di kontainernya. Sementara, styling “line-height: 20px;” bertujuan untuk mengatur jarak atas antara kalimat menjadi sebesar 20 pixels. Maksud dari jarak atas ini adalah ketika <p> memiliki sepanjang dua baris kata, maka jarak antara baris di bawah ke baris yang atas adalah 20 pixels.

1. Menambahkan tombol pada halaman dengan HTML



Tentu saja, halaman ini membutuhkan button untuk *Call to Action* (CTA), sehingga di HTML, kita bisa menggunakan tag <button>, sementara properti type-nya adalah “button” juga, yang berarti tombol ini adalah tombol biasa yang tidak dimaksudkan untuk tujuan tertentu di database seperti “submit”.

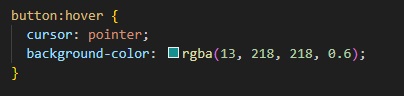
1. Styling tombol di CSS



Tombol ini saya atur sehingga memiliki lebar 200 pixels dengan styling “width: 200px;” kemudian tingginya menjadi 35 pixels dengan styling “height: 25px;”, sehingga tombolnya akan terlihat memanjang. Kemudian, styling “text-align: center;” bertujuan agar text yang ada berada pada tengah-tengah tombol.

Sementara, styling “border-radius: 25px;” tujuannya adalah agar sudut-sudut tombol tersebut melengkung 1/4 lingkaran, karena secara *default*, tombol memiliki sudut tajam 90 derajat. Styling “background-color” adalah untuk warna tombolnya, yang mana saya memilih biru dengan menggunakan “rgba” sehingga saya bisa mengatur opasitasnya untuk keperluan hover.

Terakhir, styling “border: none;” tujuannya adalah agar garis-garis tombol hilang, sehingga tombol tampil tanpa garis (lebih clean).

1. Styling tombol ketika hover

Seperti pada bagian navigasi, untuk keperluan UI dan UX yang enak, saya memberikan styling hover pada bagian button atau tombol. Yaitu apabila kursor diarahkan, maka kursor akan menjadi pointer, ini dilakukan dengan styling “cursor: pointer;” seperti di atas. Sementara, bagian “background-color” dengan rgba terakhir 0.6, bertujuan agar warna tombol menjadi berubah saat kursor diarahkan kesana.

1. Menambahkan foto kecil pada bagian terakhir

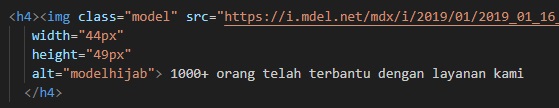


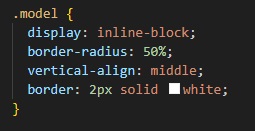
Foto kecil ini merepresentasikan orang-orang yang sudah terbantu (seolah-olah) dengan jasa yang ada pada halaman ini. Saya menggunakan <h4> untuk tulisan yang lebih kecil dari yang lainnya, lalu seperti biasa tag <img src=””> untuk memasukkan gambar, disini saya mengambil gambar dari google sehingga bagian “src” tersebut saya isi dengan link google.

Properti *class=”model”* bertujuan untuk kepentingan modifikasi di CSS nanti. Width sebesar 44 pixels dan height sebesar 49 pixels menyesuaikan dengan ukuran dari konten <h4>, sehingga terlihat seperti ini:



Dan properti “alt” bertujuan sebagai tulisan alternatif yang muncul, apabila gambar tidak bisa atau gagal dimuat nanti.

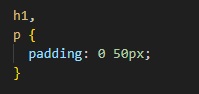
1. Modifikasi model pada CSS



Styling “display: inline-block;” adalah agar foto model ini sejajar dengan konten lain. Sementara styling “border-radius: 50%;” adalah agar bagian tepi dari foto menjadi seperti lingkaran, karena secara *default* foto tersebut berbentuk persegi. Kemudian, styling “vertical-align: middle;” bertujuan agar kedua foto ini berada di tengah-tengah dari item “inline-block” tadi.

Terakhir, styling “border: 2px solid white;” adalah garis tepian dari foto model, yang berarti setebal 2 pixels solid, dan berwarna putih.

1. Mengatur padding konten



Setelah seluruh konten sudah selesai, saya melihat bahwa masih ada yang kurang rapi, yaitu bagian tulisan-tulisan di <h1> dan <p>. Sehingga, saya mengatur paddingnya menjadi tidak ada secara vertikal atau atas-bawah, dan menjadi 50 pixels secara kiri-kanan. Ini membuat konten memiliki jarak yang pas terhadap elemen-elemen lain di halaman.