

## BAB 1

### Perkembangan Website

Pada bab ini akan dibahas mengenai sejarah dan perkembangan website serta trend yang berkembang saat ini seperti virtualisasi, botnet, internet of thing dan lainnya. Setelah mempelajari bab ini maka pembaca diharapkan dapat:

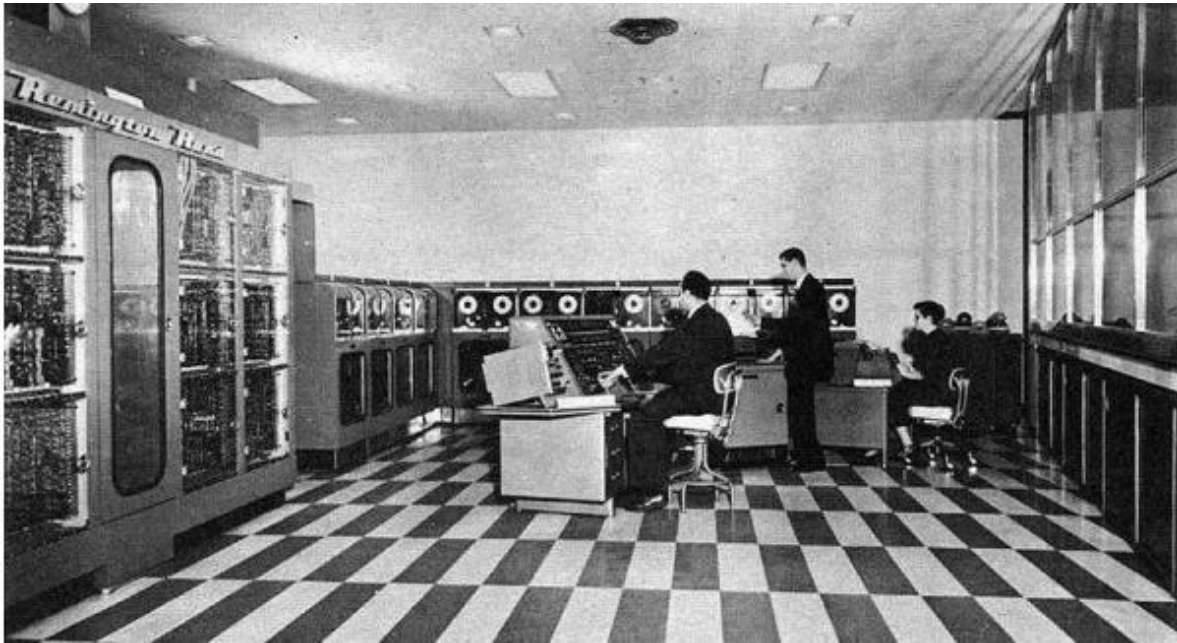
1. Mengerti perkembangan website
2. Mengerti trend yang sedang berkembang dalam pengembangan website
3. Memahami cara kerja cloud computing

## Sejarah Internet

Pada awal perang Dingin sekitar tahun 1940an, konsep membuat jaringan antar komputer sedang dikembangkan oleh pemerintah dan universitas. Pemerintah dalam hal ini diwakili oleh militer mengembangkan alat komunikasi menggunakan sinyal microwave. Namun serangan yang tak terduga membuat militer sadar bahwa teknologi tersebut sangat rentan akan serangan. Di lain sisi, peneliti dari universitas sedang mencari cara untuk saling berbagi hasil penelitian mereka kepada universitas-universitas yang lain namun juga mengalami kegagalan karena sinyal yang tidak stabil. Akhir kata kedua pihak baik militer maupun universitas akhirnya memutuskan untuk saling bekerja sama untuk membuat sebuah jaringan yang lebih baik.

Pada saat itu, komputer belum seperti yang kita kenal saat ini. Sebuah komputer berukuran sangat besar dan hampir memenuhi satu ruangan. Entri data dilakukan menggunakan kertas berlubang atau menggunakan magnetic tape (teknologi paling baru pada jaman itu). Berinteraksi dengan komputer akan sangat menghabiskan banyak waktu karena banyaknya peralatan dan letak antar komponen yang berjauhan. Pada saat itu komputer hanya dimiliki oleh universitas, perusahaan besar dan pemerintah karena membutuhkan biaya dan sumber daya yang sangat banyak untuk melakukan maintenance.

Gambar dibawah ini adalah UNIVAC 1, sebuah sistem komputer yang dimiliki oleh United States Census Bureau. UNIVAC 1 merupakan komputer tercepat pada zaman itu karena dapat melakukan perhitungan 1000 kali dalam satu detik.



## UNIVAC 1

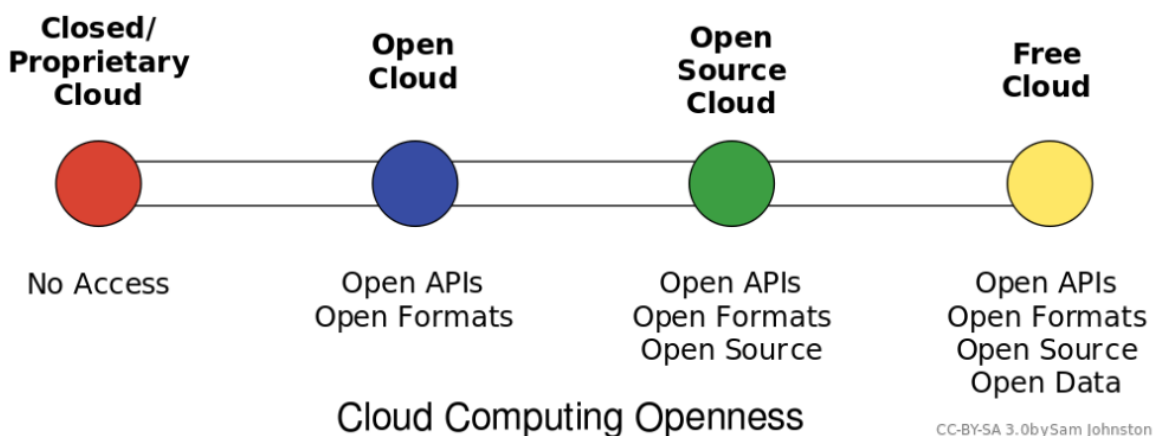
Sebagai perbandingan, pada tahun 2012 Fujitsu memproduksi K Computer, sebuah super komputer yang mampu melakukan perhitungan 10 petaflops per detik. FLOPS sendiri merupakan singkatan dari floating point operation per second yang berarti jumlah yang mampu dikerjakan oleh sistem dalam waktu satu detik. Satu petaflop merupakan indikator seberapa banyak  $10^{15}$  perhitungan yang dapat dikerjakan dalam satu detik. Jadi 10 petaflop berarti K computer dapat mengerjakan  $10^{15}$  perhitungan dikalikan 10 dalam satu detik.

Sejak peluncuran Sputnik, satelit pertama buatan Uni Sovyet, Lembaga pertahanan Amerika Serikat (DoD) mulai mencari cara untuk mengirimkan informasi agar lebih aman. DoD kemudian menemukan metode decentralized, dimana ketika terjadi serangan pada satu titik tidak akan merusak komunikasi pada titik yang lain. Selanjutnya untuk menyeragamkan metode komunikasi maka disusunlah sebuah protokol yang dinamakan Transfer Control Protocol / Internetwork Protocol (TCP/IP). Semakin lama semakin banyak badan pemerintah yang menggunakan standar protokol ini sehingga mereka dapat saling berhubungan walaupun berbeda jaringan. Di sinilah internet dilahirkan.

## Trend Internet Saat ini

Setelah sebelumnya kita membahas sejarah internet, maka kali ini kita juga harus tahu trend-trend yang saat ini sedang berkembang. Inovasi-inovasi baru telah dilahirkan, produk-produk baru seperti Facebook, Twitter dan Youtube menjadi contoh bahwa teknologi web berkembang dengan sangat pesat.

## Cloud Computing



Cloud computing dapat diartikan sebagai penggunaan hardware dan software dibawah sebuah service model (sumber daya dialokasikan saat akan digunakan). Yang sering kita lihat saat ini Cloud Computing merupakan konsep B2B (Business to Business) yang melibatkan Perusahaan A menjual atau menyewakan jasa mereka ke Perusahaan B melalui Internet. Sebuah cloud dapat berupa publik dan dapat diakses secara umum melalui internet atau Private dimana ketika ingin menggunakan jasa tersebut harus melalui firewall dan berada di internal network.

Cloud computing sendiri memiliki beberapa model seperti dalam Infrastructure as a Service (IaaS) sebuah model cloud computing yang paling banyak digunakan saat ini. Dalam IaaS, satu atau beberapa server yang memiliki kapasitas CPU, memory dan tempat penyimpanan yang besar dapat dikelola menjadi sistem yang lebih kecil sesuai dengan kebutuhan dari pengguna. Model ini disebut dengan virtualisasi atau virtual server. Model lain dari cloud computing antara lain Software as a Service (SaaS), Platform as a Service (PaaS) dan masih

banyak lagi yang menggunakan istilah as-a-service seperti Storage as a Service. Konsep yang digunakan juga sama yaitu penggunaan resource yang dapat disesuaikan dengan kebutuhan dari pengguna.

Sebagai contoh, bayangkan dirimu akan membuat sebuah startup internet. Dalam memulai perusahaan anda akan dihadapkan dengan biaya pembelian server and software untuk data storage, web hosting hingga office yang masing-masing komponen ini membutuhkan biaya yang besar. Daripada melakukan setup sendiri yang menghabiskan biaya dan tenaga maka kamu dapat menyewa dari penyedia service cloud computing. Misal dapat menggunakan service dari Amazon Web Services (AWS), dengan biaya yang lebih murah maka kamu akan mendapatkan hosting dan file storage yang sesuai dengan perusahaan kamu. Dengan sistem yang dapat dikelola dan ditambah jika perusahaan kamu berkembang pesat dan membutuhkan kapasitas yang lebih besar.

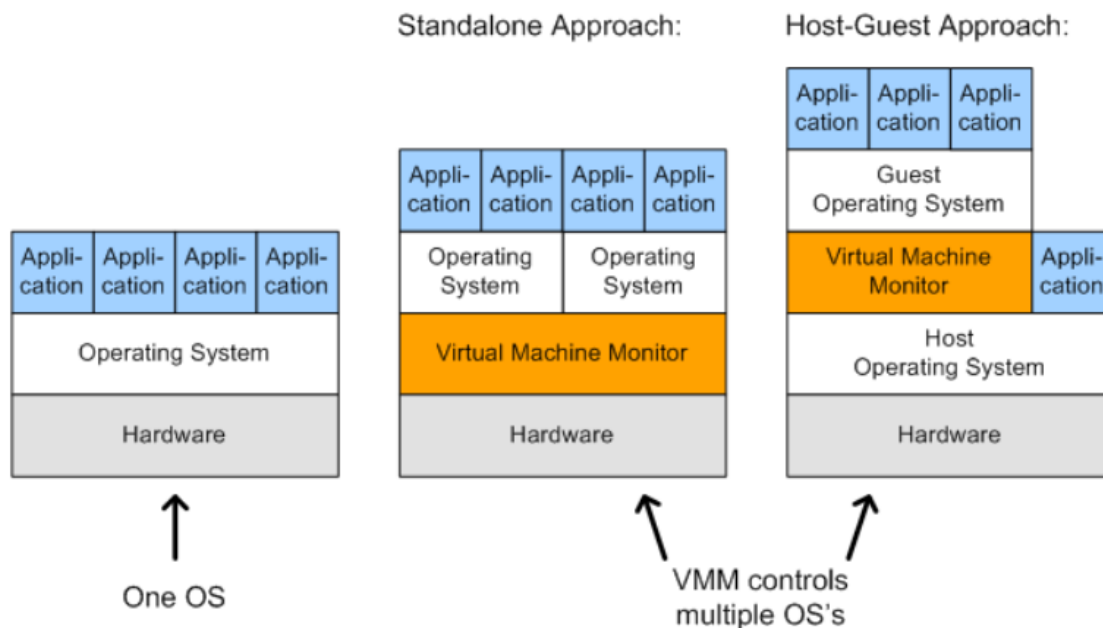
Contoh lain yang sering kita pakai yaitu Google Drive atau Dropbox, dengan menggunakan service mereka kita dapat mendapatkan tambahan storage yang dapat digunakan untuk menyimpan data-data penting dan dapat diakses dari mana saja selama ada sambungan internet.

## Virtualisasi

Virtualisasi merupakan cara untuk menjalankan banyak sistem operasi ataupun software dalam satu hardware yang sama. Sebuah software akan berperan untuk mengelola resource pada layer diantara Operating System dan Hardware itu sendiri. Dengan ini maka resource yang dimiliki dapat dibagi ke dalam beberapa server kecil yang dapat diatur sesuai dengan kebutuhan pengguna.

Virtualisasi dapat dianalogikan sebagai badut yang sedang melakukan atraksi jugling bola. Imajinasikan bahwa tangan badut adalah server sedangkan bola menjadi operating System. Saat ingin memulai atraksi jugling maka tiap tangan akan memegang satu bola dan jika ingin melakukan kegiatan lain maka harus meletakkan bola tersebut terlebih dahulu. Posisi ini bisa

disebut tanpa virtualisasi. Jika atraksi jugling dimulai maka badut dapat memegang tiga bola atau lebih secara bergantian. Sehingga setiap bola (operating system) dapat menggunakan resource yang disediakan oleh tangan (server) secara bergantian.



*By Daniel Hirschbach [CC-BY-SA-2.0 Germany] via Wikimedia*

Pada gambar diatas dengan adanya virtualisasi maka pengelolaan resource dapat dilakukan oleh virtual machine monitor. Monitor ini membagi resource menjadi beberapa bagian yang dapat bekerja secara independent.

## Botnets

Botnet sebenarnya bukan hal yang baru pada dunia internet, namun sampai sekarang tetap menjadi ancaman yang selalu ada bagi user. Botnet merupakan program yang menggunakan koneksi jaringan untuk berkomunikasi dengan botnet yang lain untuk saling berkoordinasi dan melakukan serangkain serangan. Pada awalnya botnet diciptakan untuk menstabilkan koneksi untuk program Internet Relay Chat (IRC) namun sekarang lebih banyak digunakan sebagai sarana untuk serangan internet.



*By Tom-B [CC-BY-SA-3.0], via Wikimedia*

### Internet of Things

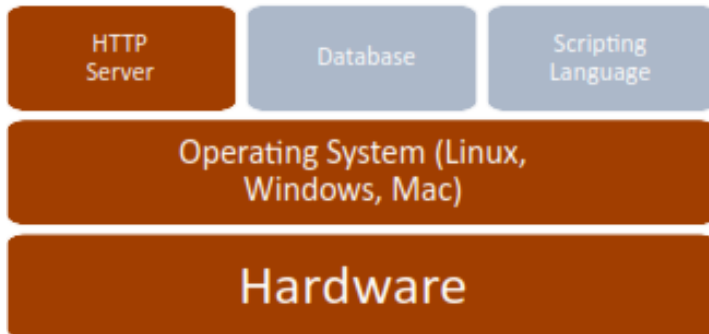
Internet of things (IoT) merupakan konsep yang dicetuskan oleh Kevin Ashton pada tahun 1999, konsep ini bertujuan untuk memanfaatkan konektivitas internet yang tersambung secara terus-menerus untuk berbagi data melalui sensor yang tertanam pada benda di dunia nyata.

IoT sendiri sudah banyak diterapkan pada kehidupan di sekitar kita seperti barcode pada sebuah produk. Kartu e-money yang digunakan untuk membayar pintu tol dan MRT hingga penerapan pada kendali mobil yang bisa kita remote secara jarak jauh.

### Web server

Walaupun tujuan dari buku ini untuk mempelajari pembuatan suatu website namun alangkah lebih baiknya kita juga tahu bagaimana cara kerja hardware dan software yang berada di belakang website tersebut. Untuk menyajikan suatu website yang dapat diakses oleh orang

maka dibutuhkan suatu web server. Secara umum suatu web server dapat digambarkan seperti berikut



### Hardware

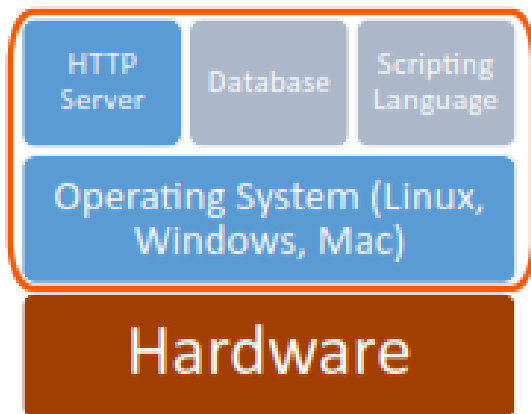
Jika bicara tentang hardware pada web server maka yang terbayang adalah sebuah data center yang besar berisi ratusan rak server yang berjejer disambung dengan ribuan kabel yang terjulur rapi padahal laptop yang biasa kita gunakan untuk mengetik laporan praktikum sebenarnya juga bisa berfungsi sebagai web server. Tentu kapasitas dan fungsinya tidak bisa menyaingi data center milik Google atau Facebook, yang penting sudah mampu untuk menjalankan fungsi minimal sebagaimana web server pada umumnya.

Beberapa perusahaan besar seperti Google, Yahoo dan Facebook mulai berinovasi untuk perancangan sebuah data center. Google misalnya sudah membuat server yang mampu hidup dari sebuah baterai 9 volt, google juga sudah bereksperimen dengan menempatkan server dibawah air atau didasar lautan untuk meminimalkan penggunaan pendingin ruangan.

### Software

Sebuah web server biasanya memiliki 4 elemen selain hardware, antara lain operating system, http server, database dan scripting language. Salah satu software yang paling sering digunakan adalah LAMP. Sesuai namanya software ini terdiri dari Linux, Apache, MySQL dan PHP. Beberapa software tersebut dapat dikombinasikan sesuai dengan operating system yang digunakan sehingga menciptakan singkatan baru seperti WAMP untuk Windows dan MAMP untuk Macintosh.





### Operating System – Linux

Operating System (OS) merupakan system yang menghubungkan aplikasi dan hardware sehingga komputer dapat hidup dan menjalankan fungsinya. OS berfungsi menyediakan resource yang dibutuhkan oleh aplikasi dengan berjalan diantara hardware dan software. Biasanya, OS pada sebuah web server dibagi menjadi tiga kategori yaitu berbasis Linux, berbasis Windows dan berbasis Mac. Setiap kategori ini memiliki keunggulan dan kelemahan masing-masing, namun pada bahasan kali ini akan dikhususkan pada server yang berbasis linux saja.

Linux dikembangkan oleh Linus Torvalds pada awal tahun 1990 saat dia masih menjadi mahasiswa. Linux dikembangkan oleh linux untuk mengakses sistem UNIX yang ada pada universitasnya. Semakin lama project pribadi linux ini semakin berkembang sehingga dia memiliki ide untuk membagi source codenya kepada orang lain. Dengan semakin banyaknya orang yang bergabung untuk mengembangkan linux maka sistem ini semakin besar dan semakin banyak diterapkan pada sistem sekitar kita. Contohnya smartphone android yang kita pakai sehari-hari menggunakan linux sebagai inti operating systemnya.

### HTTP Server – Apache

Apache merupakan web server yang diciptakan untuk system UNIX namun saat ini bisa dijalankan hampir di semua OS seperti Linux, Windows dan Mac. Dikembangkan pertama kali pada tahun 1995 dengan sistem open source sehingga memungkinkan banyak orang

untuk ikut berkontribusi pada pengembangan aplikasi ini. Orang-orang yang ikut mengembangkan apache kemudian mendirikan The Apache Foundation.

Apache memiliki fungsi untuk menampilkan tampilan halaman website yang berasal dari hasil compile bahasa pemrograman, database dan HTML. Apache memungkinkan kita untuk menampilkan halaman web, script dan file dengan user yang membuka website kita. Apapun hasil dari database dan bahasa pemrograman yang digunakan akan diubah kedalam format HTML dan ditampilkan pada browser.

### Configuration Files

Ketika kita membuat sebuah server baru kadang setingan default yang ada tidak sesuai dengan apa yang kita inginkan. Semisal kita ingin menambah, menghapus atau mengganti setingan pada server maka kita perlu mengubah configuration file. Configuration files pada apache bernama httpd.conf, untuk PHP memiliki nama php.ini dan MySQL memiliki nama my.cnf. Letak configuration file ini berbeda-beda tergantung dengan OS yang digunakan. Configuration file biasanya berbentuk file text yang dapat dibuka menggunakan text editor seperti notepad atau vim. Jika ingin mengubah setingan pada server seperti pengaturan port, folder default http server, mengubah nama maka editlah file httpd.conf. Jika ingin mengubah module yang diinstal, menerapkan atau mematikan module maka editlah file php.ini. Sedang my.cnf yang merupakan file konfigurasi MySQL dapat digunakan untuk mengatur port database, mengatur user database dan admin database.

Setiap ubahan yang kita lakukan pada configuration files ini mengharuskan kita untuk melakukan restart pada web server. Jika menggunakan web server yang memiliki tampilan / GUI tombol restart ada pada control panel aplikasi web server. Berbeda jika menggunakan sistem linux untuk merestart server dapat dilakukan di terminal dengan mengetikkan “service httpd restart”

### FTP

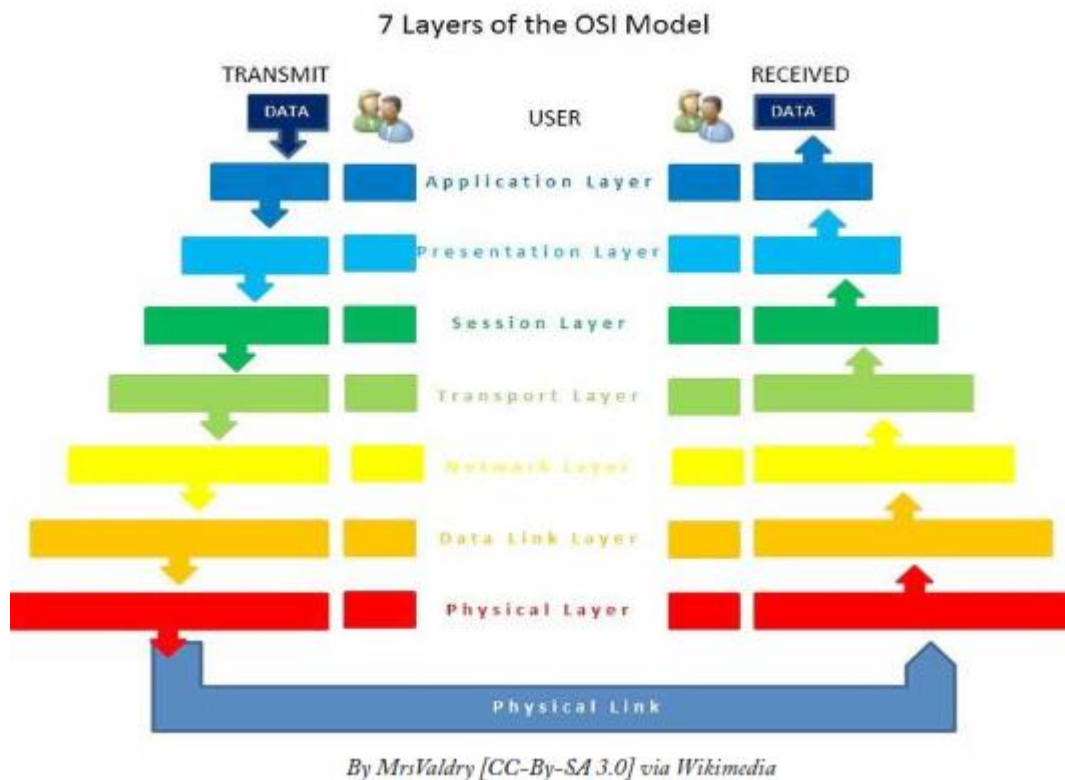
Walaupun tidak termasuk pada kepanjangan LAMP, setiap aplikasi webserver akan memiliki FTP (File transfer protocol) Server. FTP merupakan protokol yang digunakan untuk memindah file dari komputer ke server atau sebaliknya. Untuk menggunakan protokol ini maka diperlukan FTP client seperti filezilla. Filezilla dapat menampilkan file yang berada

pada komputer dan file yang berada pada server. Untuk memindah file antar system cukup dengan melakukan drag and drop pada file tersebut

## Network

### IP Address

IP (internet Protocol) Address merupakan kode unik yang digunakan untuk mengidentifikasi perangkat yang terhubung ke internet. Alamat ini digunakan sebagai sarana komunikasi antar perangkat jaringan seperti wireless pada laptop dengan router wifi atau dengan perangkat yang ada pada ISP (Internet Service Provider).

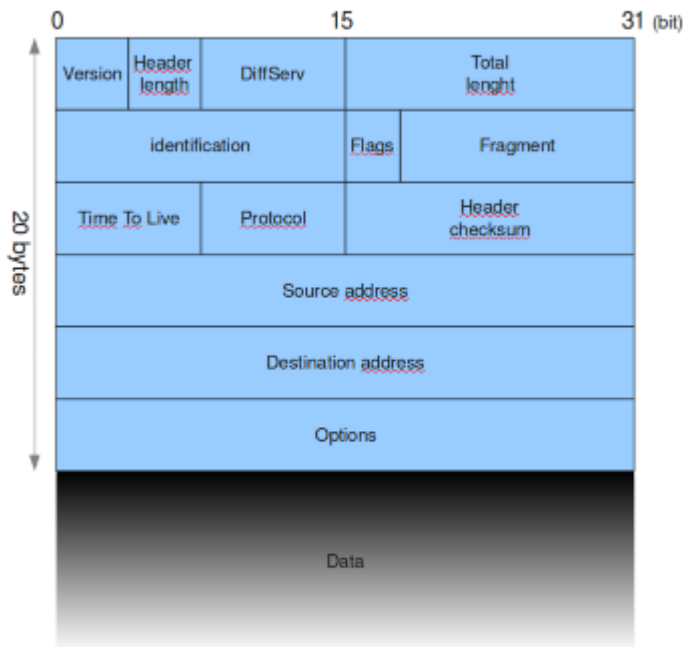


IP Address bekerja pada layer network, satu diantara tujuh layer pada model OSI. Model OSI memiliki kepanjangan Open System Interconnection sebuah standar yang dikembangkan oleh International Organization for Standardization. Model OSI membagi cara kerja network menjadi beberapa layer sesuai dengan fungsi yang ada pada layer tersebut. Pada diagram diatas, data dari user akan dikirim menuju kebawah sesuai dengan arah anak panah. Pertama

akan melalui Application Layer hingga Physical Layer kemudian akan dikembalikan kepada user dengan cara sebaliknya.

Saat ini, kebanyakan perangkat jaringan menggunakan IP versi 4 yang sudah ada sejak 30 tahun yang lalu. Alamat pada IPv4 direpresentasikan ke dalam 4 blok dibatasi titik yang masing-masing terdiri dari 3 digit angka. Angka ini memiliki nilai maksimal 255. Contoh IPv4 seperti 127.0.0.1 atau 103.107.100.1. Khusus untuk IP Address dengan segmen 192.168.xxx.xxx, 172.16.xxx.xxx hingga 172.31.xxx.xxx dan 10.0.xxx.xxx hingga 10.255.xxx.xx merupakan IP Address yang digunakan untuk jaringan lokal. Maksudnya jika kita ingin membangun jaringan pada suatu laboratorium komputer maka kita dapat menggunakan segment alamat IP tersebut. Secara matematika IPv4 akan memiliki 4.3 juta alamat yang bisa digunakan namun jumlah tersebut telah habis pada tanggal 4 Februari 2012. Untuk itu maka dikembangkan IPv6, pada IPv6 penulisan alamat menggunakan hexadecimal yang dibagi menjadi delapan blok yang dipisahkan oleh tanda titik dua. Contoh alamat IPv6 yaitu 2001:0db8:85a3:0042:1000:8a2e:0370:7334.

## IP version 4 packet



*By Nicolargo [ CC-BY-SA-3.0-2.5-2.0-1.0] via Wikimedia Commons*

Pesan atau data yang dikirim antar mesin akan dipecah menjadi beberapa bagian. Bagian-bagian ini disebut dengan packet, yang akan dikirimkan satu persatu dari pengirim ke penerima. Packet akan dikirimkan dengan menggunakan jalur jaringan yang berbeda agar diterima dengan cepat. Dengan cara ini paket akan diterima secepat mungkin namun memiliki kelemahan paket yang diterima urutannya tidak sesuai dengan paket yang dikirim. Untuk mengatasi problem ini maka setiap data yang dikirim akan memiliki header. Header ini berisi berapa banyak packet yang ada, protokol apa yang digunakan, darimana paket ini datang dan beberapa informasi lainnya.

Setelah packet diterima dan disusun ulang oleh penerima maka penerima akan mengirim balik response dan proses pengiriman paket akan dilanjutkan kembali.

## URL

Karena menghafal IP Address itu susah maka untuk mengakses suatu website kita dapat menggunakan URL (Universal Resource Locator). Misal situs twitter memiliki alamat IP Address di 173.252.100.16. Akan lebih mudah jika kita menulis alamat twitter.com pada browser daripada menghafal angka-angka tersebut. Sebuah server khusus yang dinamakan DNS Server berfungsi menerjemahkan request ini. Ketika kita mengetik twitter.com maka DNS server akan mencari alamat IP Address yang sesuai dengan nama domain twitter.com kemudian menampilkan hasilnya di browser.

URL sendiri terbagi menjadi 3 bagian, yaitu protokol, nama dan resource id. Protokol berfungsi untuk mengirim dan menerima pesan contohnya http untuk akses website dan ftp untuk akses file. Nama adalah situs yang akan kita buka, contohnya twitter.com sedangkan resource id (URI) merupakan alamat apapun setelah nama yang menunjukkan halaman web tertentu.

## Ports

Jika IP address dan URL digunakan untuk mengakses website maka ada beberapa cara lain yang dapat digunakan untuk mengakses fungsi tertentu pada web server. Misal kita ingin mengakses email server atau mengakses ftp server untuk melakukan update website maka kita dapat menggunakan port yang ada pada web server tersebut. Beberapa port memiliki aturan standar seperti port 80 merupakan port untuk web service, port 21 khusus untuk FTP dll. Sebenarnya kita dapat mengganti port tersebut namun beberapa firewall akan melakukan blocking jika port yang digunakan oleh web server kita tidak sesuai dengan standar yang ada.

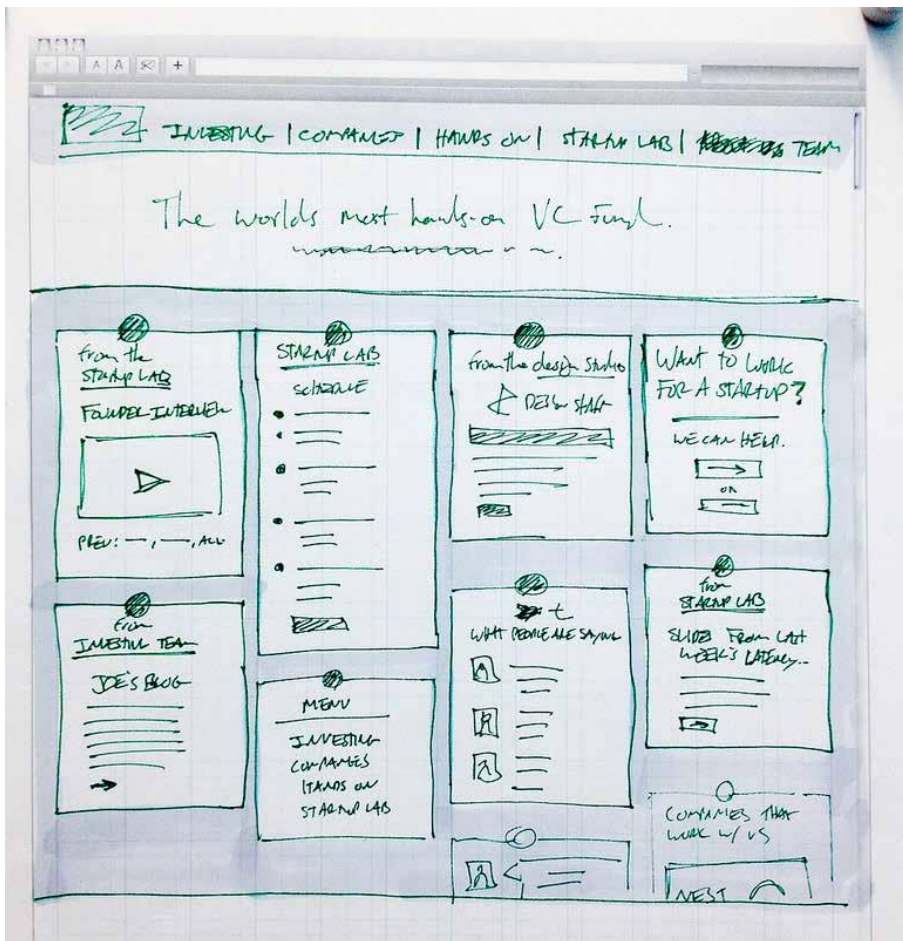
## Domain dan Hosting

Agar website yang sudah kita buat dapat diakses melalui internet maka kita perlu menaruh file website tersebut pada server. Cara paling mudah dengan membeli hosting dan domain pada perusahaan hosting. Dengan cara ini kita tidak dipusingkan dengan cara pemasangan server, biaya sewa rak server dan maintenance server. Semua hal tersebut akan dilakukan oleh perusahaan hosting dan kita dapat fokus pada pengembangan website.

## Pengembangan Website

### Wireframes

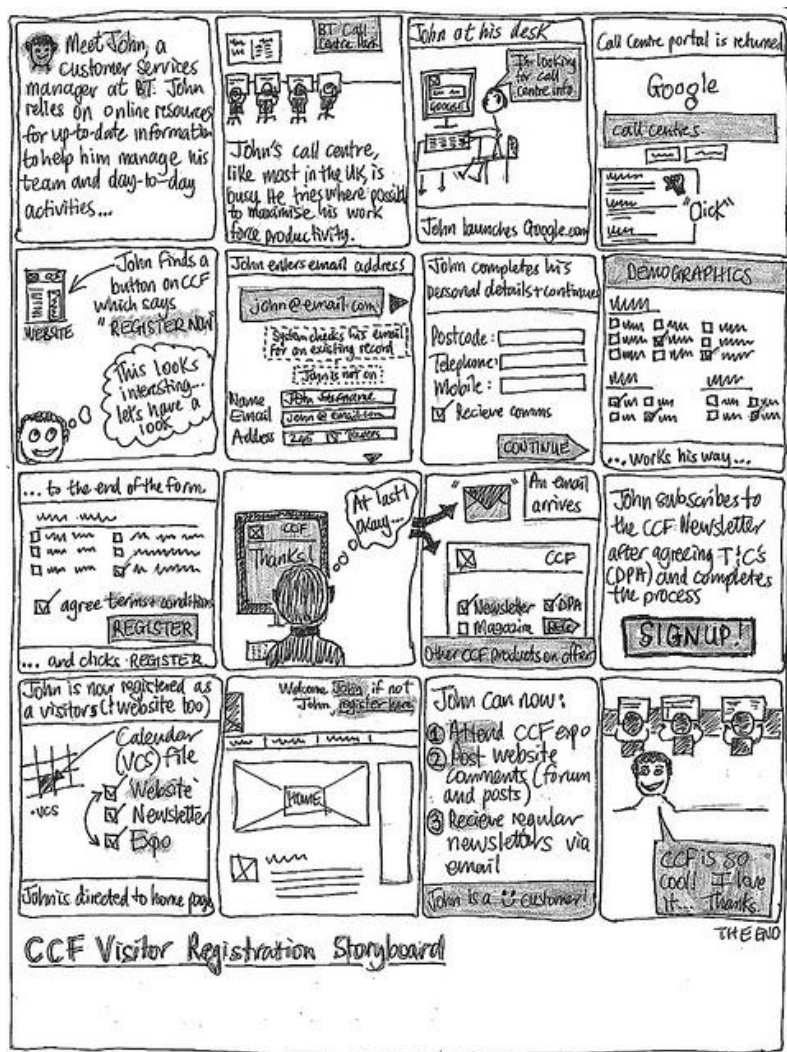
Wireframe digunakan untuk merancang desain kasar suatu website. Wireframe dapat berisi beberapa element seperti button, banner, content dan cara kerja halaman tersebut. Ketika membuat suatu wireframe maka tujuan kita adalah menyajikan kepada user bagaimana website tersebut akan beroperasi. Suatu wireframe biasanya tidak menggunakan warna, konten asli atau typography yang apik. Selama user mengerti bentuk rancangan dari website kita maka tujuan dari penggunaan wireframe telah tercapai.



### Storyboard

Storyboard pada website hampir sama jika kita melihat storyboard pada pembuatan game atau animasi. Dengan berdasar wireframe yang telah dibuat kita dapat menambah warna, text, font dan beberapa gambar untuk lebih menampilkan maksud dari setiap halaman website.

Misalnya kita ingin merancang website toko online maka pada storyboard kita dapat menambahkan gambar barang, cart, pembelian dan pembayaran. Dengan menggambarkan setiap prosesnya kita dapat melihat progress dari user experience dan mengidentifikasi problem yang kemungkinan muncul seperti letak logo cart yang ternyata terlalu keatas atau letak gambar barang yang terlalu besar. Proses storyboarding ini dapat dilakukan berulang kali hingga ditemukan versi yang paling baik.



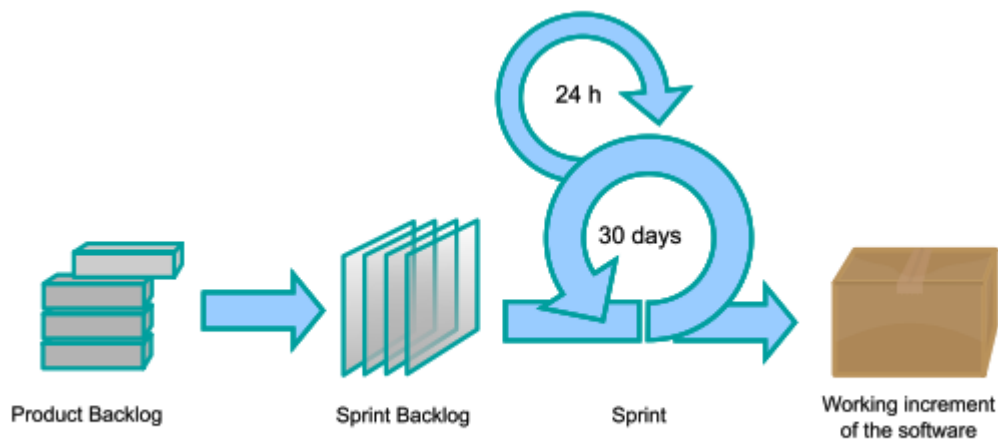


## Project Manajement

### Agile

#### Scrum

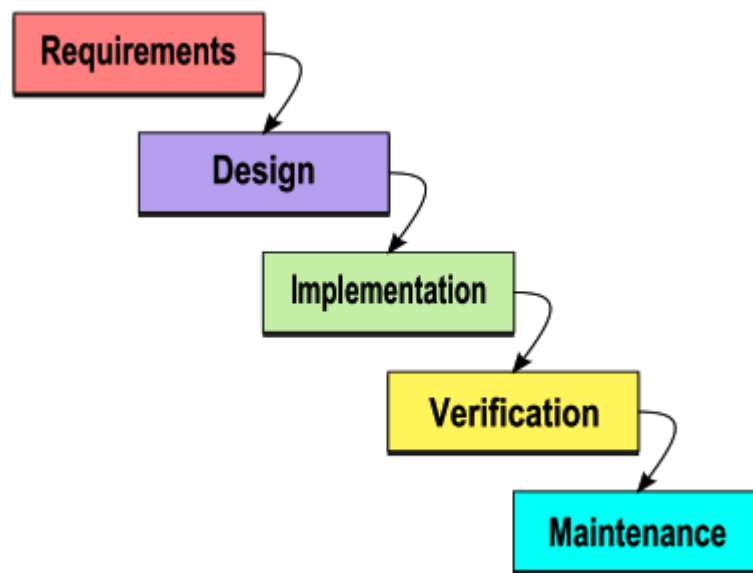
Proses pada metode scrum lebih terfokus dan terorganisir daripada metode agile. Pertemuan dilaksanakan setiap hari dengan membahas apa yang telah dilakukan sejak pertemuan terakhir dan apa yang akan dilakukan pada hari itu. Goal dari scrum disebut dengan sprint. Satu sprint biasanya memakan waktu kurang dari 30 hari diawali dengan meeting yang disebut dengan kickoff.



*By Lakeworks [CC-BY-SA-3.0-2.5-2.0-1.0], via Wikimedia Commons*

#### Waterfall

Metode waterfall dibagi menjadi 5 tahapan seperti gambar



*Paulsmith99 at en.wikipedia CC-BY-3.0, via Wikimedia Commons*

### Requirement

Pada tahap ini pengembang akan bertemu dengan user dan akan membahas tentang project yang akan dibuat

### Design

Pengembangan storyboard, wireframe dan mockup yang sekiranya diperlukan dalam pengerjaan project

### Implementation

Proses penulisan code hingga project selesai

### Verification

Testing untuk memastikan semua fungsi memenuhi requirement

### Maintenance

Tugas yang berlanjut walaupun program sudah selesai seperti maintenance database, penambahan fitur baru, modifikasi fungsi dll

## BAB 2

### Markup Language

Pada bab ini akan dibahas apa itu bahasa markup, contoh dari bahasa markup yang digunakan di pengembangan web dan cara menggunakan bahasa markup tersebut. Setelah mempelajari bab ini maka pembaca diharapkan dapat:

1. Mengerti pembuatan dokumen HTML dan CSS
2. Mengerti penggunaan tag html
3. Mengerti menggabungkan video dan audio ke halaman web

HTML5

**HTML**



Hypertext Markup Language (HTML) merupakan bahasa yang digunakan untuk membantu dalam menyediakan struktur sebuah halaman website seperti tabel, form, list, heading dan beberapa element lain yang menandakan awal dan akhir sebuah halaman website. HTML juga dapat digunakan untuk menambahkan file dengan format yang lain seperti PDF, spreadsheet, video dan audio

#### Sejarah

Pada awal pertama kali internet ditemukan, tujuan utama pembuatan suatu website adalah untuk sharing hasil penelitian dan informasi. Tag HTML hanya digunakan untuk menyediakan struktur dan format suatu layout halaman. Sebagai contoh, pengembang web pada jaman itu hanya bertanggung jawab dalam pembuatan heading, bullet, tabel dan warna. Seiring dengan perkembangan website yang mulai digunakan untuk bidang lain seperti pemasaran, pengembang web mulai berkreasi agar halaman suatu website dapat terlihat seperti halaman majalah, iklan atau bentuk lain yang menarik sesuai dengan rancangan yang mereka gambar.

Saat ini, banyak browser sudah mendukung versi HTML terbaru yaitu HTML5. HTML5 membawa banyak fitur baru seperti canvas dan dukungan terhadap file-file media sehingga dapat meningkatkan kemampuan browser untuk memproses dan menampilkan fitur yang kompleks tanpa harus melakukan coding. Pada versi HTML sebelumnya, halaman suatu website yang ingin menampilkan video dan audio harus menyediakan support ke banyak browser dan harus memasukkan library yang bermacam-macam agar dapat berjalan di semua sistem. Dengan HTML5, hal-hal seperti itu sudah tidak perlu lagi sehingga pengembang web dapat fokus untuk menyediakan user experience yang lebih baik.

Pada HTML5 ini kita menggunakan tag untuk memberikan makna pada text, tag ini akan diterjemahkan oleh browser sebagai cara bagaimana text tersebut akan ditampilkan ke layar. Tag yang kita gunakan dalam pengembangan web sudah terstandar oleh W3C (World Wide Web Consortium) sebuah komunitas internasional yang membantu pengembangan web dengan memberikan aturan-aturan standar sehingga dapat memberikan user experience yang terbaik. Komunitas ini mengeluarkan aturan bagaimana CSS dan HTML digunakan, tag apa saja yang bisa digunakan dll. Di dalam website W3C juga disediakan tool yang dapat kita gunakan untuk mengetes apakah dokumen HTML/CSS yang kita buat sudah sesuai dengan standar mereka atau belum.

## CSS



CSS memiliki kepanjangan Cascading Style Sheet dan digunakan untuk membuat aturan bagaimana warna font dan layout suatu halaman website kita. CSS juga dapat digunakan untuk mengontrol kapan aturan tersebut digunakan berdasar perangkat apa yang digunakan untuk mengakses web, ukuran layar dari user hingga berdasar kapan user melakukan klik. Biasanya pembuatan CSS akan dipisah dengan HTML dengan begitu pengembangan web dapat terpisah antara isi dengan tampilan. HTML akan digunakan untuk membuat struktur dari halaman suatu web sedangkan CSS akan digunakan untuk mempercantik tampilan halaman web tersebut.

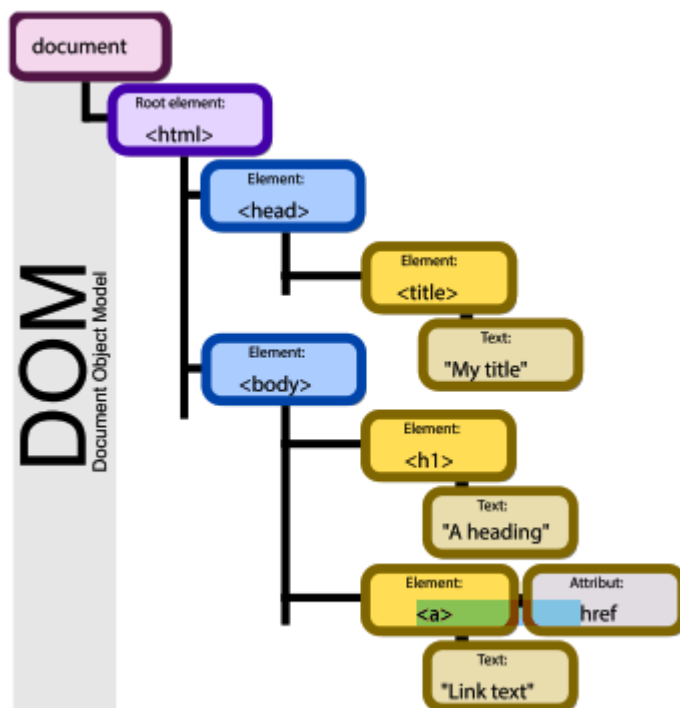
## Sejarah

Dengan berkembangnya HTML yang semakin populer dan banyaknya browser yang memiliki standar berbeda untuk menampilkan suatu halaman web maka semakin besar pula tantangan yang dimiliki oleh pengembang web dalam melakukan perancangan tampilan. CSS

pada awalnya bernama CHSS atau Cascading Hypertext Style Script. Suatu bahasa markup yang digunakan untuk menambahkan fitur-fitur yang tidak dimiliki oleh HTML seperti pengaturan gambar, pembuatan layout yang mirip dengan majalah dan pemilihan bentuk font.

Diperkenalkan pertama kali pada tahun 1996, CSS1 dapat digunakan untuk membuat selector dan properties yang masih dapat kita gunakan sampai sekarang. Versi selanjutnya yaitu CSS2, mulai ada fitur agar halaman web mampu beradaptasi dengan ukuran layar dan perangkat yang digunakan untuk mengakses halaman web. Sedangkan pada CSS3, perubahan besar terjadi dengan dipecahnya CSS3 menjadi modul-modul kecil. Dengan begitu penambahan fitur baru akan tergantung dengan modul mana yang diperbaharui.

## Document Object Model (DOM)



*By Birger Eriksson CC-BY-SA-3.0*

DOM dibuat sebagai bahasa dan platform yang independent sehingga software atau bahasa pemrograman apapun dapat menggunakannya untuk mengatur tampilan pada dokumen. DOM sendiri dapat direpresentasikan sebagai interface atau object dari element suatu dokumen. Gambar diatas merupakan contoh suatu dokumen HTML. Dapat dilihat bahwa

suatu dokumen HTML terdiri dari dua elemen utama yaitu head dan body. Pada masing-masing elemen tersebut memiliki anak elemen lain yang berada di sebelah kanannya. Dengan memahami struktur ini maka kita dapat mengatur tampilan DOM dengan langsung menunjuk nama elemen tersebut.

## Membuat File HTML

### File Format

Sebelum kita membuat website kita, kita harus membuat suatu file yang dapat dibaca sebagai suatu halaman web yaitu html. Untuk membuatnya maka buka text editor yang kalian gunakan dalam project ini. Buat sebuah halaman baru kemudian klik menu File kemudian klik “Save As” dan beri nama index.html. Kenapa diberi nama index.html karena halaman ini akan menjadi halaman pertama dari website kita. Secara default, index merupakan file pertama yang akan dibaca oleh web server. Jangan lupa tipe filenya menggunakan .html. Kalian juga dapat menggunakan .htm . Dua tipe file ini sama, pastikan saja kalian konsisten dalam menggunakan tipe filenya.

### Document Type

Setiap file HTML yang kita buat harus memiliki deklarasi tipe dokumen (doctype) pada baris pertama. Deklarasi ini digunakan sebagai identitas versi berapa HTML yang digunakan sehingga browser mengerti cara menampilkannya ke layar. Pada versi HTML sebelumnya terdapat dua versi doctype yaitu pada versi HTML4.01 dan XHTML. Namun semenjak versi HTML5 deklarasi doctype dapat menggunakan `<!DOCTYPE html>` saja. Doctype ini harus selalu ditulis pada baris pertama halaman HTML.

### Tags dan Attribute

Langkah selanjutnya adalah mendefinisikan tag yang akan kita gunakan dalam dokumen. Tag pembuka halaman html adalah `<html>` sedangkan penutupnya adalah `</html>`. Tag ini merupakan tag yang sepasasang karena menggunakan dua kata yang sama dan kata yang terakhir menggunakan garis miring (/) yang mengindikasikan bahwa tag tersebut sudah ditutup. Selanjutnya, kita tuliskan beberapa tag lain yang biasa digunakan sebagai struktur halaman website. Tag tersebut antara lain head, body, header, dan footer. Sehingga struktur layout dokumen html akan menjadi seperti ini :

```
<html>

<head>

</head>

<body>

<header>

</header>

<footer>

</footer>

</body>

</html>
```

Selanjutnya kita akan menambah spacing untuk memudahkan kita dalam membaca dan menambahkan kata Hello World diantara tag header dan footer kita.

```
<html>

  <head>

  </head>

  <body>

    <header>

    </header>

    Hello World

    <footer>

    </footer>

  </body>
```



</html>

Dengan menggunakan spasing maka kita sebagai pengembang web akan dimudahkan ketika akan melakukan editing file.

Save dokumen tersebut dengan klik menu File – Save. Buka folder tempat dimana kalian menyimpan kemudian klik dua kali file tersebut. Browser bawaan dari komputer akan terbuka dan menampilkan satu halaman background putih dengan tulisan Hello World pada pojok kiri atas. Inilah website pertama kalian!

Karena masih dalam bentuk HTML maka kita tidak perlu menggunakan web server. Web server baru akan kita gunakan saat memulai PHP pada bab berikutnya.

Pada langkah berikutnya kita akan melakukan modifikasi pada file html kita ini. Untuk memudahkan dalam pengerjaan maka tetap buka text editor dan browser secara bersamaan. Setelah melakukan save maka kalian cukup melakukan refresh pada browser untuk melihat perubahan yang terjadi.

## Head

Tag pertama yang ada pada halaman kita setelah html adalah head. Head merupakan tempat dimana kita dapat memberikan informasi tentang halaman yang kita buat. Informasi ini disebut dengan metadata. Selain itu di dalam tag head kita juga dapat mendeklarasikan resource external yang digunakan seperti script, css dll. Pastikan tag <head> pasti berada di bawah tag <html> karena browser akan membaca informasi yang ada pada head.

Informasi yang dapat kita tulis disini antara lain tentang title, keyword, deskripsi dll. Informasi ini akan membantu user dan bot untuk memahami isi dari halaman web kita.

Setiap item dari informasi tersebut akan ditulis dengan tag meta (<meta/>). Beberapa tag meta memiliki tambahan informasi yang disebut dengan attribut. Tag meta sendiri mendukung attribut sseperti nama dan konten. Bagaimana cara kita menulis attribut ini akan membantu browser dalam membaca halaman web kita. Pada project ini title akan diberi nama Halaman Pertama, keyword akan berisi html pertama dan belajar html , deskripsi akan berisi informasi singkat tentang apa yang kita pelajari kemudian tag terakhir yaitu nama.

```
<head>

<title>Halaman Pertama</title>

<meta name="keywords" content="html pertama, belajar html" />

<meta name="description" content="cara mudah belajar html" />

<meta name="author" content="Nama kamu" />

</head>
```

Dari contoh diatas kita dapat mempelajari beberapa hal. Pertama dapat kita lihat bahwa title memiliki tag sendiri. Apa yang kita tulis pada title akan muncul pada nama tab di browser bukan pada halaman web kita.

Tag selanjutnya adalah name dan content. Untuk mengisi tag tersebut kita menggunakan tanda petik dua. Jika isiannya lebih dari satu maka dipisahkan dengan koma dengan catatan masih di dalam tanda petik dua tersebut.

## Body

Semua isi yang ingin kita tampilkan di layar maka harus ditulis diantara tag body. Header, content, footer dan div merupakan beberapa contoh tag yang dapat kita taruh pada body. Tag ini digunakan sebagai guide bagi browser untuk menampilkan isi html ke layar

## Header, Footer

Header dan footer merupakan tag baru yang diperkenalkan sejak HTML5. Tag ini digunakan untuk membantu mendefinisikan bagian header dan footer dari suatu halaman website. Pastikan lagi bahwa kedua tag ini harus berada di dalam tag <body>.

```
<body>

    <header>

        <h1>This is our first page!</h1>

    </header>

    Hello World
```

```
<footer>
```

```
&copy; 2020 Your Name Here
```

```
</footer>
```

```
</body>
```

Simpan file index.html kalian kemudian lakukan refresh pada browser untuk melihat hasilnya. Harusnya kalian akan melihat kata this is our first page pada kiri atas diikuti dengan kata hello world kemudian © 2020 Your Name Here. Kita menggunakan tag <h1> pada title untuk membuat text menjadi besar. H1 sendiri merupakan heading 1, heading terbesar yang tersedia. Kalian dapat mencoba mengubah ukuran text dengan menggunakan tag Heading dari h1 hingga h6

### Div dan Span

Div berasal dari kata divide. Tag ini digunakan untuk mendefinisikan bagian tertentu dari halaman web yang akan diubah. Sedangkan span hampir mirip dengan div namun span hanya untuk satu baris text saja. Secara fungsi kedua tag ini memiliki kegunaan yang sama. Untuk menambahkan div dan span pada halaman baiknya disertakan dengan id. Karena nantinya akan digunakan dalam pembuatan css.

```
<body>
```

```
<header>
```

```
<h1>This is our first page! </h1>
```

```
</header>
```

```
<div id="left">
```

```
some menu items
```

```
</div>
```

```
<div id="content">
```

```
Hello World
```

```
</div>
```

```
<div id="right">
```

and some content on the right

```
</div>
```

```
<footer>
```

&copy; 2013 Your Name Here

```
</footer>
```

```
</body>
```

## Text Layout

### Paragraf

Jika memiliki susunan text yang panjang kita dapat memecahnya dalam bentuk paragraf dengan menambah tag breaks `<br/>` pada kode. Namun jika memiliki paragraf yang panjang sebaiknya menggunakan tag yang khusus untuk paragraf yaitu tag pembuka `<p>` dan tag penutup `</p>`. Dengan menggunakan tag ini maka secara otomatis setiap pergantian paragraf akan ditambah dengan jarak sehingga memudahkan pembaca dalam membaca

Contoh sebelum :

```
<body>
```

This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document.



so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document.

</p>

<p>

This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs

so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. </p>

</body>

### Ordered dan Unordered Lists

Ordered dan unordered lists merupakan tag yang memiliki fungsi yang sama untuk membuat sebuah list. Unordered lists digunakan jika list yang akan kita buat tidak harus dalam suatu urutan. Sedangkan jika list yang dibuat harus dalam bentuk yang urut maka kita menggunakan ordered list. Unordered list menggunakan tag <ul> dan hasilnya akan menggunakan logo titik, sedang ordered list menggunakan tag <ol> dan hasilnya akan menggunakan angka yang urut.

Contoh dari penggunaan tag ini adalah:

<ol>

<li>First</li>

<li>Second</li>

<li>Third</li>

</ol>

1. First

2. Second

3. Third

```
<ul>
```

```
<li>An item</li>
```

```
<li>Another item</li>
```

```
<li>Yet another item</li>
```

```
</ul>
```

- An item

- Another item

- Yet another item

#### Definition Lists

Jika kalian ingin membuat list yang berisi arti dari suatu istilah maka dapat menggunakan definition lists. Ada beberapa tag yang digunakan dalam definition lists. Tag <dl> untuk memulai lists, tag <dt> untuk istilahnya, tag <dd> untuk arti dari istilah yang dimaksud

```
<dl>
```

```
<dt>Coffee</dt>
```

```
<dd>Bean-based caffeinated
```

```
beverage </dd>
```

```
Coffee
```

```
<dt>Tea</dt>
```

```
<dd>Leaf-based caffeinated
```

```
beverage</dd>
```

```
Bean-based caffeinated beverage
```

Tea

Leaf-based caffeinated beverage

<dt>Water</dt>

<dd>Standard H2O</dd>

Water

</dl>

### Address

Tag address digunakan jika kalian ingin menampilkan alamat suatu kontak atau alamat dari konten kreatornya

<address>

Article by <a href="mailto:professor@school.

edu">Prof. Essor</a>.<br>

Fredonia, NY<br>

USA

</address>

Article by Prof. Essor.

Fredonia, NY

USA

### Article

Tag article digunakan jika isi dari tag tersebut dapat digunakan kembali pada layout yang lain. biasanya tag article digunakan untuk tipe konten artikel, berita dan blog

<article>

<h1>Our Blog Post</h1>



`<p>This is our great content that is now identified as`

`something that can exist on its own as a piece of`

`work.</p>`

`</article>`

### Aside

Tag aside digunakan jika kalian ingin menandai suatu potongan artikel yang ingin didefinisikan lebih lanjut.

`<p>`

This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document. This is some text. It is really long. We want to break this into paragraphs so it looks more like a document.

`</p>`

`<aside>`

`<h4>Side Bar</h4>`

`<p>This is something related to our content that is not actually a part of it</p>`

`</aside>`

### Button

Tag button digunakan untuk tombol dalam sebuah halaman web. Dalam sebuah button kita dapat menulis kata atau menampilkannya menggunakan gambar

`<button type="button">Click Here!</button>`

`<button type="button">`

`</button>`



### Caption

Tag caption digunakan sebagai label untuk sebuah tabel. Tag ini hanya boleh digunakan satu kali dalam setiap tabel.

```
<table>

<caption>This is our table</caption>

<tr>

<td>First Col</td><td>Second Col</
td><td>Third Col</td>

</tr>

<tr>

<td>1</td><td>2</td><td>3</td>

</tr>

<tr>

<td>4</td><td>5</td><td>6</td>

</tr>

</table>
```

This is our table		
First Col	Second Col	Third Col
1	2	3
4	5	6

## Meter

Tag meter digunakan untuk membuat visualisasi suatu progress bar. Tag ini biasa digunakan untuk visualisasi sebuah proses

```
<meter value="3" min="0" max="15">One
```

```
Fifth</meter><br>
```

```
<meter value="0.65">65%</meter>
```



## Progress

Tag progress digunakan untuk membantu sebuah status upload atau download. Tag ini membutuhkan dua attribut yaitu value yang menandakan posisi saat itu dan max yang menandakan nilai tertinggi.

```
<progress value="46" max="100"></progress>
```



## Nav

Jika kita ingin membuat grup dari suatu link, biasanya digunakan saat akan membuat menu maka dapat menggunakan tag nav. Nav akan dikenali oleh browser sebagai kumpulan dari link

```
</nav>
```

```
<a href="/">Home</a> |
```

```
<a href="/css/">CSS</a> |
```

```
<a href="/js/">JavaScript</a> |
```

```
<a href="/js/jquery/">jQuery</a>
```

```
</nav>
```

## Navigasi

Fitur yang pasti ditemukan dalam semua website adalah navigasi. Navigasi digunakan untuk mempermudah user jika ingin berpindah halaman. Walaupun tidak ada aturan baku dalam membuat sebuah menu namun sebagian besar pengembang web akan menggunakan tag unordered list kemudian menambahkan beberapa baris kode CSS untuk membuat list tersebut menjadi lebih menarik dan interaktif.

Beberapa menu yang biasa digunakan antara lain menu dropdown, breadcrumbs dan event driven response. Menu dropdown adalah menu yang paling sering kita lihat dalam sebuah website. Ketika kita mengarahkan kursor mouse ke sebuah menu biasanya akan muncul menu baru dibawahnya. Walaupun dinamakan dropdown menu (menu ke bawah), beberapa menu dapat diubah agar memiliki arah ke samping atau keatas dengan menggunakan CSS.

Menu Breadcrumbs merupakan menu horizontal yang menandakan riwayat sebuah halaman yang pernah dibuka

Home >> Sports >> Football >> Buffalo Bills >> Patriots >> Golf

Menu ini akan sangat membantu jika sebuah website memiliki konten yang sangat banyak, sehingga membantu user dalam berpindah topik atau kategori dalam suatu website,

Event-driven navigation berguna untuk membantu mengarahkan user dalam suatu jalur. Metode ini akan menggunakan beberapa link, link tersebut akan aktif dalam kondisi tertentu. Misal suatu link pembayaran tidak akan aktif jika user belum melakukan login.

Beberapa menu diatas dapat digunakan secara terpisah atau digunakan bersamaan tergantung dengan user experience yang ada.

## Linking

Link pada HTML memiliki dua bentuk, link yang menunjuk ke halaman lain dan link yang menunjuk pada posisi tertentu tanpa harus pindah halaman. Kedua bentuk link ini sama-sama menggunakan tag <a>.

Absolute, Base dan Relative Path

Bentuk Link sendiri di bedakan menjadi tiga yaitu absolute, base dan relative. absolut dan relative dapat digunakan untuk membuat link yang menunjuk halaman pada website kita. Tapi hanya absolute yang dapat digunakan untuk menunjuk halaman eksternal.

Absolute path merupakan link yang digunakan untuk menunjukkan resource baik itu halaman, gambar , script atau media. Bentuk link dari absolute path misalnya <http://www.msn.com/news/index.htm> , sedangkan relative path dapat digunakan dalam suatu website jika tujuan dari link tersebut tidak mengarah ke situs eksternal. Misal akan berpindah ke halaman sport dapat menggunakan link relative seperti ini ../sports/index.htm

Base path sendiri merupakan link utama dari suatu website. Contohnya [www.msn.com](http://www.msn.com) atau [www.namakamu.com](http://www.namakamu.com).

### Target

Tag link sendiri memiliki beberapa atribut. Atribut yang sering digunakan adalah target. Tag ini digunakan apakah link yang akan dibuka tetap berada pada tab yang sama atau akan membuka tab baru

Value	Deskripsi
_blank	Membuka link pada tab baru
_self	Membuka link pada tab yang sama
_parent	Membuka link pada parent frame
_top	Membuka link pada window yang baru

### Link didalam halaman

Kita dapat membuat link dalam suatu halaman untuk memudahkan user bernavigasi. Ini akan berguna jika kita memiliki halaman yang panjang. Ketika membuat link ini pastikan tujuan dari link adalah nama dari salah satu id yang digunakan dalam halaman web kita

Some text here.

<a href="#ourDestination">Click here to go

further down.</a>

Some more text.

Even more text!

<a name="ourDestination">

This is where we want to "jump" to.

Some text here.

[Click here to go further down.](#)

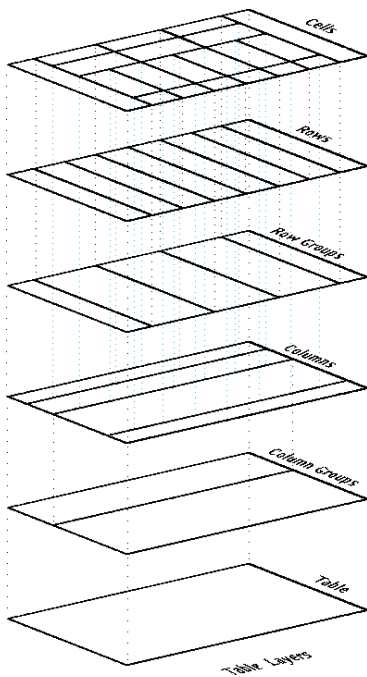
Some more text.

Even more text!

This is where we want to "jump"  
to.

## Tabel

Sebelum HTML5 diluncurkan, tabel bahkan digunakan untuk membuat suatu layout halaman website. Saat ini penggunaan tabel untuk layout halaman website sudah ditinggalkan. Saat ini Tabel hanya digunakan untuk menampilkan koleksi data atau informasi dalam suatu halaman web. bahkan jika kalian ingin menampilkan suatu data atau informasi yang tidak perlu diatur seperti bentuk tabel lebih baik menggunakan tag div atau span.



## Struktur Tabel

Tabel terdiri dari baris dan kolom. Setiap perpotongan baris dan kolom disebut dengan cell.

Di sinilah kita akan mengisi data. Untuk membuat tabel kita akan menggunakan tag :

```
<table></table>
```

Selanjutnya untuk menambah baris pertama kita menggunakan tag <tr>. Biasanya baris pertama adalah judul dari kolom. Maka kita akan menggunakan tag <th>

```
<table>
```

```
<tr>
```

```
<th>ID</th><th>First
```

```
Name</th><th>Last
```

```
Name</th>
```

```
</tr>
```

```
</table>
```

ID	First Name	Last Name
----	------------	-----------

Untuk menambah data, maka kita perlu menambah baris dari tabel diatas. Gunakan tag <td> untuk merepresentasikan data reguler

```
<table>

<tr>

<th>ID</th><th>First Name</
th><th>Last Name</th>

</tr>

<tr>

<th>1</th><td>John</td><td>Doe</td>

</tr>

<tr>

<th>2</th><td>Jane</td><td>Doe</td>

</tr>

</table>
```

ID	First Name	Last Name
1	John	Doe
2	Jane	Doe

### Spanning

Jika kita memiliki tabel dengan data yang banyak dan layout yang rumit. Biasanya beerapa cell akan digabung untuk mempermudah dalam membaca data. Kita dapat mengabungkan



beberapa cell menggunakan atribut colspan dan divspan . misal colspan dengan nilai 2, artinya kita akan menggabungkan 2 cell secara horisontal. Sedangkan rowspan akan menggabungkan 2 baris cell secara vertikal

```
<table>

<tr>

<th>ID</th><th>First Name</
th><th>M</th><th>T</
th><th>W</th><th>Th</
th><th>F</th>

</tr>

<tr>

<th>1</th><td>John</td><td
colspan =/2/>work</
td><td></td><td>
rowspan=/2/>closed</
td><td></td>

</tr>

<tr>

<th>2</th><td>Jane</
td><td></td><td></
td><td>work</
td><td>work</td>
```

</tr>

</table>

ID	First Name	M	T	W	T	F
1	John	work				
2	Jane			work	closed	work

## Form

Form merupakan tempat dimana user berinteraksi dengan halaman web. user dapat menambah, mengganti, berinteraksi atau menghapus data pada sistem. Form merupakan elemen paling penting dalam proses interaksi dengan user dan disini lah tempat pengembang mendapatkan input dari user.

Form memiliki banyak jenis dari form untuk username dan password, form yang memiliki dropdown list, checkbox dll. Untuk membuat form, maka kita harus memberikan form suatu nama, id, action dan method. Contoh pembuatan form adalah seperti berikut

```
<form name="" id="" action="" method=""></form>
```

Nama dan id for merupakan identitas form tersebut dan akan kita gunakan untuk berinteraksi dengan CSS, javascript atau PHP. Action merupakan halaman tujuan dari data yang akan user kirim setelah mereka klik tombol submit. Sedangkan method adalah cara form tersebut mengirim data. Method dibagi menjadi GET dan POST

## GET

Mengirim data menggunakan get akan menampilkan semua data yang dikirim pada address bar di browser. Ini akan membuat alamat URI menjadi panjang sesuai data yang dikirim oleh form tersebut. Keuntungan menggunakan metode GET adalah halaman tujuan pengiriman form dapat dibookmark sehingga memudahkan user ketika ingin mengulangi mengisi form.

Sedangkan kelemahannya adalah ketika data yang dikirim merupakan data sensitif seperti password atau data yang dikirim merupakan informasi yang sangat panjang sehingga akan membuat browser menjadi lambat.

## POST

Berkebalikan dengan GET, data yang dikirim menggunakan metode post akan dilakukan di background sehingga user tidak akan melihat perpindahan data. Metode POST digunakan ketika data yang dikirim merupakan data sensitif atau rahasia

## Form Fields

Ketika suatu halaman website yang memiliki form tampil, biasanya kursor secara otomatis akan berpindah ke form pertama yang harus diisi. Fungsi ini dinamakan autofocus. Untuk menggunakan fungsi ini maka tambahkan atribut autofocus pada form. Kita juga dapat menambahkan beberapa kata sebagai panduan untuk mengisi form. Gunakan atribut placeholder untuk menggunakan fungsi ini.

```
<form name="" id="" action="" method="">  
  
<input type="text" placeholder="Your First Name" autofocus  
name="name" />  
  
</form>
```

HTML5 memberikan beberapa tambahan atribut untuk memudahkan pengembang dalam melakukan validasi input. Biasanya input akan divalidasi setelah data dikirim ke server jika menggunakan PHP, atau validasi secara client side menggunakan javascript. Namun HTML5 mulai menambah beberapa atribut sehingga akan menambah kemudahan dalam melakukan validasi. Contohnya:

```
<input type="email">
```

Input ini akan memastikan bahwa benar-benar suatu alamat email yang diinput oleh user.

```
<input type="password">
```

Input ini mengganti tampilan input menjadi lingkaran kecil sehingga menambah privacy user ketika mengetik password.

Kita juga dapat membuat input yang memiliki limit ke user

```
<input type="number" min="10" max="50" step="5" value="30">
```

Sejak HTML5 juga diperkenalkan input calendar dan waktu.

Calendar options:

```
<input type="date" name="date"/>
```

```
<input type="week" name="week"/>
```

```
<input type="month" name="month"/>
```

Time Options:

```
<input type="time" name="time"/>
```

```
<input type="datetime" name="dateTime"/>
```

```
<input type="datetime-local" name="localDateTime"/>
```

## Media Support

### Video

Penggunaan audio dan video pada internet semakin banyak seiring dengan semakin murah nya harga bandwidth dan penyimpanan. Banyak cara yang dapat dilakukan untuk memasang video pada website. Menggunakan software, file extension hingga browser extension. Namun sejak ada HTML5, diperkenalkan tag khusus untuk menampilkan audio dan video ke dalam halaman website. Tag video pada HTML5 memiliki beberapa fitur seperti autoplay, looping file, control, preload, error handling bahkan dapat menampilkan gambar saat video tidak sedang dimainkan.

Format video yang didukung oleh tag ini adalah OGG, contoh menampilkan video seperti berikut

```
<video src="ourfile.ogg" width="200" height="150" autoplay>
```

Dengan tag ini maka video akan langsung dimainkan saat halaman website sudah tampil. Untuk menambahkan fungsi kontrol dan display cover, kode tersebut dapat ditambah menjadi seperti berikut :

```
<video src="ourfile.ogg" width="200" height="150" autoplay  
controls preload="auto" poster="videoImage.png">
```



Sekarang video kita memiliki kontrol player dan display cover sebelum video dimainkan.

#### Audio

Hampir sama dengan tag yang digunakan pada video, kita dapat menambah audio dengan beberapa fungsi kontrol seperti berikut:

```
<audio src="ourAudio.ogg" autoplay controls preload="auto">
```



#### Tag HTML yang harus dihindari

Beberapa tag html sudah jarang digunakan karena membuat halaman website menjadi tidak responsive. Beberapa tag yang harus dihindari antara lain

`<b>`, `<i>`, `<strong>`

Huruf tebal dan huruf miring disarankan diatur menggunakan CSS saja. Penggunaan tag tersebut dalam halaman website akan membuat kerja dua kali jika terjadi modifikasi. Edit pada html dan edit pada CSS

<font>

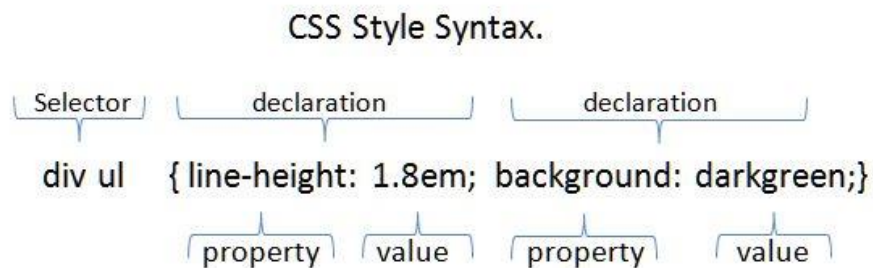
Penggunaan font lebh disarankan melalui CSS

<br>

Tag br mulai ditinggalkan karena tidak memberikan konteks apapun pada konten. Tag yang lebih cocok adalah <p>

## CSS

### Selector



Mengenal selector merupakan start yang baik untuk belajar CSS. Selector digunakan untuk mengidentifikasi item apa yang akan dikenai rule CSS. Selector sendiri dapat menunjuk suatu tag HTML atau struktur HTML seperti list, paragraf, input, atau class, nama element, atau ID yang kita buat. Penulisan selector akan diikuti dengan tanda { }, penulisan kode CSS akan ditaruh diantara tanda { } tersebut. Salah satu yang harus diperhatikan adalah kita menggunakan tanda titik dua : bukan sama dengan = untuk memberikan nilai pada CSS

### Class

Class digunakan sebagai tanda kode CSS mana yang akan kita terapkan pada halaman. Penulisan class sebagai atribut tag html seperti berikut

`class="namaClass"`

ketika kita ingin mengakses nama class dalam CSS maka gunakan tanda titik . sebelum nama class tersebut

## ID

Atribut ID memiliki fungsi yang hampir sama dengan class namun memiliki dua perbedaan utama yaitu

ID menggunakan tanda pagar # bukan tanda titik .

ID hanya boleh digunakan satu kali dalam halaman, ID harus menunjukan suatu bagian yang unqi dalam halaman tersebut

Penulisan atribut ID dalam html seperti berikut

`id="ourNameHere"`

## Penerapan CSS

CSS dapat diterapkan langsung pada bagian awal kode HTML. Penulisan CSS dibuka dengan tag `<style>` diantara tag `<head>`.

```
<style>
```

```
ul {
```

```
list-style-type: none;
```

```
}
```

```
</style>
```

Dengan menaruh kode CSS diatas pada bagian head maka kita telah membuat css dengan tipe internal stylesheet. Internal stylesheet akan memuat semua kode css menjadi satu file dengan kode html. Kita juga dapat membuat external CSS. Penulisan kode CSS akan dipisah dengan kode HTML sehingga kita dapat memiliki lebih dari satu file CSS.

Untuk menghubungkan eksternal CSS kedalam halaman html kita maka kita perlu menulis tag seperti berikut

```
<link rel="style sheet" type="text/css" href="mystyle.css">
```

Di dalam tag <link> diatas kita memiliki tag rel yang berfungsi memberitahu jenis file yang dituju. Selanjutnya ada tag type yang menunjukkan tipe file yang dituju yaitu text/css. Terakhir ada tag href yang menunjukkan lokasi file css kita.

Metode selanjutnya dalam penerapan kode css adalah inline css. kode CSS dalam inline CSS akan langsung ditulis dalam tag html yang ingin kita modifikasi.

```
<p style="color:red;">Our now RED paragraph!</p>
```

Karena langsung ditulis pada tag HTMLnya maka kita tidak perlu menulis selector atau tanda { } dalam kode CSS kita. Beberapa hal lain yang dapat kita gunakan untuk mempermudah penulisan CSS adalah tanda \*. Tanda \* digunakan jika css yang kita buat akan diterapkan pada semuanya di halaman tersebut

```
*{
```

```
Color : blue;
```

```
}
```

Kode diatas akan membuat semua warna pada halaman tersebut menjadi biru. Sedangkan kode berikut ini akan membuat semua text yang berada dalam tag footer menjadi biru

```
footer *{
```

```
Color : blue;
```

```
}
```

Kita juga dapat menargetkan secara lebih spesifik dengan menggunakan tag ID. Kode berikut ini memiliki dua div dengan nama todo dan completed.

```
<div id="todo">
```

```
<ul>
```

```
<li>Check the mail</li>
```



```
<li>Go to the store</li>

<li>Take the dog to the vet </li>
```

```
</ul>
```

```
</div>
```

```
<div id="completed">
```

```
<ul>
```

```
<li>Rent a movie</li>
```

```
<li>Make grocery list</li>
```

```
<li>Make vet appointment</li>
```

```
</ul>
```

```
</div>
```

- Check the mail
- Go to the store
- Take the dog to the vet
- Rent a movie
- Make grocery list
- Make vet appointment

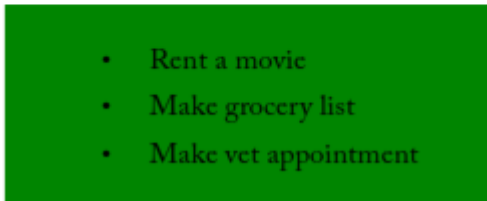
Dengan membuat kode CSS seperti berikut ini maka kita membuat div todo menjadi merah dan div completed menjadi hijau

```
<style>
```

```
#todo { background-color:red; }
```

```
#completed { background-color: green; }
```

</style>



Kita dapat menerapkan kode diatas pada semua tag yang memiliki ID sebagai atributnya. Namun jika ingin menerapkan kode tersebut pada beberapa tempat maka lebih baik menggunakan class. Kode sebelumnya akan kita modifikasi menjadi seperti berikut

<style>

.todo { background-color:red; }

.completed { background-color: green; }

</style>

<div id="today" class="todo">

<ul>

<li>Check the mail</li>

<li> Go to the store</li>

</ul>

</div>

<div id="tomorrow" class="todo">

<ul>

```

<li>Take the dog to the vet </li>

</ul>

</div>

<div id="done" class="completed">

<ul>

<li>Rent a movie</li>

<li>Make grocery list</li>

<li>Make vet appointment</li>

</ul>

</div>

```

- Check the mail
- Go to the store

- Take the dog to the vet

- Rent a movie
- Make grocery list
- Make vet appointment

Pada kode diatas penggunaan ID diubah class dengan mengganti tanda pagar # ke tanda titik . . kemudian pada tag todo list dibagi menjadi dua yaitu today dan tomorrow dan mengubah div completed menjadi done.

Hyperlink merupakan salah satu tag yang paling sering diberi kode CSS. Salah satu contoh kecil ketika ada kursor yang berada diatas link biasanya warna link tersebut akan berubah.

Efek ini dinamakan hover. Untuk membuat efek hover pada hyperlink maka kita dapat membuat kode css seperti berikut:

```
a:link {  
  
color:blue;  
  
}  
  
a:visited {  
  
color:grey;  
  
}  
  
a:hover{  
  
color:red;  
  
font-weight:bold;  
  
}
```

Kode diatas akan membuat suatu link berwarna biru, kemudian jika kita mendekatkan kursor ke link tersebut warna akan berubah menjadi merah. Jika kita klik link tersebut maka warna akan menjadi abu-abus.

Selain hover kita juga dapat menggunakan :before dan :after. Selector ini membuat kode css diterapkan diatas atau dibawah suatu element. Misal kita akan membuat sebuah judul kecil diatas to do element kita maka kode css menjadi seperti berikut :

```
.todo:before{  
  
content:"To Do:";  
  
background-color:yellow;  
  
color:red;  
  
font-weight:bold;
```

```
}
```

Setelah ditambahkan beberapa css maka kode kita menjadi seperti berikut :

w look like this:

```
<style>
```

```
.todo {
```

```
background-color:red;
```

```
}
```

```
.completed {
```

```
background-color: green;
```

```
}
```

```
.completed ul {
```

```
text-decoration: line-through;
```

```
div:not(.completed){
```

```
font-size:larger;
```

```
a:link {
```

```
color:blue;
```

```
a:visited {
```

```
color:grey;
```

```
a:hover{
```

```
color:red;
```

```
font-weight:bold;
```

```
.todo:before{
```

```
content:"To Do:";

background-color:yellow;

color:red;

font-weight:bold;

</style>

<div id="today" class="todo">

<ul>

<li>Check the mail</li>

<li> Go to the store</li>

</ul>

</div>

<div id="tomorrow" class="todo">

<ul>

<li>Take the dog to the

vet </li>

</ul>

</div>

<div id="yesterday"

class="completed">

<ul>

<li>Rent a movie</li>

</ul>
```

</div>

<div id="lastWeek"

class="completed">

<ul>

<li>Make grocery list</

<li>

<a href="http://www.dunkirkanimalclinic.com/"><li>Make

vet appointment</li></a>

</ul>

</div>

#### To Do:

- Check the mail
- Go to the store

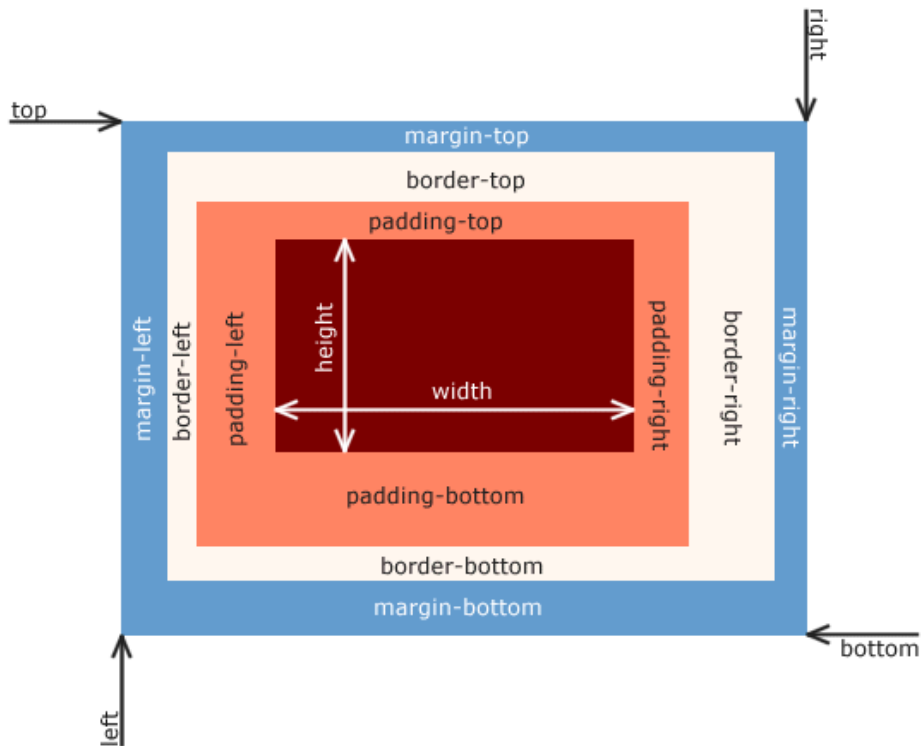
#### To Do:

- Take the dog to the vet

- Rent a movie

- ~~Make grocery list~~
- ~~Make vet appointment~~

## Box Model



### CSS Box

Sebagai pembatas suatu konten kita bisa menggunakan border. Border sendiri dapat dipasang pada bagian atas, bawah, kanan dan kiri. Border juga memiliki beberapa macam style seperti solid, double, grooved, dotted dan dash line

Contoh penggunaan border seperti berikut

<style>

p.one {

border-style:solid;

border-width:5px;

}

p.two {

border-style:groove;



```
border-width:medium;
```

```
}
```

```
p.three {
```

```
border-style:dotted;
```

```
border-width:1px;
```

```
border-color:red;
```


```
}
```

```
</style>
```


```
<p class="one">Some text.</p>
```

```
<p class="two">Some more text.</p>
```


```
<p class="three">Even more text.</p>
```



Some text.



Some more text.



Even more text.

Value	Deskripsi
None	Tidak ada border
Dotted	Border titik
Dashed	Border garis putus putus
Solid	Border dengan garis tebal
Groove	Border 3D

## Margin dan Padding

Konsep margin dan padding digunakan untuk memberikan ruang diantara konten dan batas konten tersebut. Padding digunakan untuk memberikan jarak di dalam batas konten, sebagai contoh jarak antara text dan sebuah paragraf

```
<style>

p.one {

border-style:solid;

border-width:5px;

}

p.two {

border-style:groove;

border-width:medium;

}

p.three {

border-style:dotted;

border-width:1px;

border-color:red;

}

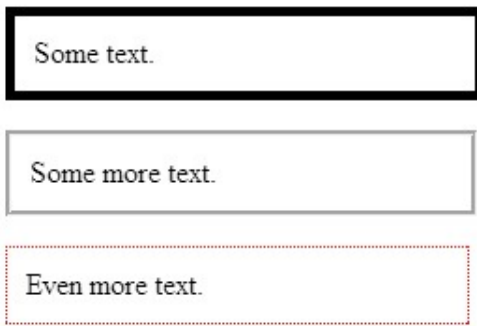
p {padding:10px;}

</style>

<p class="one">Some text.</p>

<p class="two">Some more text.</p>

<p class="three">Even more text.</p>
```



Sedangkan margin mengatur jarak yang ada diluar batas konten, jika kita menambah margin pada kode diatas maka penampakan hasilnya akan seperti berikut:

```
<style>

p.one {

border-style:solid;

border-width:5px;

}

p.two {

border-style:groove;

border-width:medium;

}

p.three {

border-style:dotted;

border-width:1px;

border-color:red;

}

p {padding:10px;}
```

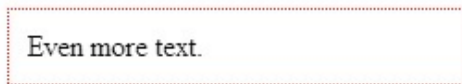
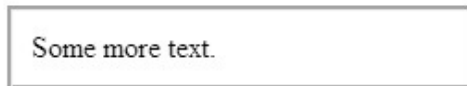
```
p {margin:50px;}
```

```
</style>
```

```
<p class="one">Some text.</p>
```

```
<p class="two">Some more text.</p>
```

```
<p class="three">Even more text.</p>
```



Dua contoh diatas, pengaturan nilai border dan padding dapat emnggunakan pixel atau persen. Kita juga dapat mengatur jarak yang berbeda untuk tiap sisinya dengan menambah kode untuk top, bottom, left atau right

```
<style>
```

```
p.one {
```

```
border-style:solid;
```

```
border-width:5px;
```

```
}
```

```
p.two {
```

```
border-style:groove;
```

```
border-width:medium;
```

```
}
```

```
p.three{
```

```
border-style:dotted;
```

```
border-width:1px;
```

```
border-color:red;
```

```
}
```

```
p {padding-left:10px;}
```

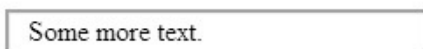
```
p {margin-bottom:50px;}
```

```
</style>
```

```
<p class="one">Some text.</p>
```

```
<p class="two">Some more text.</p>
```

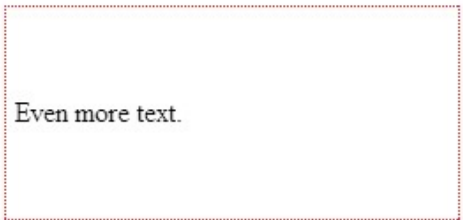
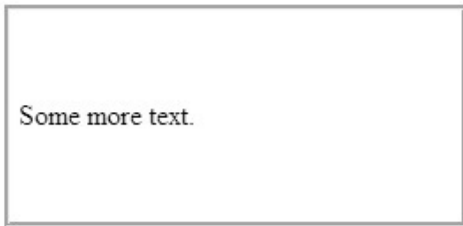
```
<p class="three">Even more text.</p>
```



Daripada membuat beberapa baris kode untuk mengatur tiap sisi, kita dapat menggabungkannya menjadi satu baris dengan menulis nilai dari tiap sisi sesuai dengan arah jarum jam, dimulai dari sisi top kemudian right, bottom left dan terakhir right

```
<style>
```

```
p.one {  
  
border-style:solid;  
  
border-width:5px;  
  
}  
  
p.two {  
  
border-style:groove;  
  
border-width:medium;  
  
}  
  
p.three {  
  
border-style:dotted;  
  
border-width:1px;  
  
border-color:red;  
  
}  
  
p {padding:50px 30px 50px 5px;}  
  
p {margin:50px;}  
  
</style>  
  
<p class="one">Some text.</p>  
  
<p class="two">Some more text.</p>  
  
<p class="three">Even more text.</p>
```



## Background

Kita dapat membuat warna background yang berbeda dengan menulis kode warna yang diinginkan. Warna dapat ditulis sesuai dengan nama warna tersebut dalam bahasa inggris, bisa menggunakan kode hex atau menggunakan kode rgb

```
<style>
```

```
p.one {
```

```
border-style:solid;
```

```
border-width:5px;
```

```
Background-color: green;
```

```
}
```

```
p.two {
```

```
border-style:groove;
```

```
border-width:medium;

Background-color:#ff3355;

}

p.three {

border-style:dotted;

border-width:1px;

border-color:red;

background-color: rgb(33,66,99);

}

p {padding:50px 30px 50px 5px;}

p {margin:50px;}

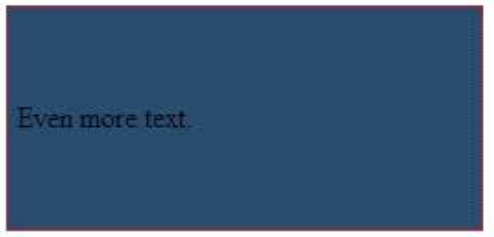
</style>

<p class="one">Some text.</p>

<p class="two">Some more text.</p>

<p class="three">Even more text.</p>
```





Hal yang menarik kita juga dapat menggunakan gambar sebagai background. Untuk menggunakan gambar sebagai background maka kita harus tahu alamat lokasi dari file gambar tersebut.

```
<style>
```

```
p.one {
```

```
border-style:solid;
```

```
border-width:5px;
```

```
background:url(clouds.jpg);
```

```
}
```

```
p.two {
```

```
border-style:groove;
```

```
border-width:medium;

background:url(calendar.jpg) no-repeat;

background-color:#ff3355;

}

p.three {

border-style:dotted;

border-width:1px;

border-color:red;

background:url(calendar.jpg);

}

p {padding:50px 30px 50px 5px;}

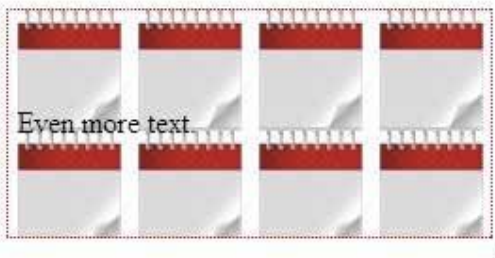
p {margin:50px;}

</style>

<p class="one">Some text.</p>

<p class="two">Some more text.</p>

<p class="three">Even more text.</p>
```



## Float

Float digunakan untuk membuat object css dapat melayang diantara object yang lain. Float hanya memiliki nilai right dan left. Ketika beberapa objek memiliki style float yang sama maka object tersebut akan menjadi sejajar selama tidak melewati batas dari kontainer yang ada pada objek tersebut. Analoginya seperti jika kita memiliki sebuah kotak yang besar. Didalam kotak tersebut terdapat kotak kecil yang disusun berjajar dari kiri ke kanan. Jika posisi sudah penuh maka kotak kecil selanjutnya akan membuat baris baru dibawah kotak kecil diatasnya.

Float digunakan untuk membuat style yang responsif. Kotak yang berisi konten yang biasanya cukup pada layar besar ketika dilihat menggunakan layar yang lebih kecil maka akan membuat satu baris baru dibawahnya. Kode berikut ini merupakan penerapan float:

```
<style>
```

```
.thumbnail {
```

```

float:left;

width:80px;

height:80px;

margin:5px;

}

</style>

<div>



















</div>

```

### Positioning

CSS membuat kita dapat mengontrol lokasi dari suatu element pada halaman website. Ada dua macam metode yang dapat digunakan sebagai deklarasi posisi yaitu fixed dan realtive.

Fixed digunakan jika kita jarak yang kita buat tidak berubah, sedangkan relative akan membuat jarak yang bisa diubah

```
<style>
```

```
h2 {
```

```
position:absolute;
```

```
left:50px;
```

```
top:100px;
```

```
}
```

```
</style>
```

```
<h2>This is a heading with an absolute position</h2>
```

```
<p>With absolute positioning, an element can be placed anywhere on a page. The heading below is placed 50px from the left of the page and 100px from the top of the page. As this
```

Text continues, you will see that the heading sits on top of the text as if it was not even there. As this text continues, you will see that the heading sits on top of the text as if it was not even there. As this text continues, you will see that the heading sits on top of the text as if it was not even there. As this text continues, you will see that the heading sits on top of the text as if it was not even there. As this text continues, you will see that the heading sits on top of

The text as if it was not even there. As this text continues, you will see that the heading sits on top of the text as if it was not even there. As this text continues, you will see that

the heading sits on top of the text as if it was not even there. As this text continues, you will see that the heading sits on top of the text as if it was not even there.</p>

With absolute positioning, an element can be placed anywhere on a page. The heading below is placed 50px from the left of the page and 100px from the top of the page. As this text continues, you will see that the heading sits on top of the text as if it weren't even there. As this text continues, you will see that the heading sits on top of the text as if it weren't even there. As this text continues, you will see that the heading sits on top of the text as if it weren't even there. As this text continues, you will see that the heading sits on top of the text as if it weren't even there. As this text continues, you will see that the heading sits on top of the text as if it weren't even there. As this text continues, you will see that the heading sits on top of the text as if it weren't even there. As this text continues, you will see that the heading sits on top of the text as if it weren't even there. As this text continues, you will see that the heading sits on top of the text as if it weren't even there. As this text continues, you will see that the heading sits on top of the text as if it weren't even there.

## This is a heading with an absolute position

Jika kita ingin mengubah letak judul heading menjadi normal atau berada diatas paragraf kita dapat membuat posisi heading tersebut menjadi relative

```
<style>
```

```
h2.pos_left {
```

```
position:relative;
```

```
left:-20px;
```

```
}
```

```
h2.pos_right {
```

```
position:relative;
```

```
left:20px;
```

```
}
```

```
</style>
```

```
<h2>This is a heading with no position</h2>
```

```
<h2 class="pos_left">This heading is moved left according to
```

```
its normal position</h2>
```

```
<h2 class="pos_right">This heading is moved right according to its normal position</h2>
```

<p>This is our paragraph that has a heading with relative positioning. Unless we put a negative bottom offset on our heading large enough to cover it, it will stay above the paragraph this time. </p>

**This is a heading with no position**

**his heading is moved left according to its normal position**

**This heading is moved right according to its normal position**

This is our paragraph that has a heading with relative positioning. Unless we put a negative bottom offset on our heading large enough to cover it, it will stay above the paragraph this time.

## Z-Index

z-index dari suatu element menandakan susunan tumpukan elemen tersebut dalam suatu halaman. Semakin besar nilai z-indexnya maka objek tersebut akan berada semakin diatas. Background suatu halaman biasanya memiliki posisi terendah dalam suatu halaman. Text biasanya akan berada di atas layer background tersebut. Kode berikut ini akan membuat posisi gambar memiliki nilai z-index minus 1

```
<style>
```

```
img {
```

```
position:absolute;
```

```
left:0px;
```

```
top:0px;
```

```
z-index:-1;
```

```
}
```

```
</style>
```

```
<h1>Here is some text</h1>
```

```




Sedangkan kode dibawah ini, gambar akan memiliki nilai z-index 1

```
<style>
```

```
img {
```

```
position:absolute;
```

```
left:0px;
```

```
top:0px;
```

```
z-index:1;
```

```
}
```

```
</style>
```

```
<h1>Here is some text</h1>
```

```

```





### Font dan dekorasi text

Saat kita memulai belajar HTML, biasanya kita akan menulis kata hello world dengan satu heading tag seperti h1 atau h2. Kita dapat mengubah warna dan background dari heading tersebut menjadi seperti berikut

```
<style>

h1{

color:red;

background-color:yellow;

}

</style>

<h1>This is an H1 heading</h1>
```

**This is an H1 heading**

Kita juga dapat mengubah jenis font dan ukuran text tersebut. Kode berikut akan membuat jenis font menjadi miring dan ukuran font menjadi 20 px

```
<style>

h1{

color:red;

background-color:yellow;

font-style:italic;

font-size:20px;

}

</style>
```

<h1>This is an H1 heading</h1>

*This is an H1 heading*

Beberapa metode lain dapat digunakan untuk memperindah tampilan seperti penggunaan font-variant

<style>

h1{

color:red;

background-color:yellow;

font-variant:small-caps;

}

</style>

<h1>This is an H1 heading</h1>

**THIS IS AN H1 HEADING**

### Text Styles

Kita dapat menambah jarak pada awal paragraf dengan menggunakan text indent , selain itu kita juga dapat mengatur posisi text apakah menjadi rata kiri, rata kanan atau center

<style>

p {

text-indent:15px;

text-align:justify;

}

</style>

<p>This is our paragraph for demonstrating some of the things we can do to text and font through the use of CSS. This is our paragraph for demonstrating some of the things we can do to text and font through the use of CSS. This is our paragraph for demonstrating some of the things we can do to text and font through the use of CSS. </p>

This is our paragraph for demonstrating some of the things we can do to text and font through the use of CSS. This is our paragraph for demonstrating some of the things we can do to text and font through the use of CSS. This is our paragraph for demonstrating some of the things we can do to text and font through the use of CSS.

Selain itu kita juga dapat menambah beberapa efek seperti garis bawah, garis tengah atau menghilangkan garis bawah pada suatu link

<style>

.strikeOut{text-decoration:line-through;}

.titles{text-decoration:underline;}

a {text-decoration:none;}

</style>

<span class="strikeOut">Text we want crossed out</span><br/>

<span class="titles">Hitchiker's Guide to the Galaxy</span><br/>

<span><a href="">A link with no underline</span>

~~Text we want crossed out~~  
Hitchiker's Guide to the Galaxy  
A link with no underline

## Anchor

Jika pada bab sebelumnya kita dapat menghilangkan garis bawah pada suatu link, maka kita juga dapat mendefinisikan apa yang terjadi pada suatu link dengan menggunakan CSS. Kita

dapat mengubah warna link, warna background, mengubah text dan ukuran font dengan menggunakan css seperti pada kode dibawah

```
<style>

a:link {color:#FF0000; background-color:yellow;} /* unvisited link */

a:visited

{color:#00FF00; background-color:orange;} /* visited link */

a:hover

{color:#FF00FF; background-color:green;} /* mouse over link */

a:active {color:#0000FF; background-color:white;} /* selected link */

</style>

<a href="" target="_blank">Here is our fake link!</a>
```

### Responsive Styling

Responsive design merupakan konsep dimana suatu tampilan dapat beradaptasi sesuai dengan ukuran layar device pengguna. Ukuran layar secara garis besar terbagi menjadi empat bagian yaitu

- Smart phones (portrait)—320 pixels
- Tablets (portrait)—768 pixels
- Tablets (landscape), netbooks—1024 pix
- Laptops, desktops—1024 atau lebih

Agar website kita menjadi lebih responsive maka kita dapat menggunakan fitur media queries dari CSS. Langkah pertama kita akan membuat suatu css yang dapat beradaptasi sesuai ukuran layar tersebut. Namun beberapa browser seperti safari akan membuat ukuran layar menjadi ukuran penuh satu layar. Untuk mengatasi masalah tersebut maka kita menggunakan tag HTML seperti berikut

```
<meta name="viewport" content="width=device-width; initial-scale=1.0">
```

Baris ini akan memberitahu browser untuk menggunakan ukuran layar asli dari device yang ada.

Selanjutnya kita akan membuat beberapa selector sebagai testing bagi empat tipe layar yang disebutkan diatas

```
@media only screen and (max-width: 959px) { }
```

```
// Smaller than desktop/laptop (tablets, netbooks)
```

```
@media only screen and (min-width: 768px) and (max-width: 959px) { }
```

```
//Portrait or landscape tablets, netbooks
```

```
@media only screen and (max-width: 767px) { }
```

```
// devices smaller than portrait tablets (mobile)
```

```
@media only screen and (min-width: 480px) and (max-width: 767px) { }
```

```
// mobile landscape to tablet portrait
```

```
@media only screen and (max-width: 479px) { }
```

```
// mobile up to landscape mode
```

Langkah selanjutnya kita akan membuat layout halaman website seperti berikut

Logo		
Link 1 Link 2 Link 3	Flash ad Content	An ad Another ad
Contact information		

Kode dari layout diatas seperti berikut

HTML

```
<header></header>
```

```
<div id="container">
```

```
<div id="left">
```

```
<ul>
```

```
<li><a href="">Link 1</a></li>
```

```
<li><a href="">Link 2</a></li>
```

```
<li><a href="">Link 3</a></li>
```

```
</ul>
```

```
</div>
```

```
<div id="main">
```

```
<div id="splash">[flash with id
```

```
"video"]</div>
```

```
<div id="content">Our text content
```

```
here</div>
```

```
</div>
```

```
<div id="right">
```

```
<div class="ad">an ad here</div>
```

```
<div class="ad">another ad </div>
```

```
</div>
```

```
<br style="clear: left;" />
```

```
</div>
```

```
<footer>Contact Information</footer>
```

## CSS

<style>

div{

border-style:dashed;

border-width:1px;

}

header{

background:url(ourlogo.png)

no-repeat;

background-size:200px 100px;

background-position:center;

height:100px;

}

#left{

min-width:100px;

max-width:150px;

float:left;

}

#main{

float:left;

}

#right{

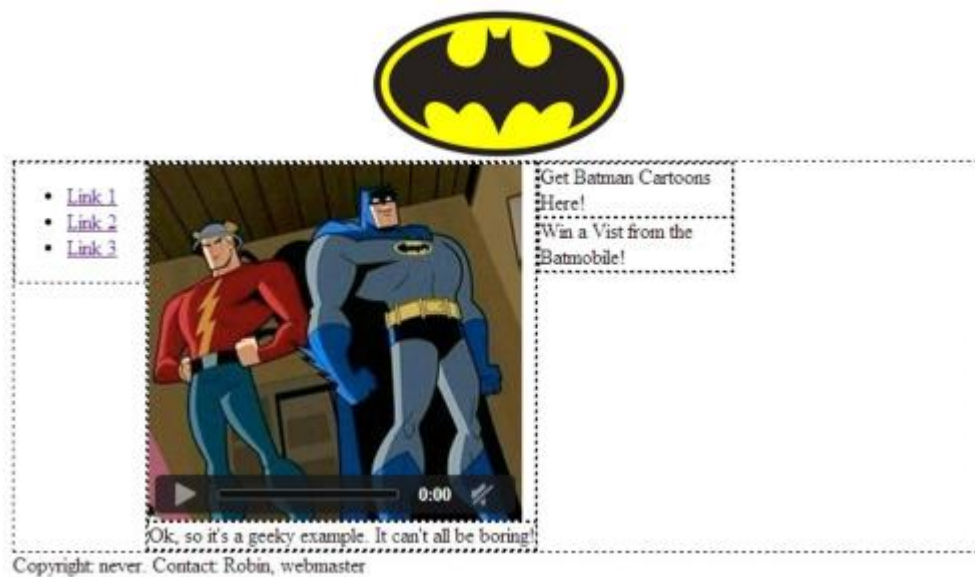
```
min-width:100px;
```

```
max-width:150px;
```

```
float:left;
```

```
}
```

```
</style>
```



Untuk beradaptasi pada layar yang kecil maka logo dan video akan kita sembuhkan dengan menggunakan kode berikut

```
<header></header>
```

```
<div id="container">
```

```
<div id="left">
```

```
<ul>
```

```
<li><a href="">Link 1</a></li>
```

```
<li><a href="">Link 2</a></li>
```



```

<li><a href="">Link 3</a></li>

</ul>

</div>

<div id="main">

<div id="splash">[flash width id "video"]</div>

<div id="content">Our text content here</div>

</div>

<div id="right">

<div class="ad">an ad here</div>

<div class="ad">another ad </div>

</div>

<br style="clear: left;" />

</div>

<footer>Contact Information</footer>

CSS

<style>

div{

border-style:dashed;

border-width: 1px;

}

header{

background:url(ourlogo.png) no-repeat;

```

```
background-size:200px 100px;

background-position:center;

height:100px;

}

#left{

min-width:100px;

max-width:150px;

float:left;

}

#main{

float:left;

}

#right{

min-width:100px;

max-width:150px;

float:left;

}

@media only screen and (max-width: 479px){

video{ display:none; }

#left{

width:100%;

float:none;
```

```
}  
  
#main{  
  
width:100%;  
  
float:none;  
  
}  
  
#right{  
  
width:100%;  
  
float:none;  
  
}  
  
#header{  
  
background:url(http://www.vectortemplates.com/raster/  
batman-logo-big.gif) no-repeat;  
  
background-size:100px 60px;  
  
background-position:center;  
  
height:100px;  
  
}  
  
}  
  
</style>
```



- [Link 1](#)
- [Link 2](#)
- [Link 3](#)

Ok, so it's a geeky example. It can't all be boring!

Get Batman Cartoons Here!

Win a Visit from the Batmobile!

Copyright never. Contact Robin, webmaster