WEB PROGRAMMING

MODUL PRAKTIKUM

DAFTAR ISI

[1 HTML & Git 3](#_Toc49352681)

[1.1 HTML DOCUMENT STRUCTURE 3](#_Toc49352682)

[1.1.1 CODE EDITOR 3](#_Toc49352683)

[1.1.2 BODY 4](#_Toc49352684)

[1.2 TEXT 6](#_Toc49352685)

[1.2.1 HEADINGS 6](#_Toc49352686)

[1.2.2 PARAGRAPH 7](#_Toc49352687)

[1.2.3 BOLD, ITALIC, SUPERSCRIPT, SUBSCRIPT, 8](#_Toc49352688)

[1.2.4 WHITE SPACE, LINE BREAK, HORIZONTAL RULE 9](#_Toc49352689)

[1.3 LIST 10](#_Toc49352690)

[1.3.1 ORDERED LIST 10](#_Toc49352691)

[1.3.2 UNORDERED LIST 11](#_Toc49352692)

[1.3.3 NESTED LIST 12](#_Toc49352693)

[1.4 LINK 13](#_Toc49352694)

[1.4.1 PENDAHULUAN 13](#_Toc49352695)

[1.4.2 TAUTAN KE WEB LAIN 14](#_Toc49352696)

[1.4.3 TAUTAN KE HALAMAN LAIN DI WEB YANG SAMA 15](#_Toc49352697)

[1.4.4 STRUKTUR DIREKTORI 16](#_Toc49352698)

[1.4.5 RELATIVE URLS 17](#_Toc49352699)

[1.4.6 TAUTAN EMAIL 18](#_Toc49352700)

[1.4.7 MEMBUKA TAUTAN DI JENDELA BARU 19](#_Toc49352701)

[1.4.8 TAUTAN KE LOKASI SPESIFIK DI HALAMAN YANG SAMA 20](#_Toc49352702)

[1.5 GAMBAR 21](#_Toc49352703)

[1.5.1 Image 22](#_Toc49352704)

[1.5.2 WIDTH & HEIGHT 22](#_Toc49352705)

[1.5.3 FIGURE 23](#_Toc49352706)

[1.6 TABLE 24](#_Toc49352707)

[1.6.1 TR, TD 24](#_Toc49352708)

[1.6.2 TH 25](#_Toc49352709)

[1.6.3 COLUMN SPAN 26](#_Toc49352710)

[1.6.4 ROW SPAN 27](#_Toc49352711)

[1.6.5 LONG TABLE 28](#_Toc49352712)

[1.7 FORM 29](#_Toc49352713)

[1.7.1 ACTION 29](#_Toc49352714)

[1.7.2 TEXT INPUT, PASSWORD INPUT 30](#_Toc49352715)

[1.7.3 TEXT AREA 31](#_Toc49352716)

[1.7.4 RADIO BUTTON, CHECK BOX 32](#_Toc49352717)

[1.7.5 DROP DOWN LIST BOX 33](#_Toc49352718)

[1.7.6 MULTIPLE SELECT BOX 34](#_Toc49352719)

[1.7.7 FILE INPUT BOX 35](#_Toc49352720)

[1.7.8 SUBMIT BUTTON 36](#_Toc49352721)

[1.7.9 IMAGE BUTTON 37](#_Toc49352722)

[1.7.10 HIDDEN FIELD 38](#_Toc49352723)

[1.7.11 LABEL 39](#_Toc49352724)

[1.7.12 GROUPING FORM 40](#_Toc49352725)

[1.7.13 FORM VALIDATION 41](#_Toc49352726)

[1.7.14 DATE, EMAIL, URL INPUT 42](#_Toc49352727)

[1.8 EXTRA MARKUP 43](#_Toc49352728)

[1.8.1 DOCTYPE 43](#_Toc49352729)

[1.8.2 COMMENT 44](#_Toc49352730)

[1.8.3 DIV 45](#_Toc49352731)

[1.8.4 SPAN 46](#_Toc49352732)

[1.8.5 IFRAME 47](#_Toc49352733)

[1.8.6 META 48](#_Toc49352734)

# HTML & Git

## HTML DOCUMENT STRUCTURE

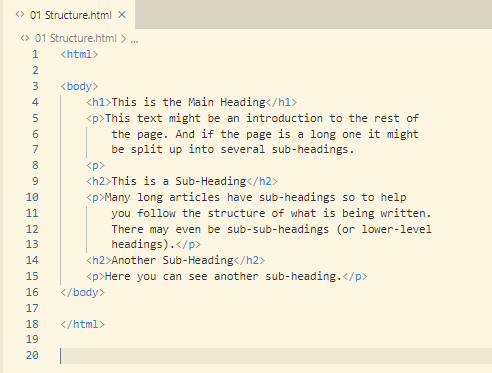
### CODE EDITOR

Untuk menggambarkan struktur halaman web, Anda dapat melihat kode HTML di bawah ini. Gunakan Code Editor apa saja, misalnya Notepad, Notepad++, Visual Studio Code, dll. Lalu simpan dengan ektensi (dot) html.

Referensi

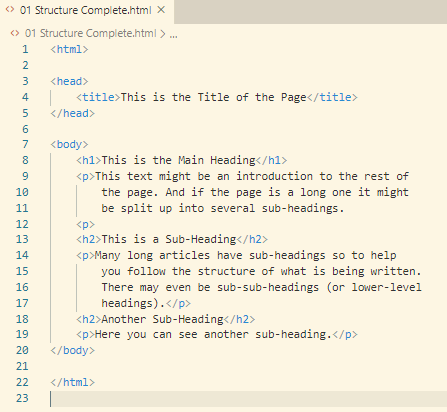
* Visual Studio Code unduh dari <https://code.visualstudio.com/>
* Notepad++ unduh dari <https://notepad-plus-plus.org/>
* Programmer’s Notepad unduh dari <https://www.pnotepad.org/>

### BODY

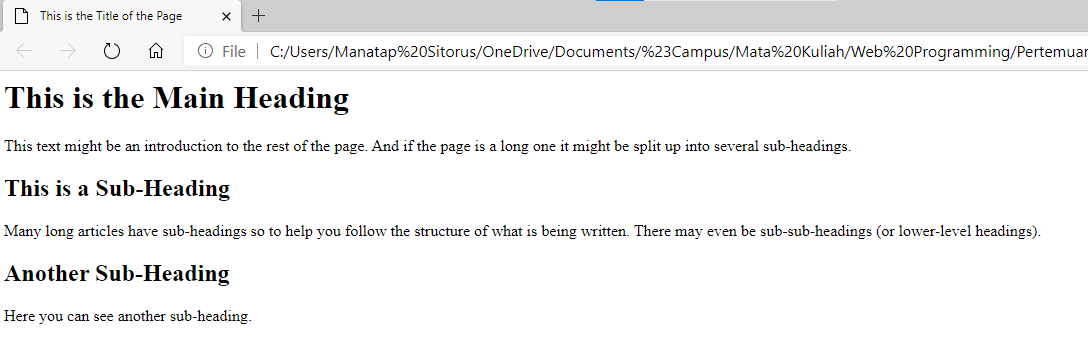


Kode HTML terdiri dari karakter yang berada di dalam tanda kurung siku - disebut sebagai elemen HTML. Elemen biasanya terdiri dari dua tag: tag pembuka dan tag penutup. Tag penutup memiliki garis miring di dalamnya. Setiap elemen HTML memberi tahu browser sesuatu tentang informasi yang berada di antara tag pembuka dan penutupnya.

* Tag pembuka <html> menunjukkan bahwa apa pun di antara tag itu dan tag </html> penutup adalah kode HTML.
* Tag <body> menunjukkan bahwa ada sesuatu di antara itu dan penutup. Tag </body> harus ditampilkan di dalam jendela browser utama.
* Kata-kata antara <h1> dan </h1> adalah judul utama.
* Paragraf teks muncul di antara tag <p> dan </p>.
* Kata-kata antara <h2> dan </h2> membentuk subtitle.



Tag HTML yang lebih lengkap. Tag HTML <head> berisi informasi tentang halaman, biasanya Anda akan menemukan elemen <title> di dalam elemen <head>. Isi elemen <title> ditampilkan di bagian atas browser, di atas di mana Anda biasanya mengetikkan URL halaman yang ingin Anda kunjungi, atau pada tab untuk halaman itu (jika browser Anda menggunakan tab untuk memungkinkan Anda untuk melihat beberapa halaman secara bersamaan). Tampilan kode HTML yang dijalankan di Browser dapat dilihat di bawah ini.

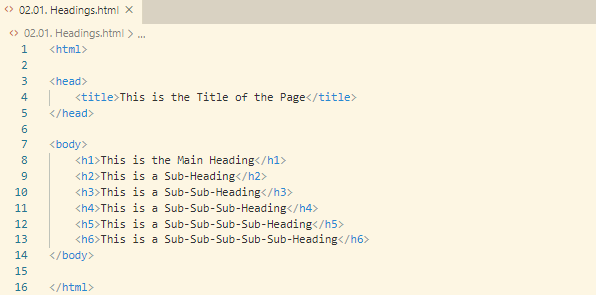


## TEXT

### HEADINGS

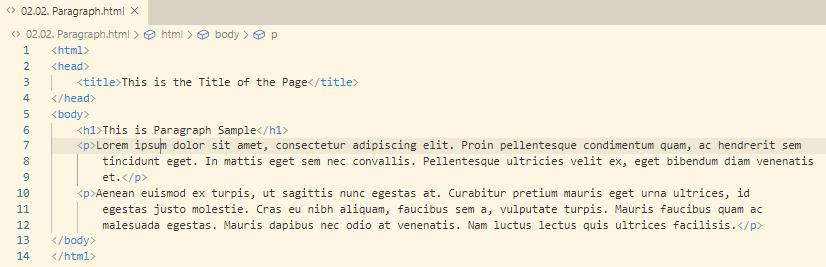
Peramban (Browser) menampilkan isi judul pada berbagai ukuran. Konten elemen <h1> adalah yang terbesar, dan konten elemen <h6> adalah yang terkecil.

Ukuran persis di mana setiap browser menunjukkan judul dapat sedikit berbeda. Pengguna juga dapat menyesuaikan ukuran teks di browser mereka. Anda akan melihat bagaimana mengontrol ukuran teks, warnanya, dan font yang digunakan ketika kita melihat CSS



### PARAGRAPH

Untuk membuat paragraf, kelilingi kata-kata yang membentuk paragraf dengan tag <p> pembuka dan tag </p> penutup. Secara default, browser akan menampilkan setiap paragraf pada baris baru dengan sedikit ruang di antara paragraf tersebut dan paragraf berikutnya.





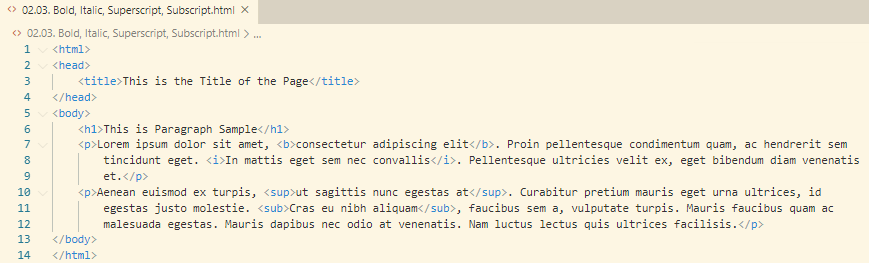
### BOLD, ITALIC, SUPERSCRIPT, SUBSCRIPT,

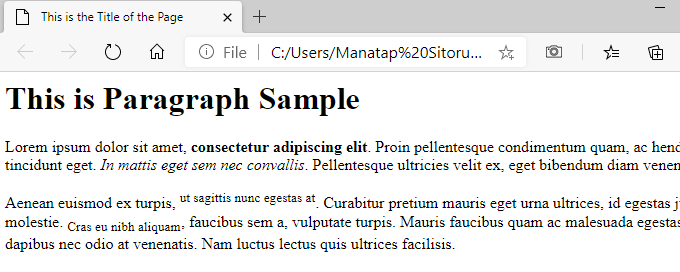
Dengan melampirkan kata-kata pada tag <b> dan </b> kita dapat membuat karakter tampak tebal. Elemen <b> juga mewakili bagian teks yang akan disajikan dengan cara yang berbeda secara visual, misalnya kata kunci dalam paragraf.

Dengan melampirkan kata dalam tag <i> dan </i> kita dapat membuat karakter tampak miring. Elemen <i> juga mewakili bagian teks yang akan diucapkan dengan cara yang berbeda dari konten di sekitarnya - seperti istilah teknis, nama kapal, kata-kata asing, pemikiran, atau istilah lain yang biasanya dicetak miring.

Elemen <sup> digunakan untuk memuat karakter yang harus superskrip seperti sufiks tanggal atau konsep matematika seperti menaikkan angka menjadi pangkat seperti 22.

Elemen <sub> digunakan untuk memuat karakter-karakter yang harus berupa subskrip. Ini biasanya digunakan dengan catatan kaki atau formula kimia seperti H20.



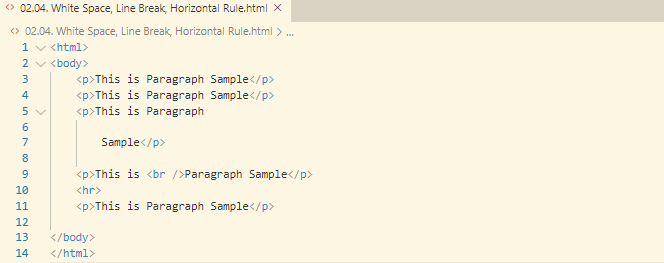


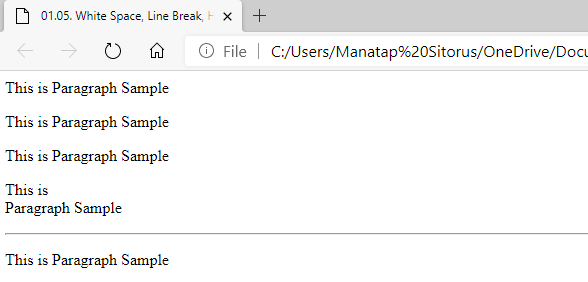
### WHITE SPACE, LINE BREAK, HORIZONTAL RULE

Agar kode lebih mudah dibaca, penulis halaman web sering menambahkan ruang ekstra atau memulai beberapa elemen pada baris baru. Ketika browser menemukan dua spasi atau lebih di sebelah satu sama lain, browser hanya menampilkan satu spasi, demikian pula jika berupa new line. Ini dikenal sebagai collapsed white space. Spasi ini kadang digunakan untuk mempermudah pembacaan source code.

Seperti tag <p> sebelumnya, browser akan secara otomatis menampilkan setiap paragraf baru atau menuju baris baru. Tetapi jika Anda ingin menambahkan satu baris di tengah paragraf, Anda bisa menggunakan tag baris tersebut <br />.

Untuk membuat jeda antara tema, seperti perubahan topik, tambahkan tag <hr />.

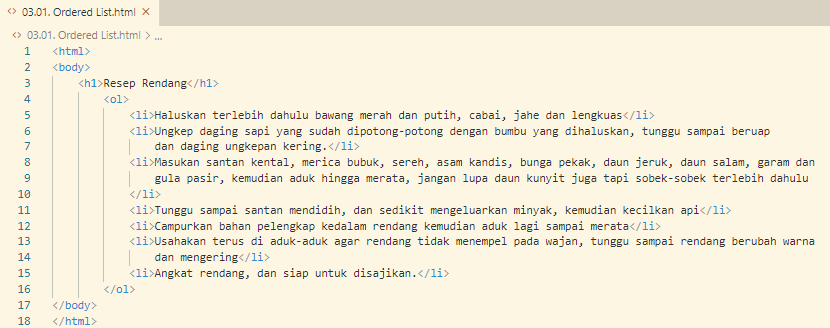


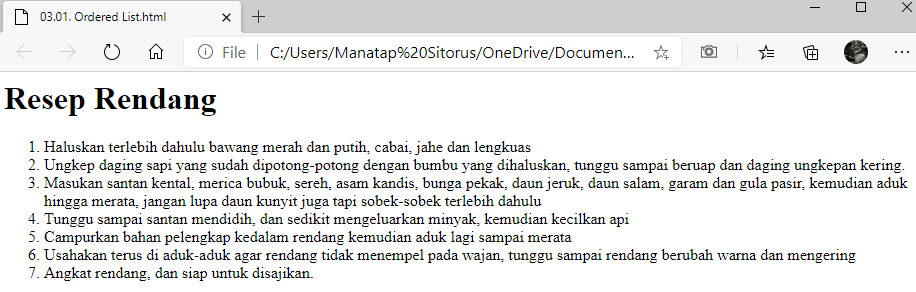


## LIST

### ORDERED LIST

Ordered List adalah daftar yang diberi nomor. Sebagai contoh, daftar tersebut dapat berupa serangkaian langkah-langkah untuk resep yang harus dilakukan secara berurutan.

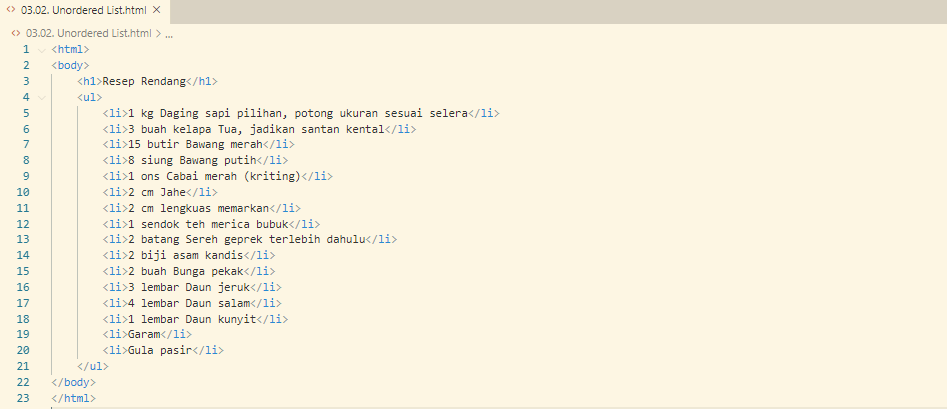


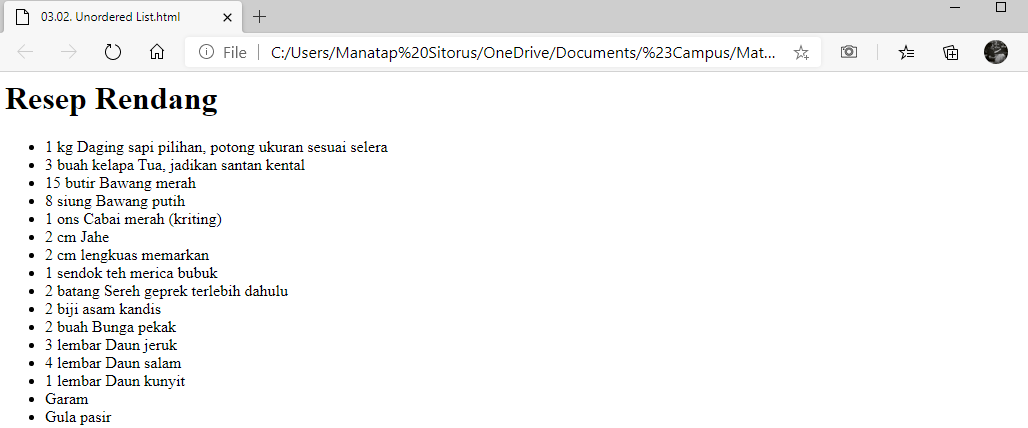


Daftar yang diurutkan dibuat dengan elemen <ol>. Setiap item dalam daftar ditempatkan di antara tag <li> pembuka dan tag </li> penutup. (li adalah singkatan dari List Item.) Browser membuat indent secara default.

### UNORDERED LIST

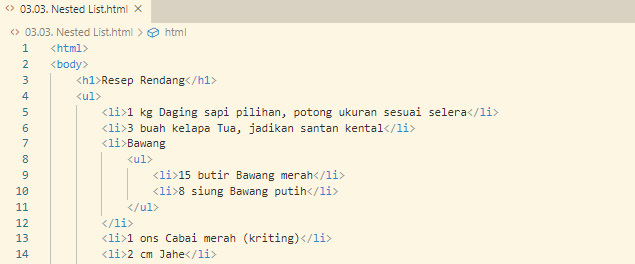
Unordered List adalah daftar yang dimulai dengan bullet poin-poin, bukan karakter yang menunjukkan urutan seperti 1,2,3, dst.

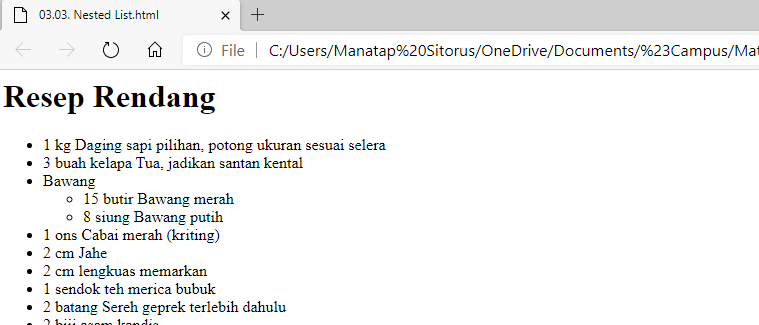




### NESTED LIST

Anda dapat menempatkan daftar kedua di dalam elemen <li> untuk membuat daftar sublist atau daftar bersarang. Peramban menampilkan daftar bersarang yang menjorok lebih dalam dari daftar induk. Dalam nested unordered list, browser biasanya akan mengubah style dari poin-poin juga.





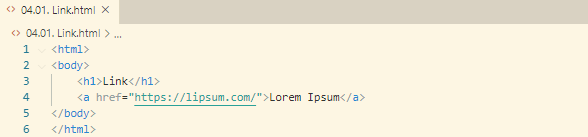
## LINK

### PENDAHULUAN

Link (Tautan) adalah fitur utama web karena memungkinkan Anda untuk berpindah dari satu halaman web ke halaman lainnya. Anda biasanya akan menemukan jenis tautan berikut:

* Tautan dari satu situs web ke situs web lainnya
* Tautan dari satu halaman ke halaman lainnya di situs web yang sama
* Tautan dari satu bagian halaman web ke bagian lain dari halaman yang sama
* Tautan yang terbuka di jendela browser baru
* Tautan yang memulai program email Anda dan alamat email baru kepada seseorang

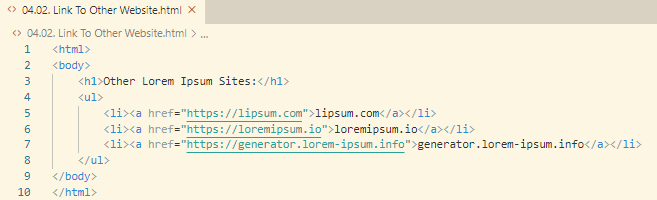
Tautan dibuat menggunakan elemen <a>. Pengguna dapat mengeklik apa saja antara tag <a> dan tag penutup </a> pembuka. Anda menentukan halaman mana yang ingin Anda tautkan menggunakan atribut href.

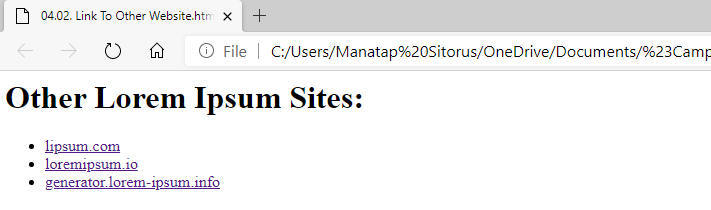


Teks antara tag pembuka <a> dan tag penutup </a> dikenal sebagai teks tautan. Jika memungkinkan, teks tautan Anda harus menjelaskan ke mana pengunjung akan diambil jika mereka mengkliknya (bukan hanya mengatakan "klik di sini"). Untuk menulis teks tautan yang bagus, Anda dapat memikirkan kata-kata yang mungkin digunakan orang saat mencari halaman yang Anda tautkan. (Misalnya, daripada menulis "tempat menginap", Anda dapat menggunakan sesuatu yang lebih spesifik seperti "hotel di Jakarta.")

### TAUTAN KE WEB LAIN

Tautan dibuat menggunakan elemen <a> yang memiliki atribut yang disebut href. Value dari atribut href adalah halaman yang Anda ingin orang kunjungi ketika mereka mengklik tautan. Pengguna dapat mengeklik apa pun yang muncul antara tag <a> dan tag penutup </a> pembuka dan akan dibawa ke halaman yang ditentukan dalam atribut href. Saat Anda menautkan ke situs web yang berbeda, value dari atribut href akan menjadi alamat web lengkap untuk situs tersebut, yang dikenal sebagai Absolute URL. Browser menampilkan tautan berwarna biru dengan garis bawah secara default.

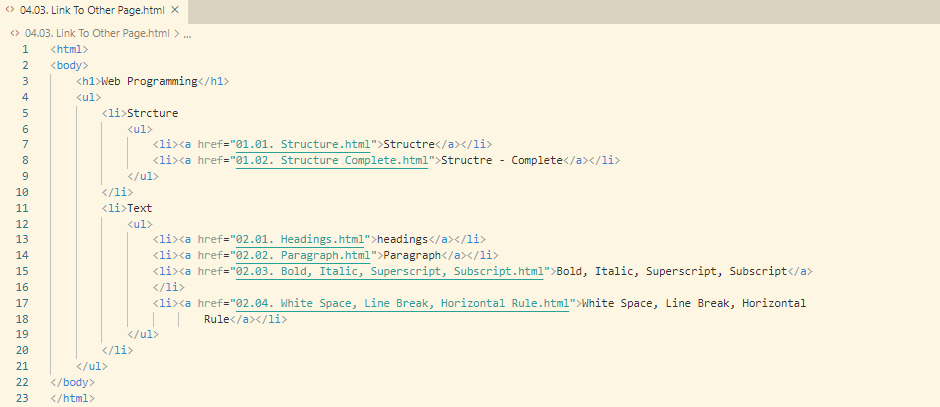


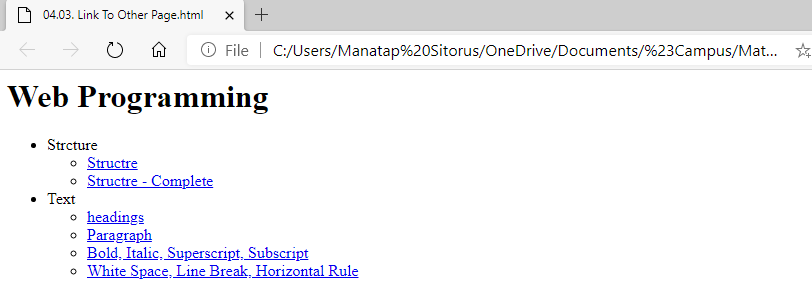


URL adalah singkatan dari Uniform Resource Locator. Setiap halaman web memiliki URL sendiri. Ini adalah alamat web yang akan Anda ketik di browser jika Anda ingin mengunjungi halaman tertentu. URL absolut dimulai dengan nama domain untuk situs itu, dan dapat diikuti oleh jalur ke halaman tertentu. Jika tidak ada halaman yang ditentukan, situs akan menampilkan homepage (index atau home).

### TAUTAN KE HALAMAN LAIN DI WEB YANG SAMA

Saat Anda menautkan ke halaman lain dalam situs yang sama, Anda tidak perlu menentukan nama domain di URL. Anda dapat menggunakan singkatan yang dikenal sebagai URL relatif. Jika semua halaman situs berada di folder yang sama, maka nilai atribut href hanyalah nama file. Jika Anda memiliki halaman situs yang berbeda di folder yang berbeda, maka Anda dapat menggunakan sintaks yang sedikit lebih kompleks untuk menunjukkan di mana halaman tersebut terkait dengan halaman saat ini.

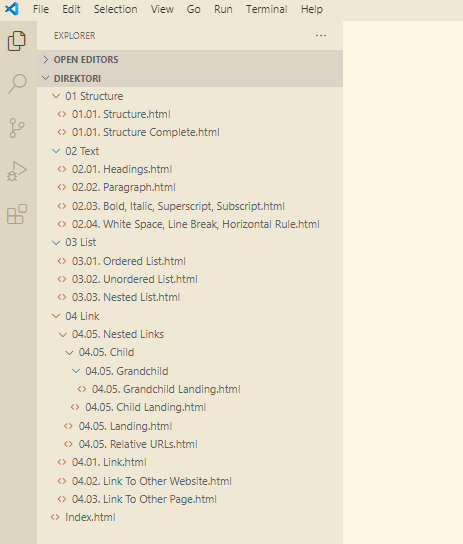




Saat menautkan ke halaman lain dalam situs yang sama, Anda dapat menggunakan URL relatif. Ini seperti versi singkat dari URL absolut karena Anda tidak perlu menentukan nama domain. URL relatif membantu ketika membangun situs di komputer Anda karena Anda dapat membuat tautan antar halaman tanpa harus mengatur nama domain atau hosting Anda.

### STRUKTUR DIREKTORI

Di situs web yang lebih besar sebaiknya halaman web diatur dengan mengkategorikan dan ditempatkan ke dalam folder yang berbeda. Folder di situs web kadang-kadang disebut sebagai direktori.



Direktori Root terdiri dari: Index.html, dan direktori-direktori seperti : 01 Structure, 02 Text, 03 List, 04 Link. Setiap direktori terdiri dari dokumen HTML sesuai dengan pembahasan sebelumnya. Apabila suatu direktori terdiri dari dokumen yang berjumlah sangat banyak, Anda dapat mengkategorikan lagi menjadi sub-direktori yang lebih dalam. Hal ini bertujuan untuk memudahkan navigasi berdasarkan konteks yang lebih jelas.

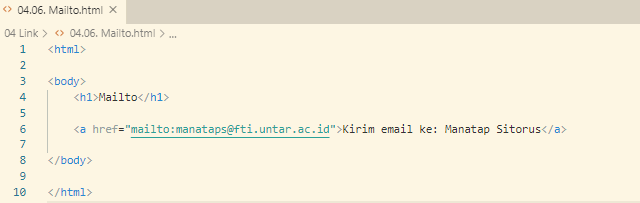
### RELATIVE URLS

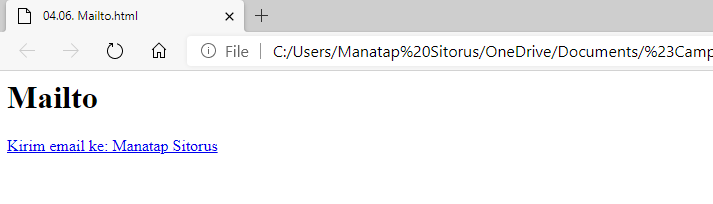
URL relatif dapat digunakan saat menautkan ke halaman dalam situs web Anda sendiri. Mereka menyediakan cara cepat untuk memberi tahu browser tempat menemukan file Anda. Tipe-tipe Relative URL dapat dilihat melalui contoh source di bawah.



### TAUTAN EMAIL

Untuk membuat tautan yang memanggil aplikasi email dan mengalamatkan email tersebut ke alamat email penerima, gunakan elemen <a>. Namun, kali ini nilai atribut href dimulai dengan mailto: dan diikuti oleh alamat email yang Anda inginkan untuk mengirim email. Di sebelah kanan Anda dapat melihat bahwa tautan email terlihat seperti tautan lain, tetapi ketika diklik, program email pengguna akan membuka pesan email baru dan mengatasinya kepada orang yang ditentukan dalam tautan tersebut.





### MEMBUKA TAUTAN DI JENDELA BARU

Jika Anda ingin tautan terbuka di jendela baru, gunakan atribut target pada tag <a> pembuka. Value dari atribut ini harus \_blank.

Salah satu alasan paling umum penulis laman web mungkin menginginkan tautan dibuka di jendela baru adalah jika tautan itu menunjuk ke situs web lain. Dalam kasus seperti itu, mereka berharap pengguna akan kembali ke jendela yang berisi situs mereka setelah selesai melihat yang lain.



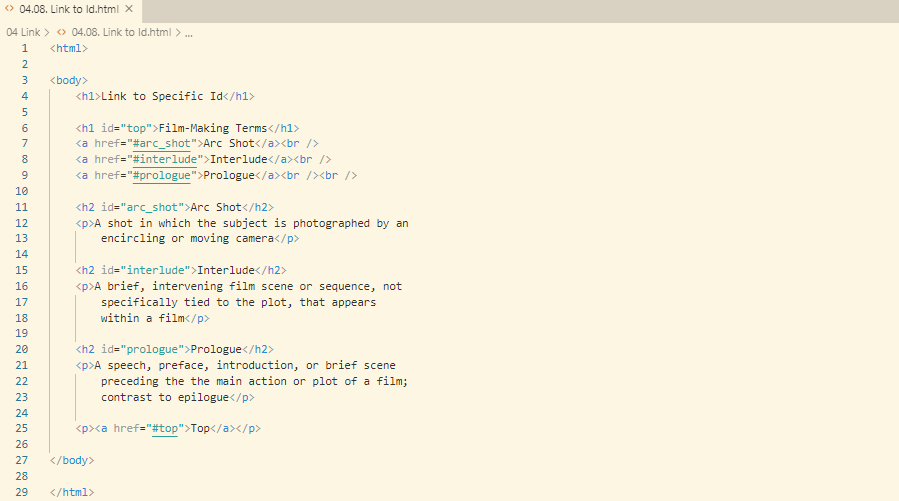
### TAUTAN KE LOKASI SPESIFIK DI HALAMAN YANG SAMA

Di bagian atas halaman yang panjang, Anda mungkin ingin menambahkan daftar konten yang tertaut ke bagian yang sesuai di bagian bawah. Atau Anda mungkin ingin menambahkan tautan dari bagian bawah halaman kembali ke atas untuk menyelamatkan pengguna dari harus gulir kembali ke atas.

Sebelum Anda dapat menautkan ke bagian tertentu dari suatu halaman, Anda perlu mengidentifikasi titik-titik di halaman yang akan dituju. Anda melakukan ini menggunakan atribut id (yang dapat digunakan pada setiap elemen HTML).

Anda dapat melihat bahwa elemen <h1> dan <h2> dalam contoh ini telah diberi atribut id yang mengidentifikasi bagian-bagian halaman tersebut. Nilai atribut id harus dimulai dengan huruf atau garis bawah (bukan angka atau karakter lain) dan, pada satu halaman, tidak ada dua atribut id yang harus memiliki nilai yang sama.

Untuk menautkan ke elemen yang menggunakan atribut id Anda menggunakan elemen <a> lagi, tetapi nilai atribut href dimulai dengan simbol #, diikuti dengan nilai atribut id dari elemen yang ingin Anda tautkan. Dalam contoh ini, <a href="#top"> tautan ke elemen <h1> di bagian atas halaman yang atribut id-nya memiliki nilai atas.



## GAMBAR

Beberapa alasan untuk menambahkan gambar ke halaman web: Anda mungkin ingin menyertakan logo, foto, ilustrasi, diagram, atau bagan.

Seiring dengan pertumbuhan situs web, menyimpan gambar dalam folder terpisah membantu Anda memahami bagaimana situs tersebut dikelola. Di sini Anda dapat melihat contoh file untuk situs web; semua gambar disimpan dalam folder yang disebut gambar. Di situs besar Anda mungkin ingin menambahkan subfolder di dalam folder gambar.

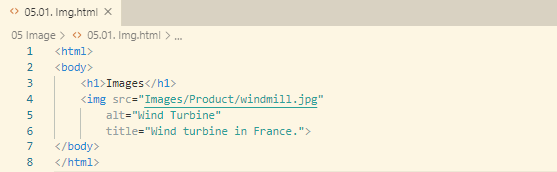
Sebagai contoh, gambar seperti logo dan button boleh disimpan di folder yang disebut Interface, foto produk mungkin ada di halaman yang disebut Product, dan gambar yang terkait dengan berita mungkin ada di folder yang disebut News.

### Image

Untuk menambahkan gambar ke halaman, Anda perlu menggunakan elemen <img>. Ini adalah elemen kosong (yang berarti tidak ada tag penutup). Elemen ini membawa dua atribut berikut:

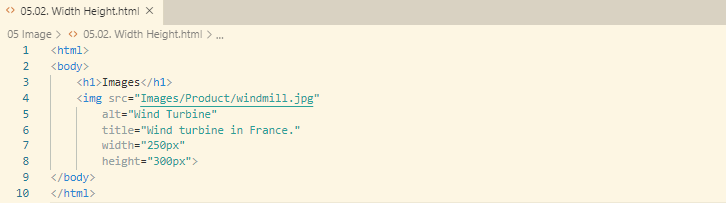
* src Digunakan untuk memberitahu browser di mana dapat menemukan file gambar. Biasanya berupa URL relatif yang menunjuk ke gambar di situs Anda sendiri. (Di sini Anda dapat melihat bahwa gambar-gambar itu ada di folder bernama Images/Product).
* alt Digunakan untuk memberikan deskripsi teks dari gambar yang menggambarkan gambar jika Anda tidak dapat melihatnya. Informasi yang ditulis haruslah deskripsi yang akurat tentang konten gambar, sehingga dapat dipahami oleh perangkat lunak pembaca layar (digunakan oleh orang-orang dengan gangguan penglihatan) dan mesin pencari.
* title Digunakan untuk memberikan informasi tambahan tentang gambar. Sebagian besar browser akan menampilkan konten dari atribut ini dalam sebuah tootip ketika pengguna mengarahkan kursor ke gambar.

Jika gambar hanya untuk membuat halaman terlihat lebih menarik (dan tidak memiliki makna, seperti garis pemisah grafik), maka atribut alt masih harus digunakan tetapi tanda kutip harus dibiarkan kosong.



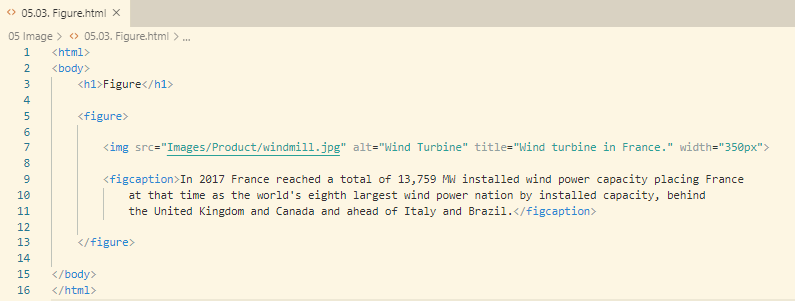
### WIDTH & HEIGHT

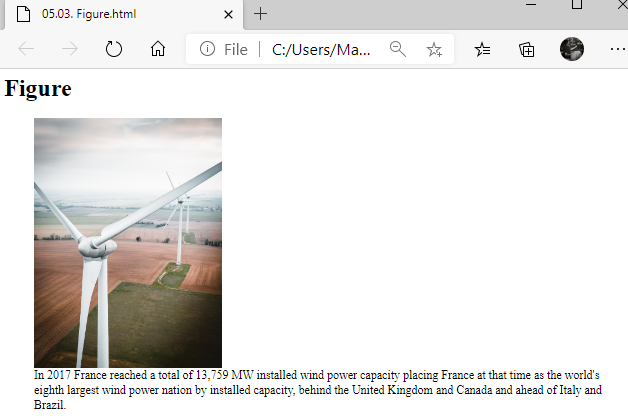
Anda dapat mengatur lebar dan tinggi dari suatu gambar dengan menggunakan opsi seperti di bawah ini. Perhatikan bahwa menggunakan Width dan Height secara bersaman dapat menyebabkan perubahan Aspect Ratio (gambar terkompres secara horisontal atau vertikal).



### FIGURE

Gambar sering kali disertai caption teks. HTML5 telah memperkenalkan elemen <figure> untuk memuat gambar dan keterangannya sehingga keduanya terkait. Anda dapat memiliki lebih dari satu gambar di dalam elemen <figure> selama mereka semua berbagi keterangan yang sama. <figcaption> Elemen <figcaption> telah ditambahkan ke HTML5 untuk memungkinkan penulis halaman web menambahkan keterangan pada gambar.





## TABLE

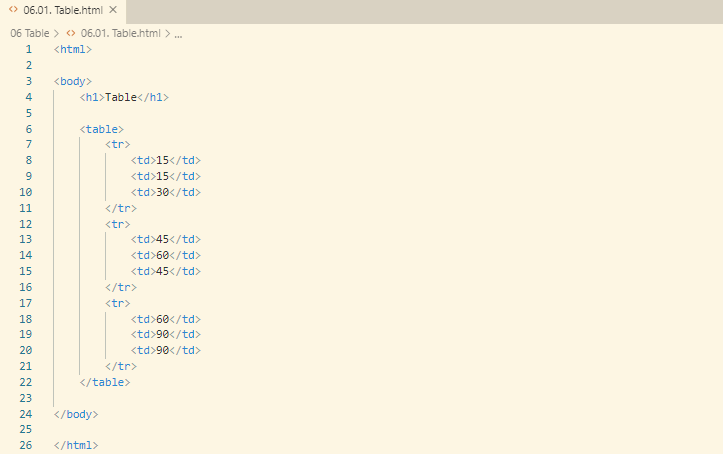
Ada beberapa jenis informasi yang perlu ditampilkan dalam kotak atau tabel. Misalnya: hasil olahraga, laporan stok saham, jadwal kereta. Saat merepresentasikan informasi dalam tabel, Anda perlu berpikir dalam hal grid yang terdiri dari baris dan kolom (sedikit seperti spreadsheet).

### TR, TD

Elemen <table> digunakan untuk membuat tabel. Isi tabel dituliskan baris demi baris.

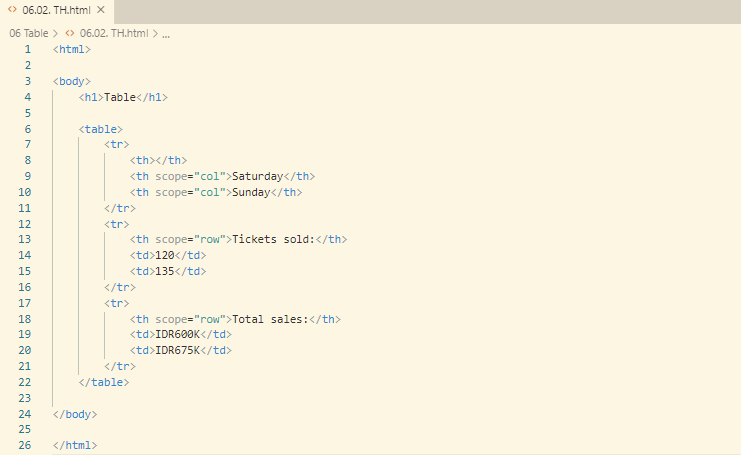
Elemen <tr> mengindikasikan awal dari setiap baris. (TR adalah singkatan dari Table Row.) TR diikuti oleh satu atau lebih elemen <td> (satu untuk setiap sel di baris itu). Di akhir baris Anda menggunakan tag </tr> penutup.

Elemen <td> mewakili setiap sel. (TD adalah singkatan dari Table Data.) Di akhir setiap sel gunakan tag </td> penutup.



### TH

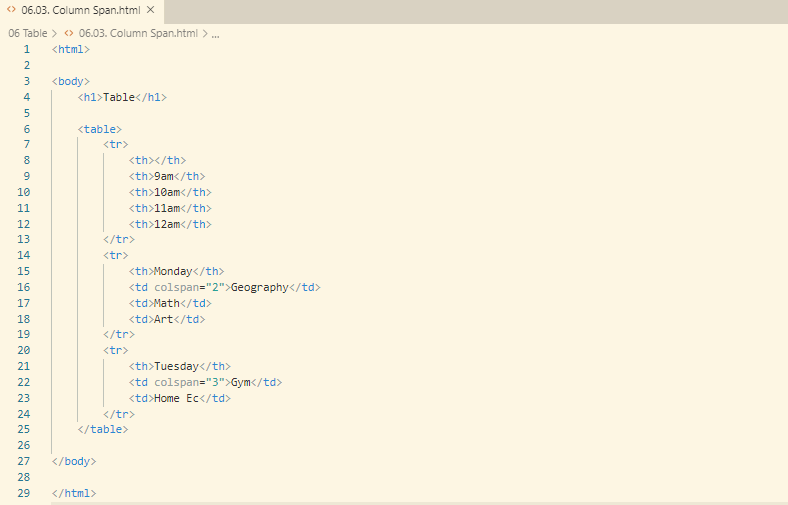
Elemen <th> (singkatan dari Table Heading) digunakan sama seperti elemen <td> tetapi tujuannya adalah untuk mewakili judul kolom atau baris. Bahkan jika sebuah sel tidak memiliki konten, Anda masih harus menggunakan elemen <td> atau <th> untuk mewakili keberadaan sel kosong jika tidak, tabel tidak akan merender dengan benar.



Menggunakan elemen <th> untuk heading membantu orang yang menggunakan pembaca layar, meningkatkan kemampuan mesin pencari untuk mengindeks halaman Anda, dan juga memungkinkan Anda untuk mengontrol tampilan tabel lebih baik saat Anda mulai menggunakan CSS. Anda dapat menggunakan atribut lingkup pada elemen <th> untuk mengindikasikan apakah itu adalah tajuk untuk kolom atau baris.

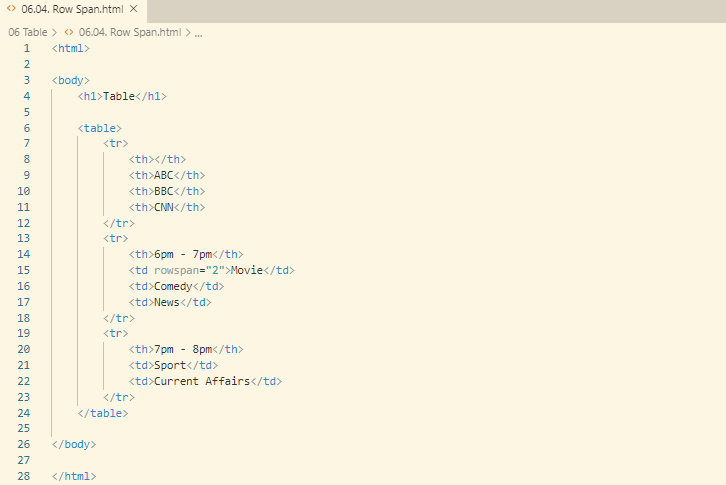
### COLUMN SPAN

Terkadang Anda mungkin membutuhkan entri dalam tabel untuk merentangkan lebih dari satu kolom. Atribut colspan dapat digunakan pada elemen <th> atau <td> dan menunjukkan berapa banyak kolom yang harus dijalankan sel. Dalam contoh di bawah ini, dapat dilihat jadwal dengan lima kolom; kolom pertama berisi tajuk untuk baris itu (hari), empat sisanya mewakili slot waktu satu jam. Jika Anda melihat sel tabel yang berisi kata-kata 'Geografi' Anda akan melihat bahwa nilai atribut colspan adalah 2, yang menunjukkan bahwa sel harus berjalan melintasi dua kolom. Di baris ketiga, 'Gym' berjalan melintasi tiga kolom. Anda dapat melihat bahwa baris kedua dan ketiga memiliki lebih sedikit elemen <td> daripada kolom. Ini karena, ketika sebuah sel memanjang di lebih dari satu kolom, sel-sel <td> atau <th> yang seharusnya berada di tempat sel yang lebih luas tidak termasuk dalam kode.



### ROW SPAN

Anda juga mungkin perlu entri dalam tabel untuk meregangkan lebih dari satu baris. Atribut rowspan dapat digunakan pada elemen <th> atau <td> untuk menunjukkan berapa banyak baris sel harus merentang ke bawah tabel. Dalam contoh di sebelah kiri Anda dapat melihat bahwa ABC menunjukkan film dari pukul 18.00 - 20.00, sedangkan saluran BBC dan CNN sama-sama menampilkan dua program selama periode waktu ini (masing-masing berlangsung selama satu jam). Jika Anda melihat elemen <tr> terakhir, itu hanya berisi tiga elemen meskipun ada empat kolom dalam hasil di bawah ini. Ini karena film dalam elemen <tr> di atasnya menggunakan atribut rowspan untuk meregangkan dan mengambil alih sel di bawah ini.



### LONG TABLE

Ada tiga elemen yang membantu membedakan antara konten utama tabel dan baris pertama dan terakhir (yang dapat berisi konten yang berbeda). Elemen-elemen ini membantu orang-orang yang menggunakan pembaca layar dan juga memungkinkan Anda untuk menata bagian-bagian ini dengan cara yang berbeda dari tabel lainnya (seperti yang akan Anda lihat ketika Anda belajar tentang CSS).

* Elemen <thead> Judul tabel harus berada di dalam elemen <thead>.
* Elemen <tbody> Tubuh harus duduk di dalam elemen <tbody>.
* Elemen <tfoot> Footer berada di dalam elemen <tfoot>.



## FORM

Secara umum, istilah 'formulir' mengacu pada dokumen cetak yang berisi tempat bagi Anda untuk mengisi informasi. HTML meminjam konsep formulir untuk merujuk ke berbagai elemen yang memungkinkan Anda mengumpulkan informasi dari pengunjung ke situs Anda. Apakah Anda menambahkan kotak pencarian sederhana ke situs web Anda atau Anda perlu membuat aplikasi asuransi yang lebih rumit, formulir HTML memberi Anda satu set elemen untuk mengumpulkan data dari pengguna Anda.

### ACTION

Elemen Form ini harus selalu membawa atribut Action dan biasanya akan memiliki atribut Method dan Id.

Setiap elemen <form> membutuhkan atribut action. Valuenya adalah URL untuk halaman di server yang akan menerima informasi dalam bentuk ketika dikirimkan.

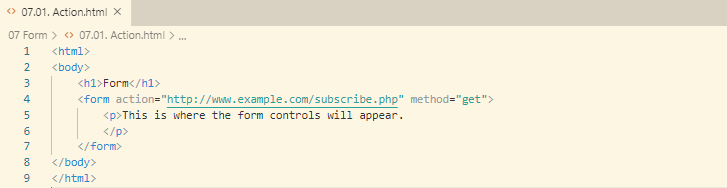
Form Method dapat menggunakan salah satu dari dua metode: GET atau POST. Dengan metode Get, nilai-nilai dari formulir ditambahkan ke akhir URL yang ditentukan dalam atribut action. Metode get sangat ideal untuk:

* Formulir pendek (seperti kotak pencarian)
* Ketika Anda baru saja mengambil data dari server web (tidak mengirim informasi yang harus ditambahkan atau dihapus dari database)

Dengan metode Post, value dikirim dalam apa yang dikenal sebagai header HTTP. Sebagai aturan praktis Anda harus menggunakan metode posting jika formulir Anda:

* memungkinkan pengguna untuk mengunggah file
* sangat panjang
* berisi data sensitif (mis. Kata sandi)
* menambahkan informasi ke, atau menghapus informasi dari, basis data

Jika atribut metode tidak digunakan, data formulir akan dikirim menggunakan metode get.

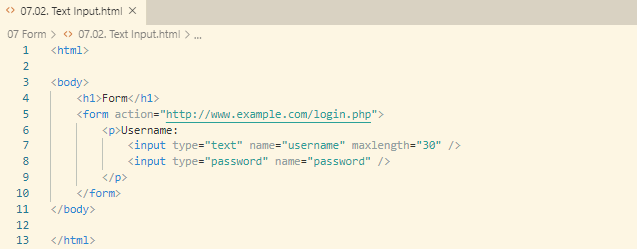


### TEXT INPUT, PASSWORD INPUT

Elemen <input> digunakan untuk membuat beberapa form control yang berbeda. Value dari atribut Type menentukan tipe input apa yang akan dibuat. Type = "text" memiliki value berupa string teks singleline.

Ketika pengguna memasukkan informasi ke dalam formulir, server harus nama dari kontrol yang digunakan. (Misalnya, dalam formulir login, server perlu tahu apa yang telah dimasukkan sebagai nama pengguna dan apa yang telah diberikan sebagai kata sandi.) Oleh karena itu, setiap kontrol bentuk memerlukan atribut Name. Value dari atribut ini mengidentifikasi form control dan dikirim bersama dengan informasi yang mereka masukkan ke server.

Anda dapat menggunakan atribut MaxLength untuk membatasi jumlah karakter yang dapat dimasukkan pengguna ke dalam bidang teks. Valuenya adalah jumlah karakter yang dapat mereka masukkan. Misalnya, jika Anda meminta satu tahun, atribut maxlength dapat memiliki nilai 4 (empat karakter yg mewakili penulisan tahun).



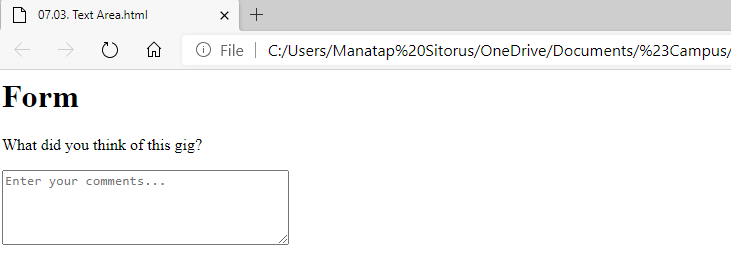
Ketika atribut type memiliki value Password, maka kotak teks yang bertindak seperti input teks satu baris, kecuali karakter yang diblokir. Teks disembunyikan dengan cara ini sehingga jika seseorang melihat dari balik bahu pengguna, mereka tidak dapat melihat data sensitif seperti kata sandi.

Meskipun kata sandi disembunyikan di layar, ini tidak berarti bahwa data dalam kontrol kata sandi dikirim dengan aman ke server. Anda tidak boleh menggunakannya untuk mengirim data sensitif seperti nomor kartu kredit. Untuk keamanan penuh, server perlu diatur untuk berkomunikasi dengan browser pengguna menggunakan Secure Sockets Layer (SSL).

### TEXT AREA

Elemen <textarea> digunakan untuk membuat input teks mutli-line. Tidak seperti elemen input lainnya, ini bukan elemen kosong. Karena itu harus memiliki tag pembuka dan penutup. Teks apa pun yang muncul antara tag pembuka <textarea> dan penutup </textarea> akan muncul di kotak teks saat halaman dimuat. Jika pengguna tidak menghapus teks di antara tag-tag ini, pesan ini akan dikirim ke server bersama dengan apa pun yang diketik pengguna. (Beberapa situs menggunakan JavaScript untuk menghapus informasi ini ketika pengguna mengklik di area teks.)





Atribut Placeholder digunakan untuk membantu pengguna untuk memberika informasi apa yang dibutuhkan oleh form control ini.

### RADIO BUTTON, CHECK BOX

Radio Button memungkinkan pengguna untuk memilih hanya satu dari sejumlah opsi.

* Atribut Name dikirim ke server dengan nilai opsi yang dipilih pengguna. Ketika sebuah pertanyaan memberikan opsi jawaban kepada pengguna dalam bentuk tombol radio, nilai atribut nama harus sama untuk semua tombol radio yang digunakan untuk menjawab pertanyaan itu.
* Atribut Value menunjukkan nilai yang dikirim ke server untuk opsi yang dipilih. Value masing-masing tombol dalam grup harus berbeda (sehingga server tahu opsi mana yang telah dipilih pengguna).
* Atribut Checked dapat digunakan untuk menunjukkan nilai mana (jika ada) yang harus dipilih saat halaman dimuat. Hanya satu tombol radio dalam grup yang dapat menggunakan atribut ini.



Check Bos memungkinkan pengguna untuk memilih (dan membatalkan pilihan) satu atau lebih opsi dalam menjawab pertanyaan.

* Atribut Name dikirim ke server dengan nilai opsi yang dipilih pengguna. Ketika sebuah pertanyaan memberikan opsi jawaban kepada pengguna dalam bentuk kotak centang, nilai atribut Name harus sama untuk semua tombol yang menjawab pertanyaan itu.
* Atribut Value menunjukkan nilai yang dikirim ke server jika kotak centang ini dicentang.
* Atribut Checked menunjukkan bahwa kotak ini dipilih oleh pengguna.

### DROP DOWN LIST BOX

Drop Down List Box memungkinkan pengguna untuk memilih satu opsi dari daftar.

Elemen <select> digunakan untuk membuat Drop Down List Box. Elemen ini mengandung dua atau lebih elemen <option>.

Atribut Name menunjukkan nama form control yang dikirim ke server, bersama dengan nilai yang dipilih pengguna.

Elemen <option> digunakan untuk menentukan opsi yang bisa dipilih pengguna. Kata-kata antara tag pembuka <option> dan penutup </option> akan ditampilkan kepada pengguna di kotak drop down.

Elemen <option> menggunakan atribut Value untuk menunjukkan value yang dikirim ke server bersama dengan form control Name jika opsi ini dipilih

Atribut Checked dapat digunakan untuk menunjukkan opsi yang harus dipilih ketika halaman dimuat. Jika atribut ini tidak digunakan, opsi pertama akan ditampilkan ketika halaman dimuat. Jika pengguna tidak memilih opsi, maka item pertama akan dikirim ke server sebagai nilai untuk kontrol ini.



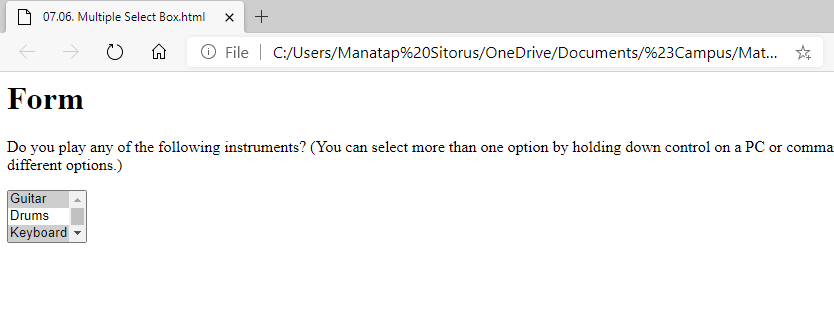
Fungsi Drop Down List Box mirip dengan Radio Button (hanya satu opsi yang dapat dipilih). Ada dua faktor utama dalam memilih mana yang akan digunakan: 1. Jika pengguna perlu melihat semua opsi secara sekilas, tombol radio lebih cocok. 2. Jika ada daftar opsi yang sangat panjang (seperti daftar negara), kotak daftar turun berfungsi lebih baik.

### MULTIPLE SELECT BOX

Anda dapat mengizinkan pengguna untuk memilih beberapa opsi dari daftar ini dengan menambahkan atribut Multiple dengan nilai “multiple”. Beri tahu pengguna jika mereka dapat memilih lebih dari satu opsi.

Juga bermanfaat untuk menunjukkan bahwa pada PC mereka harus menahan tombol Ctrl sambil memilih beberapa opsi dan pada Mac mereka harus menggunakan tombol Command sambil memilih opsi.





### FILE INPUT BOX

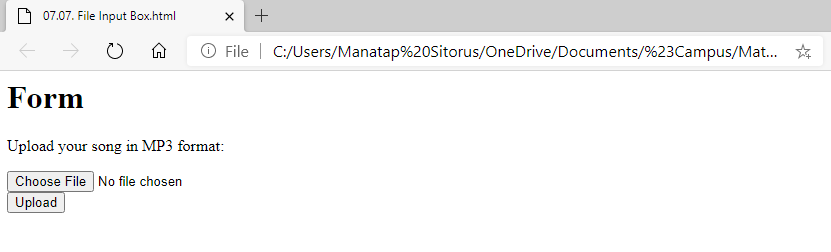
Jika Anda mengizinkan pengguna untuk mengunggah file (misalnya gambar, video, mp3, atau PDF), Anda harus menggunakan kotak input file (Type=”file”).

Jenis input ini membuat sebuah kotak yang tampak seperti input teks diikuti oleh tombol Browse. Ketika pengguna mengklik tombol browse, sebuah jendela terbuka yang memungkinkan mereka untuk memilih file dari komputer mereka untuk diunggah ke situs web.

Saat Anda mengizinkan pengguna untuk mengunggah file, atribut metode pada elemen <form> harus memiliki nilai Post. (Anda tidak dapat mengirim file menggunakan metode get HTTP).

Ketika pengguna mengklik tombol browse, presentasi dari jendela yang memungkinkan mereka untuk mencari file yang ingin mereka unggah akan cocok dengan jendela sistem operasi pengguna. Anda tidak dapat mengontrol tampilan windows ini.



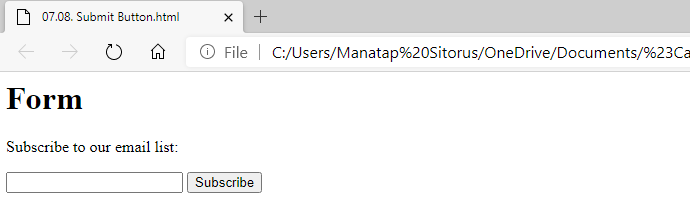


### SUBMIT BUTTON

Tombol Submit digunakan untuk mengirim form ke server. Kontrol ini dapat menggunakan atribut Name tetapi tidak perlu memilikinya.

Atribut value digunakan untuk mengontrol teks yang muncul pada tombol. Sebaiknya tentukan kata-kata yang ingin Anda tampilkan di tombol karena nilai default tombol di beberapa browser adalah ‘Submit’ dan ini mungkin tidak sesuai untuk semua jenis formulir.

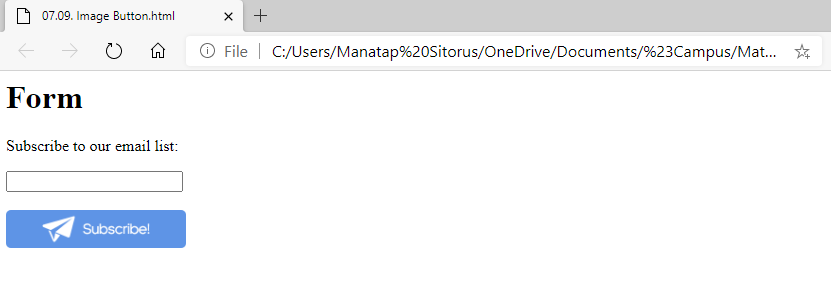




### IMAGE BUTTON

Jika Anda ingin menggunakan gambar untuk tombol Submit, Anda bisa memberi atribut Type dengan value Image. Atribut src, width, height, dan alt bekerja persis seperti elemen <img>.





### HIDDEN FIELD

<input> type = "hidden" adalah salah satu bentuk kontrol tersembunyi. Form control ini tidak ditampilkan pada halaman (meskipun Anda bisa melihatnya jika Anda menggunakan opsi View Source di browser).

Hal ini memungkinkan pembuat halaman web untuk menambahkan value ke form yang tidak bisa dilihat pengguna. Misalnya, penulis halaman web mungkin menggunakan field tersembunyi untuk menunjukkan halaman mana yang pengguna gunakan ketika mereka mengirimkan form.



### LABEL

Saat menggunakan form control, gunakan Label untuk menunjukkan fungsi masing-masing. Namun, setiap form control harus memiliki elemen <label> sendiri supaya form control dapat diakses oleh pengguna dengan gangguan penglihatan.

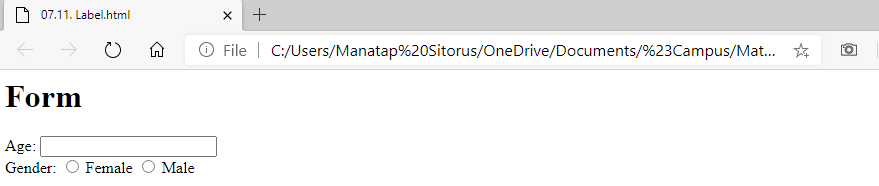
Elemen <label> dapat digunakan dalam dua cara.

* Membungkus deskripsi teks dan input formulir (seperti yang ditunjukkan pada baris pertama dari contoh di sebelah kanan Anda).
* Simpan terpisah dari kontrol formulir dan gunakan atribut for untuk menunjukkan kontrol bentuk mana yang merupakan label untuk (seperti yang ditunjukkan dengan tombol radio).

Status atribut For mereferensi sebuah label ke sebuah form control. Perhatikan bagaimana Radio Button menggunakan atribut Id. Value atribut Id secara unik mengidentifikasi elemen dari semua elemen lain pada halaman.

Value atribut For cocok dengan atribut Id pada form control yang diberi Label. Saat elemen <label> digunakan dengan Check Box atau Radio Button, pengguna dapat mengklik form control atau Label untuk memilih. Area yang dapat diklik diperluas membuat form lebih mudah digunakan. Posisi Label sangat penting, hal ini diperlukan untuk mempermudah pengguna dalam pengoperasian form.



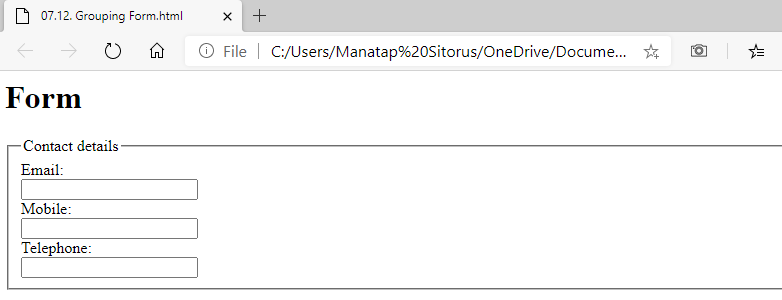


### GROUPING FORM

Anda bisa mengelompokkan form control terkait bersama-sama di dalam elemen <fieldset>. Elemen ini sangat membantu untuk formulir yang lebih panjang. Sebagian besar browser akan menampilkan fieldset dengan garis di sekitar tepi untuk menunjukkan bagaimana mereka terkait. Tampilan garis-garis ini dapat disesuaikan menggunakan CSS.

Elemen <legend> digunakan langsung setelah tag <fieldset> pembuka dan berisi teks yang membantu mengidentifikasi tujuan grup kontrol bentuk itu.





### FORM VALIDATION

Anda mungkin telah melihat formulir di web yang memberikan pesan kepada pengguna jika form control belum diisi dengan benar; ini dikenal sebagai form validation. Secara umum, form validation telah dilakukan menggunakan JavaScript (yang berada di luar cakupan buku ini). Tetapi HTML5 memperkenalkan validasi dan menyerahkan pekerjaan ke browser.

Validasi membantu memastikan pengguna memasukkan informasi dalam bentuk yang dapat dipahami server ketika form dikirimkan. Memvalidasi konten form sebelum dikirim ke server. Form validation ini bertujuan untuk:

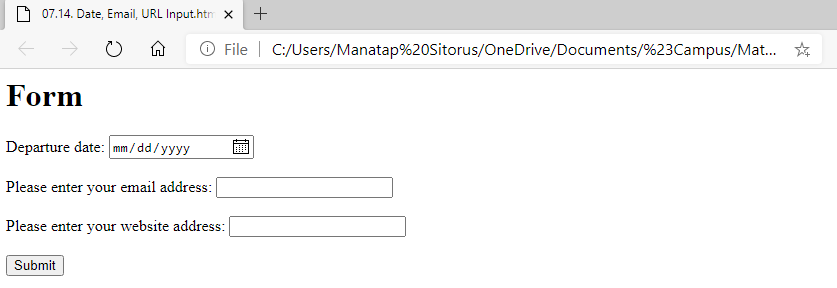
* Mengurangi jumlah pekerjaan yang harus dilakukan server
* Memungkinkan pengguna untuk melihat apakah ada masalah dengan form lebih cepat daripada jika validasi dilakukan di server.



### DATE, EMAIL, URL INPUT

Banyak form perlu mengumpulkan informasi seperti tanggal, alamat email, dan URL. Ini secara tradisional telah dilakukan dengan menggunakan input teks. HTML5 memperkenalkan form control baru untuk membakukan cara beberapa informasi dikumpulkan. Browser lama yang tidak mengenali input ini hanya akan memperlakukan mereka sebagai satu kotak teks baris.





## EXTRA MARKUP

### DOCTYPE

Karena ada beberapa versi HTML, setiap halaman web harus dimulai dengan deklarasi DOCTYPE untuk memberi tahu browser versi HTML mana yang digunakan (walaupun browser biasanya menampilkan halaman meskipun itu tidak termasuk). Penggunaan DOCTYPE juga dapat membantu browser untuk merender halaman dengan benar. Karena XHTML ditulis dalam XML, Anda kadang-kadang akan melihat halaman yang menggunakan DOCTYPE ketat XHTML mulai dengan deklarasi XML opsional. Di mana ini digunakan, itu harus menjadi hal pertama dalam sebuah dokumen.

### COMMENT

Jika Anda ingin menambahkan komentar ke kode Anda yang tidak akan terlihat di browser pengguna, Anda dapat menambahkan teks di antara karakter-karakter ini:

<! - komentar ada di sini ->

Ini digunakan untuk menambahkan komentar ke kode Anda karena, ketika Anda kembali lagi nanti (atau jika orang lain perlu melihat kode), komentar akan membuatnya lebih mudah untuk dipahami. Meskipun komentar tidak terlihat oleh pengguna di jendela browser utama, mereka dapat dilihat oleh siapa saja yang melihat kode sumber di belakang halaman.

Pada halaman yang panjang Anda akan sering melihat komentar yang digunakan untuk menunjukkan di mana bagian halaman mulai atau berakhir, dan untuk menyampaikan catatan untuk membantu siapa saja yang melihat kode memahaminya. Komentar juga dapat digunakan di sekitar blok kode untuk menghentikan kode agar tidak ditampilkan di browser. Pada contoh di sebelah kiri, tautan email telah dikomentari.



### DIV

Elemen <div> digunakan untuk mengelompokkan satu set elemen bersama dalam satu kotak tingkat blok. Misalnya, Anda dapat membuat elemen <div> untuk berisi semua elemen untuk header situs Anda (logo dan navigasi), atau Anda dapat membuat elemen <div> untuk berisi komentar dari pengunjung.

Di browser, konten elemen <div> akan dimulai pada baris baru, tetapi selain itu tidak akan ada perbedaan dengan penyajian halaman. Namun, menggunakan atribut Id atau Class pada elemen <div>, berarti Anda dapat membuat aturan gaya CSS untuk menunjukkan berapa banyak ruang yang harus ditempati elemen <div> di layar dan mengubah tampilan semua elemen yang terkandung di dalamnya. Itu juga dapat membuatnya lebih mudah untuk mengikuti kode Anda jika Anda telah menggunakan elemen <div> untuk menahan setiap bagian halaman.



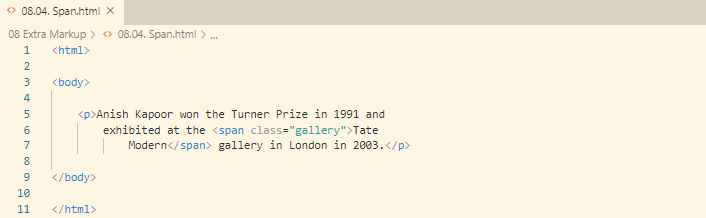
### SPAN

Elemen <span> bertindak seperti Inline yang setara dengan elemen <div>. Elemen ini digunakan untuk:

* Mengandung bagian teks di mana tidak ada elemen lain yang cocok untuk membedakannya dari teks di sekitarnya.
* Mengandung sejumlah elemen sebaris

Alasan paling umum mengapa orang menggunakan elemen <span> adalah agar mereka dapat mengontrol tampilan konten elemen-elemen ini menggunakan CSS. Anda biasanya akan melihat bahwa atribut Class atau Id digunakan dengan elemen <span>:

* Untuk menjelaskan tujuan elemen <span> ini
* Sehingga style CSS dapat diterapkan ke elemen yang memiliki nilai spesifik untuk atribut ini



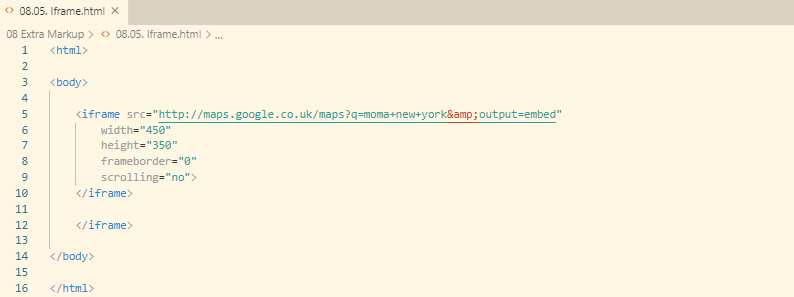
### IFRAME

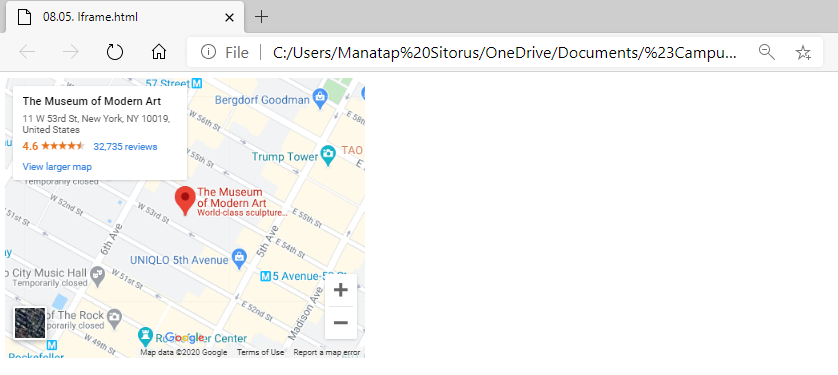
Iframe seperti jendela kecil yang telah dipotong ke halaman, dan di jendela itu Anda dapat melihat halaman lain.

Istilah iframe adalah singkatan dari Inline Frame. Salah satu penggunaan iframe yang umum (yang mungkin telah Anda lihat di berbagai situs web) adalah untuk menyematkan Google Map ke dalam sebuah halaman.

Konten iframe dapat berupa halaman html apa saja (baik yang terletak di server yang sama atau di mana pun di web). Iframe dibuat menggunakan elemen <iframe>. Ada beberapa atribut yang perlu Anda ketahui untuk menggunakannya:

* Atribut Src menentukan URL halaman yang akan ditampilkan dalam bingkai.
* Atribut Height menentukan tinggi iframe dalam piksel.
* Atribut Width menentukan lebar iframe dalam piksel.





### META

Elemen <meta> ada di dalam elemen <head> dan berisi informasi tentang halaman web itu. Elemen ini tidak terlihat oleh pengguna tetapi memenuhi sejumlah tujuan seperti memberi tahu mesin pencari tentang halaman Anda, siapa yang membuatnya, dan apakah itu sensitif terhadap waktu atau tidak. (Jika halaman sensitif terhadap waktu, dapat disetel untuk expire.) Elemen <meta> adalah elemen kosong sehingga tidak memiliki tag penutup. Ini menggunakan atribut untuk membawa informasi.

Atribut yang paling umum adalah atribut Name dan Content, yang cenderung digunakan bersama. Atribut ini menentukan properti dari seluruh halaman.

Value atribut Name adalah properti yang Anda atur, dan nilai atribut konten adalah nilai yang ingin Anda berikan ke properti ini. Di baris pertama contoh di halaman yang berlawanan, Anda bisa melihat elemen <meta> di mana atribut nama menunjukkan niat untuk menentukan deskripsi untuk halaman tersebut.



Atribut Content adalah tempat deskripsi ini sebenarnya ditentukan. Nilai atribut Name dapat berupa apa pun yang Anda inginkan. Beberapa nilai yang ditentukan untuk atribut ini yang umum digunakan adalah:

* Description
  + Ini berisi deskripsi halaman. Deskripsi ini biasanya digunakan oleh mesin pencari untuk memahami tentang halaman dan maksimum 155 karakter. Kadang-kadang juga ditampilkan dalam hasil mesin pencari.
* Keywords
  + Ini berisi daftar kata-kata yang dibuat sebelumnya yang dapat dicari pengguna untuk menemukan halaman. Dalam praktiknya, ini tidak lagi memiliki efek nyata pada bagaimana mesin pencari mengindeks situs Anda.
* Robot
  + Ini menunjukkan apakah mesin pencari harus menambahkan halaman ini ke hasil pencarian mereka atau tidak. Nilai noindex dapat digunakan jika halaman ini tidak boleh ditambahkan. Nilai “nofollow” dapat digunakan jika mesin pencari harus menambahkan halaman ini dalam hasil mereka tetapi tidak halaman mana pun yang terhubung.

Elemen <meta> juga menggunakan atribut http-equiv dan konten berpasangan. Dalam contoh kami, Anda dapat melihat tiga contoh atribut httpequiv. Masing-masing memiliki tujuan yang berbeda:

* Author
  + Ini mendefinisikan penulis halaman web.
* Pragma
  + Ini mencegah browser dari caching halaman. (Yaitu, menyimpannya secara lokal untuk menghemat waktu mengunduhnya pada kunjungan berikutnya.)
* Expires
  + Karena browser sering men-cache konten suatu halaman, opsi kedaluwarsa dapat digunakan untuk menunjukkan kapan halaman itu akan kedaluwarsa (dan tidak lagi di-cache). Perhatikan bahwa tanggal harus ditentukan dalam format yang ditunjukkan.

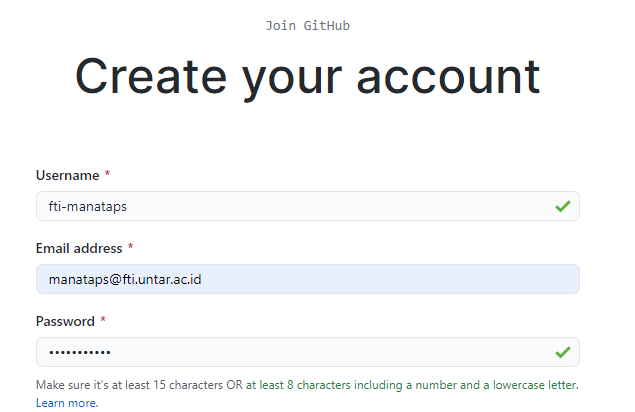
## GitHub

### Pendahuluan

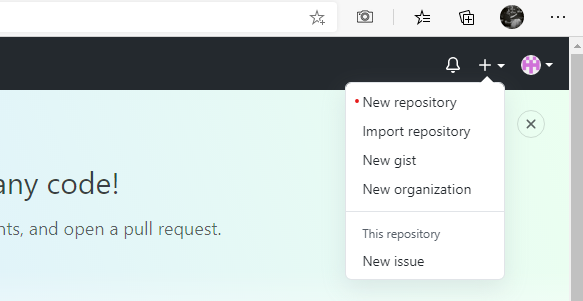
Salah satu fungsi GitHub adalah sebagai source control. Apabila Anda bekerja dalam sebuah tim, maka diperlukan suatu alat yang berfungsi untuk menyamakan source code diantara anggota tim.

### Registrasi

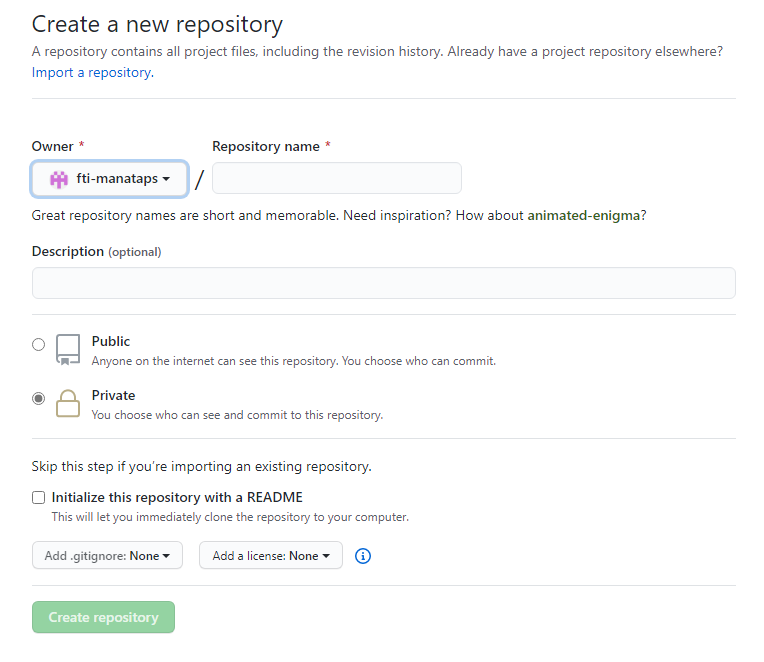
Untuk kebutuhan praktikum, gunakan format nama dan email Untar seperti di bawah ini. Setelah registrasi, pastikan Anda cek email, da klik link di dalamnya untuk menyelesaikan tahapan registrasi.



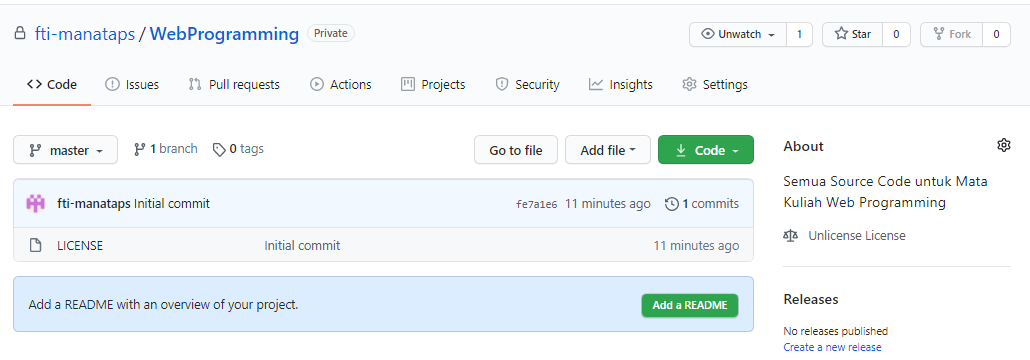
Tambahkan sebuah Repository ke halaman Github Anda. Repository adalah sebuah tempat untuk meletakkan semua resource project. Klik tanda + di atas kanan, lalu pilih New Repository.



Lengkapi informasi nama Repository, deskripsi, dan visibility Public/Private. Untuk saat ini, pilih Private. Setelah selesai klik Create Repository.



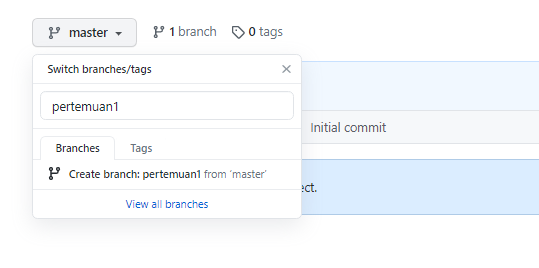
Berikut adalah tampilan repository.



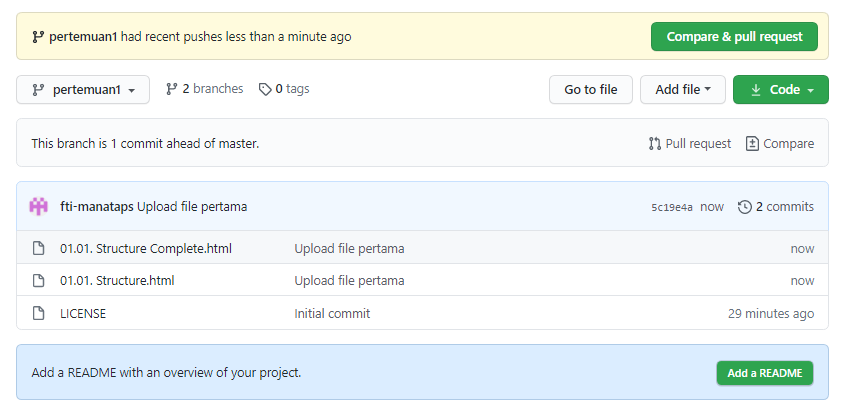
### Branch

Saat kita mengembangkan suatu software, adakalanya membutuhkan suatu perubahan, dengan tetap mempertahankan versi sebelumnya. Alih-alih mengubah nama file menjadi SignIn.html menjadi SignIn-original.html atau SignIn-v1.html atau SignIn-v2.html, GitHub memberikan suatu fitur yaitu Branch.

Apabila dicabangkan, seluruh isi branch akan dicopy ke branch yang baru. Anda dapat mengubah, menambah, dan menghapus file tanpa mempengaruhi branch utama. Untuk menambahkan cabang baru, klik pada Branch, lalu ketikkan nama branch yang baru.



Setelah sukses menambahkan Branch, maka tampilan diarahkan ke branch yang baru. Tambahkan beberapa file ke branch `pertemuan1` dengan mengklik button Add File.

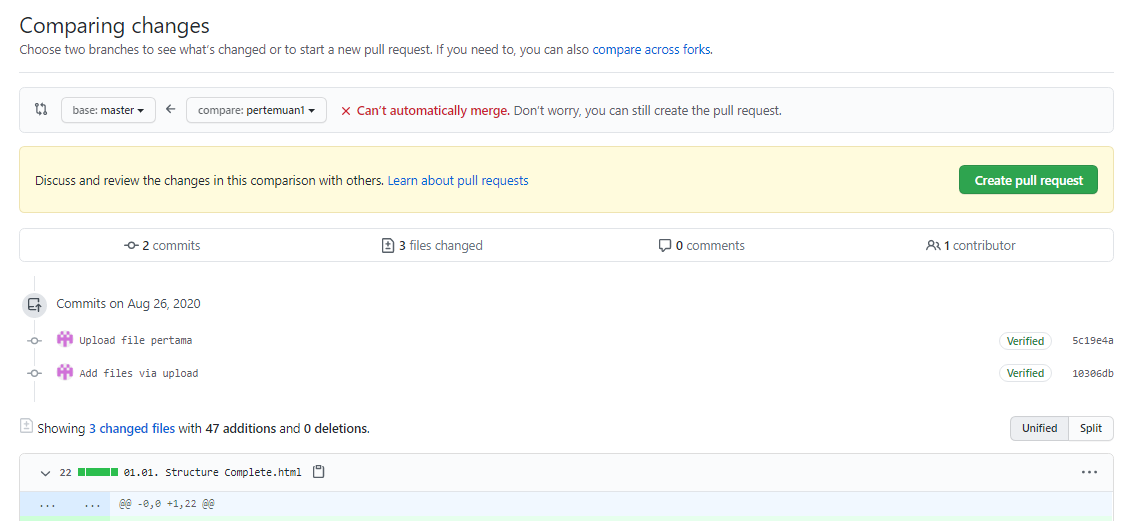


Perhatikan bahwa kita sekarang memiliki 2 versi source code yang berbeda, yaitu `Master` dan `pertemuan1`. Apabila Anda navigasi ke branch Master, maka kedua file yang diupload tadi tidak ada di branch tersebut.

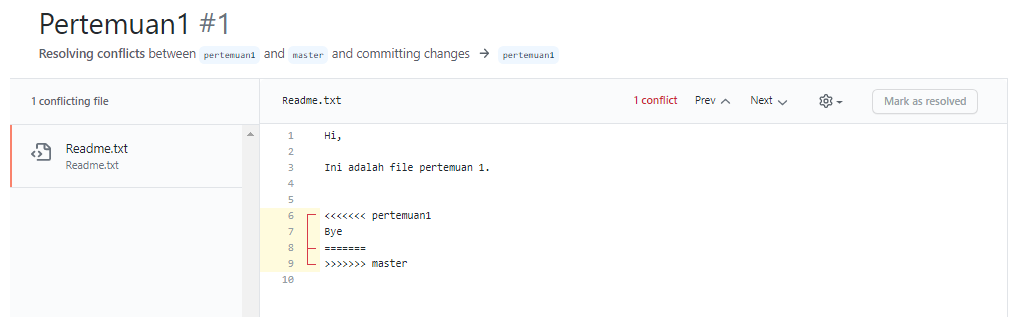
### Pull Request

Setelah melalui proses pengembangan dan ujicoba, pada akhirnya branch akan digabungkan ke Master. Untuk menggabungkan branch dengan master, GitHub menyediakan fitur Pull Request. Fitur ini bertugas untuk menggabungkan source code sekaligus memberikan informasi secara detil, file apa saja yang ditambahkan, diubah, dihapus, sampai ke level source code itu sendiri. Untuk memulai, klik tab Pull Request.

Pastikan branch Base dan branch Compare berbeda. Klik Create Pull Request untuk memulai compare dan pull.



Apabila terdapat suatu conflict, yaitu adanya suatu file yang memiliki konten berbeda antara 2 branch, maka perbaiki isi file tersebut, kali klik Mark as Resolved. Setelah semua konflik telah resolve, maka klik Merge, lau klik Confirm Merge. Lihat kembali branch Master, perhatikan terdapat 2 file baru sesuai branch `Pertemuan1`.



Apabila Anda bekerja dalam lingkungan yang terdiri dari beberapa programmer dan beberapa Test, Pull Request akan menjadi suatu proses cukup panjang. Sebuah Pull Request memungkinkan terjadinya suatu diskusi sebelum dilakukannya Confirm Merge. Diskusi antar anggota tim ini diperlukan untuk saling cek apakah perubahan yang dilakukan dapat memberikan dampak yang tidak baik di fitur-fitur lainnya, seperti fungsi yang tidak berjalan atau bahkan dapat menyebabkan crash pada aplikasi.

Pemberian deskripsi dan penamaan sangat penting dalam tool kolaborasi semacam ini. Apabila membuat Branch baru, berikan nama yang cukup dipahami, contohnya: add-new-slider, change-profile-page, remove-history-tabs dst.

Begitupula saat memberikan informasi saat mengupload file, mengubah file, melakukan pull request, contohnya: Menambahkan fungsi untuk menampilkan gambar-gambar pada slider. Membutuhkan trial sesuai test case #56.