**BAB I**

**PENDAHULUAN**

* + 1. **Landasan Teori**
    2. pengertian pemrograman web

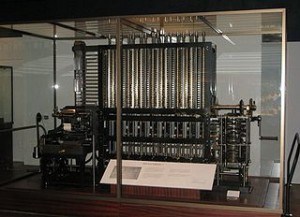
Pemrograman web diambil dari 2 suku kata yaitu pemrograman dan web. Pemrograman yang dalam bahasa English adalah programming dan diartikan proses, cara, perbuatan program.

Sedangkan web merupakan jaringan komputer yang terdiri dari kumpulan situs internet yang menawarkan teks dan grafik dan suara dan sumber daya animasi melalui protokol transfer hypertext.

Pandangan para ahli mengenai pengertian pemrograman adalah sebagai berikut:

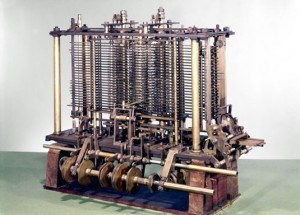
1. Menurut Amikom Yogyakarta, Program ialah kumpulan instruksi komputer, sedangkan metode dan juga tahapan sistematis dalam program ialah algoritma.
2. Ema Utami mengatakan Program ialah bahasa pemrograman.
3. Menurut Sukrisno Program merupakan kata, ekspresi, /pernyataan yang disusun dan dirangkai menjadi satu kesatuan prosedur, yang berupa urutan langkah, untuk menyelesaikan masalah yang diimplementasikan dengan menggunakan bahasa pemrograman sehingga bisa dieksesuksi oleh komputer.
4. Menurut Anwar Harjono Program ialah urutan instruksi untuk menjalankan suatu komputasi.
5. Saifuddin Anshari, menurutnya Program merupakan daftar terinci mengenai acara dan juga usaha yang akan dilaksanakan.
6. Sunarto, S.Kom mengatakan Program ialah sekumpulan instruksi yang diwujudkan dalam bentuk bahasa, kode skema, ataupun bentuk laain, yang jika digabungkan dengan media yang bisa dibaca dengan komputer akan mampu membuat komputer bekerja untuk melakukan fungsi-fungsi khusus, termasuk persiapan dalam merancaang instruksi-instruksi tersebut.
7. Wiryanto Dewobroto, menurutnya Program ialah hasil penyusunan detail langkah-langkah solusi (algoritma) masalah tersebut.
8. Menurut Sindhunata Program merupakan kelompok pernyataan yang persis dan juga berurutan yang gunanya ialah untuk memberi tahu komputer bagaimana melaksanakan sesuatu pekerjaan.
9. Agoeng Widyatmoko mendefinisikan Prograam sebagai aplikasi hasil analisis pemecahan masalah yang dibuat dalam bentuk program komputer.
10. Janner Simarmata mengatakan Program adalah suatu aplikasi yang dibuat dengan menggunakan bahasa program tertentu dan telah terinstal di dalaam komputer.
    * 1. sejarah pemrograman komputer

Pada tahun 1822, Charles Babbage seorang mahasiswa di Universitas Cambridge Inggris mengembangkan sebuah mesin untuk mengelola data-data agar mudah digunakan, mesin tersebut diberi nama ‘Difference Engine’.



Gambar 1.1.2.1 *Difference Engine*

Setelah bekerja selama 10 tahun pada mesinnya, Charles Babbage menyadari bahwa mesin yang dia ciptakan ini merupakan sebuah mesin yang bersifat *single-purpose machine* artinya hanya bisa menghasilkan satu jenis keluaran (*output*). Selanjutnya ia mengembangkan mesin lain yang bersifat multi-purpose. Mesin ini diberi nama ‘*Analytical Engine*’. Pekerjaan untuk membuat ‘*Analytical Engine*’ ini ia lakukan sampai dengan tahun 1842.



Gambar 1.1.2.2 *Analytical Engine*

Pada tahun 1847, Charles Babbage kembali menyempurnakan ‘*Difference Engine*’ hingga pada tahun 1849 ia berhasil membuat versi keduanya. Pekerjaan menyempurnakan hasil-hasil karyanya terus ia lakukan, bahkan dilanjutkan oleh anaknya, Henry Prevost. Charles Babbage sendiri meninggal pada tahun 1871. Untuk melindungi karya-karya ayahnya, Henry Prevost membuat beberapa kopian unit perhitungan aritmatika sederhana dari mesin yang dihasilkan ayahnya dan mengirimkannya ke beberapa institusi di dunia, termasuk ke Universitas Harvard.

Perkembangan dunia komputasi berlanjut pada tahun 1854, ketika seseorang bernama Charles Boole berhasil menciptakan sebuah sistem logika simbolik yang diberinama Logika Boole. Sistem ini mencakup pula logika untuk menyatakan hubungan lebih besar, lebih kecil, sama dengan dan tidak sama dengan. Sistem logika ini masih digunakan sampai dengan saat ini.

Pada tahun 1890, Amerika Serikat ingin melakukan sensus penduduk. Namun kendala yang muncul adalah keterbatasan alat yang ada pada waktu itu, mengingat jumlah penduduk yang semakin meningkat setiap tahunnya, maka diadakanlah sebuah kompetisi komputasi untuk mencari solusinya. Kompetisi ini dimenangkan oleh Herman Hollerith, yang akhirnya ia mendirikan sebuah perusahaan Hollerith Tabulating, Co. yang akhirnya berubah nama menjadi CTR (Calculating Tabulating Recording Company) setelah 3 perusahan lain ikut bergabung. Sepuluh tahun berikutnya perusahaan ini berganti nama lagi menjadi IBM (International Business Machine) hingga saat ini.

Selanjutnya perkembangan komputasi digital mulai berjalan pelan dan jarang digunakan dalam dunia bisnis sampai dengan pertengahan tahun 1920-an. Hingga pada tahun 1925, MIT (Massachusette Institute of Technology) mengembangkan sebuah mesin yang mampu menganalisis perhitungan differensiasi dan integrasi. Mesin yang didanai oleh Yayasan Rockefeller ini dapat dikatakan sebagai komputer terbesar di dunia pada tahun 1930.

Pada tahun 1935, seorang ilmuan Jerman bernama Konrad Zuse mengembangkan komputer Z-1, komputer inilah yang menjadi awal mula diterapkannya sistem biner dalam kinerjanya. Selain itu, Zuse juga berjasa dalam komputasi komputer digital ketika ia menciptakan bahasa pemrograman komputer pertama ‘Plankalkul’.

Pada tahun 1945, terjadi pula peristiwa penting dalam sejarah perkembangan komputasi komputer digital yaitu ketika terjadi kerusakan pada mesin Mark II yang ada di Universitas Harvard. Seseorang yang bernama Grace Murray Hopper yang mengetahui hal ini langsung menyelidiki sebab kerusakannya. Akhirnya dia menemukan seekor ngengat yang terjebak dalam mesin tersebut. Dalam catatan hariannya, Hopper menuliskan: “First actual case of bug being found”. Dia menyebut ngengat ini sebagai sebuah kutu busuk (*bug*), selanjutnya kata ‘bug’ ini sering digunakan untuk menunjukkan adanya ketidakberesan dalam program. Dari kata ‘bug’ ini muncul pula istilah ‘debugging’ yang artinya proses pembetulan kesalahan program.

Pada tahun 1954, IBM mulai mengembangkan bahasa pemrograman FORTRAN (FORmula TRANslator). Bahasa FORTRAN merupakan bahasa pemrograman level tinggi pertama yang dikomersialkan. Pemrograman level tinggi maksudnya adalah perintah atau kodenya mudah dibaca dan dipahami oleh manusia.

Pada tahun 1958, FORTRAN II dan ALGOL dipublikasikan bersamaan dengan diluncurkannya LISP. Sedangkan pada tahun 1959, bahasa pemrograman COBOL juga diluncurkan. Sejak saat itu perkembangan bahasa pemrograman berkembang sangat cepat.

Pada tahun 1970, bahasa PASCAL mulai dipublikasikan dan hingga saat ini masih banyak digunakan untuk keperluan pendidikan. Selain itu muncul pula dua bahasa pemrograman yang dianggap sangat penting yaitu SMALLTALK dan B-Languange. SMALLTALK penting karena merupakan bahasa pemrograman berbasis obyek yang pertama. Sedangkan B-Languange dikatakan penting karena merupakan cikal bakal munculnya bahasa C. Dengan bahasa C, pemrograman akan lebih mudah, efisien, dan fleksibel.

Pada tahun 1975, Dr. Wong merilis bahasa pemrograman hasil ciptaannya bernama TinyBASIC. TinyBASIC merupakan bahasa pemrograman pertama yang bersifat free alias tidak membayar dalam penggunaannya. Pada tahun yang sama, Bill Gates dan Paul Allen juga membuat bahasa pemrograman yang diberi nama BASIC. BASIC ini selanjutnya mereka jual ke MIT.

Bahasa pemrograman terus berkembang demikian pesat hingga saat ini. Hal ini ditandai dengan semakin banyaknya bahasa pemrograman yang bermunculan. Setiap bahasa pemrograman memiliki kelebihan dan kekurangan, tetapi semua bahasa pemrograman berjalan atas dasar logika dan algoritma sehingga kedua hal inilah yang harus diasah lebih dulu jika ingin mempelajari bahasa pemrograman. Sebaiknya fokuslah kepada sebuah bahasa pemrograman hingga dapat menguasainya dengan baik, kemudian lanjutkan dengan mempelajari bahasa pemrograman yang lain untuk menambah wawasan.

* + 1. Sejarah dan perkembangan pemrograman css

CSS (Cascading Style Sheet) adalah standar pembuatan dan pemakaian style untuk dokumen terstruktur. CSS digunakan untuk mempersingkat penulisan tag HTML seperti font, color, text,  dan table menjadi lebih ringkas sehingga tidak terjadi pengulangan tulisan. Penggunaan yang paling umum dari CSS adalah untuk memformat halaman web yang ditulis dengan HTML maupun XHTML. Meskipun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen XML termasuk juga SVG dan XUL. Spesifikasi CSS diatur oleh World Wide Web Consortium (W3C).

CSS pertama kali bermula sejak munculnya SGML pada tahun 1970an. Sejak kemunculannya tersebut, CSS mengalami perkembangan yang sangat pesat. Format dasar CSS yang banyak kita gunakan sekarang ini merupakan ide dari seoang programmer bernama Hakon Wium Lie yang tertuang dalam proposalnya mengenai Cascading HTML Style Sheet (CHSS) pada bulan Oktober 1994 (dalam konferensi W3C di Chicago, Illinois). Kemudian, beliau bersama-sama dengan seorang temannya yang bernama Bert Bos mengembangkan suatu standard CSS.

Pada akhir tahun 1996, CSS telah resmi dipublikasikan (dan menyusul kemudian CSS Level 1 pada bulan Desember). Pengerjaan proyek ini juga didukung oleh seorang programmer bernama Thomas Reardon dari perusahaan software ternama, Microsoft. CSS digunakan oleh penulis maupun pembaca halaman web untuk menentukan warna, jenis huruf, tata letak, dan berbagai aspek tampilan dokumen. CSS digunakan terutama untuk memisahkan antara isi dokumen (yang ditulis dengan HTML atau bahasa markup lainnya) dengan presentasi dokumen (yang ditulis dengan CSS).

Pemisahan ini dapat meningkatkan aksesibilitas isi, memberikan lebih banyak keleluasaan dan kontrol terhadap tampilan, dan mengurangi kompleksitas serta pengulangan pada stuktur isi. CSS memungkinkan halaman yang sama untuk ditampilkan dengan cara yang berbeda untuk metode presentasi yang berbeda, seperti melalui layar, cetak, suara (sewaktu dibacakan oleh browser basis-suara atau pembaca layar), dan juga alat pembaca braille. Halaman HTML atau XML yang sama juga dapat ditampilkan secara berbeda, baik dari segi gaya tampilan atau skema warna dengan menggunakan CSS.

Sekarang penggunaan CSS telah semakin meluas dan terus dikembangkan. Hal ini juga akan mempermudah seorang web designer dalam mengembangkan suatu halaman web (situs).

Ada 3 cara untuk memasang Script CSS pada dokumen HTML yaitu:

1. External Style Sheet (file CSS berbeda dari file HTML),
2. Internal Style Sheet (Kode CSS dipasang di dalam tag head HTML),
3. Inline Style Sheet (Kode CSS langsung dipasang di tag HTML, tidak direkomendasikan).
4. CSS level 1

Pada tanggal 17 Agustus 1996 World Wide Web Consortium (W3C) menetapkan CSS sebagai bahasa pemrograman standard dalam pembuatan web. Tujuannya adalah untuk mengurangi pembuatan tag-tag baru oleh Netscape dan Internet Explorer, karena kedua browser tersebut sedang bersaing mengembangkan tag sendiri untuk mengatur tampilan web. CSS 1 mendukung pengaturan tampilan dalam hal :

1. Font (Jenis ketebalan).
2. Warna, teks, background dan elemen lainnya.
3. Text attributes, misalnya spasi antar baris, kata dan  huruf.
4. Posisi teks, gambar, table dan elemen lainnya.
5. Margin, border dan padiing.
6. CSS level 2

W3C menyempurnakan CSS tahap awal dengan menciptakan standard CSS 2 yang menjadi standard hingga saat ini pada tahun1998. Semua atribut dari CSS 1 dimasukkan dan diperluas dengan penekanan pada International Accessibiality and Capacibilty kususnya media-specific CSS. CSS 2 dikembangkan untuk memenuhi kebutuhan terhadap format dokumen agar bisa ditampilkan di printer.

1. CSS level 3

CSS 3adalah versi terbaru dari CSS yang mampu melakukan banyak hal dalam mendesain website. CSS 3 dapat melakukan animasi pada halaman website, diantaranya animasi warna dan animasi 3D. Desainnya yang memudahkan dalam hal kompatibilitas websitenya pada smartphone dengan dukungan fitur baru yakni media query. Selain itu, banyak fitur baru pada CSS 3 yaitu : Multiple background, border-radius, drop-shadow, border-image, CSS-Math dan CSS Object Model.

Adapun fitur-fitur terbaru CSS 3 adalah sebagai berikut:

1. Animasi, sehingga pembuatan animasi tidak memerlukan program sejenis Adobe Flash dan Microsoft Silverlight.
2. Beberapa efek teks, seperti teks berbayang, kolom koran dan “Word-Wrap”.
3. Beberapa efek pada kotak, seperti kotak yang ukurannya dapat diubah-ubah, transformasi 2 dimensi dan 2 dimensi, sudut-sudut yang tumpul dan bayangan.
   * 1. Sejarah dan perkembangan HTML

Hypertext Markup Language (HTML) adalah bahasa yang digunakan untuk menulis halaman web. HTML merupakan pengembangan dari standar pemformatan dokumen teks yaitu Standard Generalized Markup Language (SGML). HTML sebenarnya adalah dokumen ASCII atau teks biasa, yang dirancang untuk tidak tergantung pada suatu sistem operasi tertentu.

HTML dibuat oleh Tim Berners-Lee ketika masih bekerja untuk CERN dan dipopulerkan pertama kali oleh browser Mosaic. Selama awal tahun 1990 HTML mengalami perkembangan yang sangat pesat. Setiap pengembangan HTML pasti akan menambahkan kemampuan dan fasilitas yang lebih baik dari versi sebelumnya. Namun perkembangan resmi dikeluarkan pada bulan November 1995 oleh IETF (Internet Engineering Task Force).

HTML 2.0 ini merupakan penyempurnaan dari HTML+ (1993).  
HTML 3.0 (1995) memberikan kemampuan lebih daripada versi sebelumnya. Sebuah usaha dari World Wide Web Consortium’s (W3C) HTML Working Group pada tahun 1996 menghasilkan HTML 3.2. HTML versi ini secara resmi diterbitkan pada bulan Januari 1997. HTML versi terbaru adalah HTML 4.01 yang dikeluarkan secara resmi oleh W3C pada tanggal 24 April 1998. HTML merupakan perbaikan dari HTML 4.0 yang lebih dulu diterbitkan. (18 Desember 1997).

1. Dasar HTML

Mendesain HTML berarti melakukan suatu tindakan pemrograman. Namun HTML bukanlah sebuah bahasa pemrograman. Namun HTML hanyalah berisi perintah-perintah yang telah terstruktur berupa tag-tag penyusun. Menuliskan tag-tag HTML tidaklah sebatas hanya memasukkan perintah-perintah tertentu agar HTML kita dapat di akses oleh browser. Mendesain HTML adalah adalah sebuah seni tersendiri. Homepage yang merupakan implementasi dari HTML adalah refleksi dari orang yang membuatnya. Untuk itu kita perlu mendesainnya dengan baik agar para pengunjung homepage yang kita buat merasa senang dan bermanfaat.

Mendesain HTML dapat dilakukan dengan dua cara:

1. Menggunakan HTML Editor, seperti Microsoft FrontPage, Adobe Dreamweaver, dan lain-lain. Dapatkan editor HTML lainnya disini.
2. Dengan cara menuliskan sendiri secara manual satu persatu tag-tag HTML ke dalam dokumen HTML.
3. Perkembangan HTML

Adapun perkembangan HTML adalah sebagai berikut :

1. HTML Versi 1.0

HTML versi ini merupakan versi pertama sejak lahirnya Bahasa HTML yang memiliki kemampuan untuk heading, paragraph, hypertext, manipulasi text (bold & italic) serta memiliki hubungan terhadap penggunaan sebuah gambar.

1. HTML Versi 2.0

HTML versi ini mulai dikembangkan pada tanggal 14 Januari 1995-1996 dengan beberapa kemampuan tambahan seperti penambahan form & hal ini menjadi cikal bakal adanya proses interaktif dengan pengguna. Mulai versi ini, HTML menjadi awal mula adanya website interaktif.

1. HTML Versi 3.0 dan 3.2

HTML versi ini lahir pada tanggal 18 Desember 1997 yang dikenal dengan HTML+ karena mempunyai beberapa fasilitas penambahan fitur table dalam paragraph, kemudian proses reasearch dilanjutkan sampai disempurnakan pada bulan Mei 1998.

1. HTML Versi 4.0

HTML versi ini lahir pada tanggal 24 Desember 1999 dengan penambahan beberapa fitur lagi seperti adanya link, image & lain-lain sebagai penyempurna dari HTML versi 3.2.

1. HTML Versi 5.0

HTML versi ini atau lebih dikenal dengan HTML5 lahir pada tahun 2008 yang menjadi standar baru untuk HTML, XHTML, & DOM HTML. HTML5 ini merupakan proyek kerjasama antara W3C dengan WHATWG (Web Hypertext Application Technology Working Group). HTML5 menambahkan beberapa perintah baru yang membantu para programmer web untuk mendesain website lebih menarik & interaktif.

* + 1. jenis- jenis website

Adapun jenis-jenis website dalam pemrograman adalah sebagai berikut

1. Website Statis

Website Statis adalah web yang konten atau isinya tidak di perbaharui dalam kurun waktu tertentu atau bisa di bilang isisnya tidak berubah sama sekali seperti web profile suatu perusahaan atau instansi pendidikan dll karena user disitu hanya dapat melihat saja dan website statis cenderung tidak responsive karena saat ingin memperbaharui webnya harus mengubah scriptnya secara langsung tidak dapat melalui CMS yang ada.

1. Website Dinamis

Website dinamis adalah web yang konten atau isinya sering kali di rubah karena user atau pengguna dapat mengupdate informasi website secara langsung biasanya di gunakan untuk toko online, web blog, jejaring sosial, dll. Pembuatan yang cukup lama di karenakan penggunaan database yang di perlukan untuk menyimpan data-data tersebut.

1. Website Interaktif

Website Interaktif adalah Website yang kurang lebih sama dengan website dinamis namun interaksi di dalamnya lebih sering karena website interaktif adalah website yang di peruntukkan untuk user atau pengguna yang ikut andil juga mengisi konten di dalamnya tidak seperti website dinamis yang hanya sebatas koment atau share doang, pada website interaktif seperti contoh youtube pengguna atau user ikut andil juga dalam mengisi konten di dalamnya.

1. Bahasa Pemrograman Web

  Banyak bahasa​​ pemrograman yang dapat di gunakan untuk membangun sebuah website seperti :

1. HTML (HyperText Markup Language)

HTML sendiri adalah bahasa markup yang digunakan untuk membuat sebuah halam website, menampilkan berbagai informasi di dalam sebuah penjelajah web internet dan penulisan​​ yang sederhana, dengan format penulisan <HTML> sebagai pembuka dan </HTML> sebagai penutup. Banyak contoh syntak seperti HEAD, BODY, TAG, dll pada tutorial-tutorial yang disediakan oleh buku, video, maupun di internet.

1. CSS (Cascading​​ Style Sheet)

Jika HTML merupakan sintak dasar pembentuk sebuah website maka CSS adalah bahasa yang mengatur sebuah website agar mempunyai tampilan yang menarik, fungsi dari CSS sendiri adalah memisahkan konten dari tampilan visualnya di situs.

Dengan CSS maka website dapat terlihat lebih rapi dan lebih menarik untuk dilihat karena CSS mengubah tampilan tesebut. cara penulisannya pun ada 3 cara :​​

1. *Internal*

Penulisan CSS​​ Internal​​ adalah cara penulisan CSS secara langsung dan di gabungkan kedalam HTML, CSS sendiri itu di tamping di dalam​​ sintak HEAD pada HTML.

1. *Eksternal*

Penulisan CSS eksternal termasuk penulisan CSS yang paling mudah karena file CSS ini di pisah dengan file HTML jadi anda tidak terlalu pusing dalam melihat sintaknya. untuk mengakses CSS ini​​ anda tinggal memanggil atau menload CSS melalui HTML dengan menulis​​ <link rel="stylesheet" href=​​ “tempat\_file\_css\_disimpan”>​​ di dalam sintak HEAD pada HTML.

*Inline*

Penulisan CSS​​ inline​​ ini termasuk yang paling jarang di gunakan karena penggunaanya yang hampir sama dengan​​ Internal​​ namun pada penulisannya di tulis langsung bersamaan di dalam sintaks HTML, cara penulisan ini sangat tidak disarankan untuk selalu di gunakan dan usahakan hanya di gunakan di saat-saat yang memang penggunaan internal CSS dan eksternal CSS tidak dapat memenuhi harapan kita.

1. PHP (PHP Hypertext Pre-Processor)

PHP adalah bahasa pemrograman Script server-side yang di design untuk pengembangan web. jika anda hanya menggunakan HTML dan CSS saja dalam membangun sebuah website maka website tersebut hanya akan menjadi website statis namun jika anda menyisipkan PHP di dalamnya maka website tersebut dapat menjadi website Dinamis karena penggunaan PHP yang dapat membuat sebuah kerangka website berubah-ubah menyesuaikan kondisi yang terjadi, sederhananya saat penggunakan layar smartphone maka tampilan website akan di sesuaikan agar terlihat bagus dan ​​ nyaman di layar smartphone, jika di gunakan pada layar tablet maka penerapan yang sama akan terjadi juga begitu seterusnya.

penggunaan PHP juga penting karena jika membuat website yang dinamis dan interaktif maka adanya PHP mempermudah kita dalam pembangunan dan pengembangan juga

* + 1. kelebihan dan kekurangan pemrograman CSS

Sama halnya dengan objek – objek lain yang ada dimuka bumi ini. Dalam penggunaanya, CSS juga tidak luput dari kelebihan dan kekurangan. Berikut beberapa kelebihan dan kekurangan CSS yang saya tahu. Berikut ini merupakan kelebihan dan kekurangan CSS :

1. Kelebihan CSS

Adapun kelebihan dari CSS (Cascading style sheets) adalah sebagai berikut :

1. CSS sangat mudah untuk dipelajari dan digunakan
2. CSS memisahkan antara Desain dan Konten Web/Blog
3. Pengaturan Desain dapat dilakukan seefisien mungkin
4. Karena satu css dapat dipakai beberapa kali ,maka dapat menghemat penulisan kode
5. Mempersingkat waktu kerja, baik saat membuat maupun saat modifikasi halaman Web/Blog
6. Kekurangan CSS :

CSS (Cascading style sheets) juga memiliki beberapa kelemahan yaitu sebagai berikut :

1. CSS sangat menyita waktu karena tidak semua browser mengartikan sintaks-sintaks CSS yang sama
2. desain sudah terlihat rapih di suatu browser namun ketika dicoba di browser lain malah jadi acak-acakan
3. Kekurangan tersebut dapat diatasi dengan script-script khusu(CSS Hack)
   * 1. Kelebihan dan kekurangan pemrograman HTML

Sama halnya dengan objek – objek lain yang ada dimuka bumi ini. Dalam penggunaanya, html juga tidak luput dari kelebihan dan kekurangan. Berikut beberapa kelebihan dan kekurangan html yang saya tahu. Berikut ini merupakan kelebihan dan kekurangan html:

1. Kelebihan-kelebihan HTML

Adapun kelebihan dari HTML (Hyper Text Markup Language) adalah sebagai berikut :

1. browser. Merupakan bahasa penkodean yang lintas platform (cross platform), maksudnya HTML dapat digunakan pada berbagai jenis mesin komputer yang berbeda dan berbagai macam sistem operasi yang berbeda. Jadi bersifat fleksibel karena ditulis cukup dengan menggunakan editor karakter ASCII.
2. Dapat disisipi gambar baik gambar statis atau dinamis (animasi) termasuk menggunakan gambar untuk dijadikan hyperlink. Gambar di sini digunakan untuk merujuk pada suatu halaman web, dimana setiap titik-titik yang sudah didefinisikan berupa rectangular (kotak), poligon (kurva tak beraturan) atau lingkaran digunakan untuk ‘jump’ ke halaman lain, atau link ke halaman di luar web yang bersangkutan.
3. Dapat disisipi animasi berupa Java Applet atau file-file animasi dari Macromedia Flash atau Macromedia Shockwave (untuk keperluan ini, browser harus memiliki plug-in khusus untuk menjalankan file-file animasi ini).
4. Dapat disisipi bahasa pemrograman untuk mempercantik halaman web seperti Javascript, VBScript, Active Server Pages, Perl, Tcl, PHP dan sebagainya.  
   Bukan merupakan bahasa pemrograman jadi tidak memerlukan kompiler. Cara menjalanakannya cukup dengan menggunakan
5. Adapun kekurangan dari HTML ini adalah :

HTML (Hyper Text Markup Language) juga memiliki beberapa kelemahan yaitu sebagai berikut :

1. Menghasilkan halaman yang statis, yang saya tahu untuk memperoleh halaman yang dinamis harus menggunakan bahasa pemrograman tertentu seperti Javascript atau VBScript dan animasi seperti Flash atau Shockwave.
2. Memiliki tag-tag yang begitu banyak sehingga susah dipelajari untuk yang masih awam.
3. Tidak dapat menghasilkan halaman yang interaktif. Interaktif di sini maksudnya client dapat berinteraksi dengan server. Untuk keperluan itu, HTML harus disisipi bahasa pemrograman yang dapat menangani hal tersebut, contohnya Perl atau Tcl.
   1. **Rumusan Masalah**

Adapun rumusan masalah yang ingin di capai dalam praktikum pemrograman web (html dan CSS) ini, yaitu :

1. Apa pengertian dari pemrograman web
2. Bagaimana struktur dasar dari pemrograman HTML
3. Apa fungsi dari pemrograman HTML dan CSS
4. Bagaiaman cara menjalankan program dengan menggunakan HTML dan CSS
5. Bagaimana Penerapan CSS pada HTML
   1. **Tujuan**

Adapun tujuan dilaksanakannya praktikum ini adalah sebagai berikut :

1. Untuk mengetahui apa pengertian dari pemrograman web
2. Untuk menegtahui bagaimana struktur dasar dari pemrograman HTML dan CSS
3. Untuk mengetahui apa fungsi dari pemrograman HTML dan CSS
4. Untuk mengetahui bagaiaman cara menjalankan program dengan menggunakan html dan CSS
5. Untuk mengetahui Bagaimana Penerapan CSS pada HTML
   1. **Manfaat**

Adapun tujuan dilaksanakannya praktikum ini adalah sebagai berikut :

1. Untuk memahami apa pengertian dari pemrograman web
2. Untuk memahami bagaimana struktur dasar dari pemrograman HTML dan CSS
3. Untuk memahami apa fungsi dari pemrograman HTML dan CSS
4. Untuk memahami bagaiaman cara menjalankan program dengan menggunakan HTML dan CSS
5. Untuk memahami bagaimana Penerapan CSS pada HTML

**BAB II**

**METODOLOGI PRAKTIKUM**

* 1. **Waktu dan Tempat Pelaksanaan**
     1. Waktu

Adapun waktu dan tempat pelaksanaan praktikum pemrograman web (html dan CSS) ini adalah pada hari Jumat, 22 November 2019 Pukul 13.30 – 14.30 WITA.

* + 1. Tempat

Adapun tempat pelaksanaan praktikum pemrograman web (html dan CSS) ini bertempat di laboratorium LaboratoriumArtificial Intelligencie and ComputerScienceTeknik Informatika, Fakultas Teknik, Universitas Halu Oleo.

* 1. **Alat dan Bahan**
     1. Alat

Adapun alat yang digunakan pada saat praktikum pemrograman web (html dan CSS) ini adalah sebagai berikut:

Tabel 2.1 Alat

|  |  |  |
| --- | --- | --- |
| No. | Nama Alat | Fungsi |
| 1 | Komputer | Sebagai media untuk mengaplikasikan aplikasi *Inkscape* |
| 2 | Proyektor | Sebagai media untuk menampilkan gambar dari komputer ke permukaan datar |
| 3 | Aplikasi *visual studio code* | Untuk mengembangkan aplikasi dalam kode asli (dalam bentuk Bahasa mesin yang berjalan di atas windows) atau kode terkelola (dalam bentuk microsoft intermediate language di atas. |

* + 1. Bahan

Adapun bahan yang digunakan pada saat praktikum Desain Grafis ini adalah sebagai berikut:

Tabel 2.2 Bahan

|  |  |  |
| --- | --- | --- |
| No. | Nama Bahan | Fungsi |
| 1 | Modul | Berisikan materi yang akan dibahas pada saat praktikum Aplikasi Perkantoran |

* 1. **Prosedur Kerja**

Adapun prosedur kerja atau langkah-langkah yang dilakukan dalam proses praktikum pemrograman webini adalah sebagai berikut:

1. Sebelum melakukan praktikum, Mahasiswa semester 1 Jurusan Teknik Informatika dibagi menjadi beberapa kelompok.
2. Sebelum melakukan praktikum, setiap Praktikan wajib mengerjakan tugas pendahuluan dan membawa ID *(identity) card* sebagai syarat masuk praktikum.
3. Setiap Praktikan wajib mempelajari modul sebelum melakukan praktikum.
4. Setiap kelompok menyiapkan alat praktikum (setidaknya 1 laptop yang telah ter-*insltall* aplikasi *Visual studio code*).
5. Selanjutnya, Asisten dosen membimbing dan memandu praktek pemrograman web (html dan CSS)
6. Kemudian Asisten dosen memberikan tugas membuat website dengan menggunakan pemrograman (html dan CSS). Mengambil dokumentasi pada saat praktikum.

**BAB III**

**PEMBAHASAN**

* 1. **Pengertian Pemrograman Web**

Pemrograman web diambil dari 2 suku kata yaitu pemrograman dan web.  
Pemrograman yang dalam bahasa English adalah programming dan diartikan proses, cara, perbuatan program (secara bahasa indonesia).

Web merupakan jaringan komputer yang terdiri dari kumpulan situs internet yang menawarkan teks dan grafik dan suara dan sumber daya animasi melalui protokol transfer hypertext.

Orang banyak mengenal web dengan istilah WWW (world wide web), World Wide Web adalah layanan internet yang paling populer saat ini internet mulai dikenal dan digunakan secara luas setelah adanya layanan WWW .

WWW adalah halaman-halaman website yang dapat saling terkoneksi satu dengan lainnya (hyperlink) yang membentuk samudra belantara informasi . WWW berjalan dengan protokol HyperText Transfer Protokol (HTTP). Halaman Web merupakan file teks murni (plain text) yang berisi sintaks-sintaks HTML yang dapat dibuka/dilihat/diterjemahkan dengan Internet Browser . Sintaks HTML mampu memuat konten text, gambar, audio, video dan animasi.

Kini internet identik dengan web, karena kepopuleran web sebagai standar interface pada lanyanan-layanan yang ada di internet, dari awalnya sebagai penyedia informasi, ini digunakan juga untuk komunikasi dari email sampai dengan chatting, sampai dengan melakukan transaksi binsin (commerce) (Betha Sidik,Ir, Pemrograman web dengan HTML, Informatika Bandung, 2005) Banyak keuntungan yang diberikan oleh Aplikasi berbasis Web dari pada aplikasi berbasis destop, sehingga aplikasi berbasis web telah diadopsi oleh perusahaan sebagai bagian dari strategi teknologi informasinya, karena beberapa alasan :

1. Akses informasi mudah,
2. Setup server lebih mudah
3. Informasi mudah didistribusikan
4. Bebas platform,
5. informasi dapat di sajikan oleh browser web pada sistem operasi mana saja karena adanya standar dokumen berbagai tipe data dapat disajikan
   1. **Stuktur Dasar HTML**

Suatu dokumen HTML dibangun berdasarkan suatu struktur. Struktur ini harus terdapat dalam *file* HTML. Apabila salah satu bagian dari struktur ini tidak tercantum dalam *file* HTML, maka *file* HTML tersebut tidak dapat dijalankan di web *browser*.

1. Tag

Tag adalah teks khusus (markup) berupa dua karakter "<" dan ">", sebagai

contoh <body> adalah tag dengan nama body.Tag harus ditulis secara berpasangan, yang terdiri atas tag pembuka dan tagpenutup (ditambahkan karakter "/" setelah karakter "<"),sebagai contoh <body> ini adalah tag pembuka isi dokumen HTML, dan</body> ini adalah tag penutup isi dokumen HTML.

1. Element

Element terdiri atas tiga bagian, yaitu tag pembuka, isi, dan tag penutup.Sebagai contoh untuk menampilkan judul dokumen HTML pada webbrowser digunakan element title, dimana:<title> ini adalah tag pembuka judul dokumen HTMLJudul Dokumen HTML ini adalah isi judul dari dokumen HTML</title> ini adalah tag penutup judul dokumen HTML

Tag-tag yang ditulis secara berpasangan pada suatu element HTML, tidak

boleh saling tumpang tindih dengan pasangan tag-tag lainnya.

1. Attribute

*Attribute* mendefinisikan *property* dari suatu element HTML, yang terdiri atasnama dan nilai.Secara umum nilai attribute harus berada dalam tanda petik satu atau dua.

1. Html

<html> (tag html) merupakan tag pembuka html yaitu awal dari dokumen html. html selalu memiliki tag pembuka dan tag penutup. Tag penutup html yaitu </html> yang merupakan akhir dari dokumen html.

1. Head

<head> merupakan tag tempat meletakkan informasi header. Pada tag ini terletak tag title, tag informasi web: meta, tag link dengan dokumen bahasa pemrograman lain seperti css, javascript, php, dll. Tag head ini juga memiliki tag penutup, yaitu </head>.

1. Title

<title> merupakan tag untuk mendefinisikan judul web. Tag ini juga memiliki tag penutup, yaitu </title>.

1. Body

<body> merupakan tag sebagai tempat untuk meletakkan seluruh isi web. Tag penutup dari tag ini adalah </body>. Seluruh isi web yang akan dibuat dikerjakan di dalam tag body, yaitu antara tag pembuka dan tag penutupnya.

* + 1. Pengelompokkan Tag-Tag HTML

Adapun beberapa pengelompokkan tag-tag dalam HTML beserta fungsinya yaitu sebagai berikut :

1. *Basic*

Adapun tag *basic* atau tag dasar yang terdapat dalam HTML sebagai berikut:

1. <!DOCTYPE>: Tag untuk menentukan tipe dokumen
2. <html> : Tag untuk membuat sebuah dokumen HTML
3. <title> : Tag untuk membuat judul dari sebuah halaman
4. <body> : Tag untuk membuat tubuh dari sebuah halaman
5. <h1> to <h6> : Tag untuk membuat heading
6. <p> : Tag untuk membuat paragraph
7. <br> : Memasukan satu baris putus
8. <hr> : Tag untuk membuat perubahan dasar kata didalam isi
9. *Formatting*

Adapun tag khusus yang berhubungan dengan format dalam HTML sebagai berikut :

1. <abbr> : Tag untuk membuat sebuah singkatan
2. <address> : Tag untuk membuat kontak alamat
3. <b> : Tag untuk membuat huruf bercetak tebal
4. <bdo> : Mengganti arah teks
5. <cite> : Tag untuk membuat judul karya
6. <code> : Tag untuk membuat potongan kode di antara text
7. <del> : Tag untuk membuat teks yang telah dihapus dari dokumen
8. <dfn> : Tag untuk membuat sebuah istilah definisi
9. <kbd> : Tag untuk membuat input keyboard
10. <meter> : Tag untuk membuat pengukuran skalar
11. *Forms*

Adapun tag-tag yang berhubungan dengan *form* dalam HTML sebagai berikut:

1. <form> : Tag untuk membuat sebuah form HTML untuk input pengguna
2. <input> : Tag untuk membuat sebuah kontrol input
3. <textarea> : Tag untuk membuat sebuah kontrol input multibaris (text area)
4. <button> : Tag untuk membuat sebuah tombol yang dapat diklik
5. <select> : Tag untuk membuat sebuah daftar drop-down
6. <option> : Tag untuk membuat pilihan dalam daftar drop-down
7. <label> : Tag untuk membuat sebuah label untuk sebuah elemen <input>
8. <iframe> : Tag untuk membuat sebuah bingkai
9. *Images*

Adapun tag-tag yang berhubungan dengan *image* atau gambar dalam HTML sebagai berikut :

1. <img> : Tag untuk membuat gambar
2. <map> : Tag untuk membuat gambar-peta
3. <area> : Tag untuk membuat area dalam gambar-peta
4. <figure> : Menentukan konten mandiri (tag baru HTML5)
5. Audio/Video

Adapun tag-tag yang berhubungan dengan audio atau video dalam HTML sebagai berikut :

1. <audio> : Tag untuk membuat isi suara.
2. <source> : Tag untuk membuat sumber beberapa media untuk elemen media (<video> dan <audio>)
3. <track> : Tag untuk membuat trek teks untuk elemen media (<video> dan <audio>).
4. <video> : Tag untuk membuat sebuah video atau film.
5. Links

Adapun tag-tag yang berkaitan dengan *link* dalam HTML sebagai berikut:

* 1. <a> : Tag untuk membuat hyperlink
  2. <link> : Tag untuk membuat hubungan antara dokumen dan sumber daya eksternal (paling sering digunakan untuk link ke style sheet)
  3. <nav> : Tag untuk membuat navigasi link (tag baru HTML5)

1. List

Adapun tag-tag yang berkaitan dengan *list* atau sumberdalam HTML sebagai berikut :

1. <ul> : Tag untuk membuat daftar dengan selain nomor
2. <ol> : Tag untuk membuat daftar dengan nomor
3. <li> : Tag untuk membuat sebuah item daftar
4. <dl> : Tag untuk membuat sebuah daftar definisi
5. <dt> : Tag untuk membuat istilah (item) dalam daftar definisi
6. <menu> : Tag untuk membuat deskripsi dari item dalam daftar definisi
7. *Tables*

Adapun tag-tag yang berhubungan dengan pembuatan tabel sebagai berikut :

1. <table> : Tag untuk membuat table
2. <caption> : Tag untuk membuat sebuah caption table
3. <th> : Tag untuk membuat sebuah sel header table
4. <tr> : Tag untuk membuat baris dalam sebuah table
5. <td> : Tag untuk membuat sel dalam sebuah table
6. <thead> : Tag untuk mengelompokan isi header dalam sebuah table
7. <tbody> : Tag untuk mengelompokan isi tubuh dalam sebuah table
8. <tfoot> : Tag untuk mengelompokan isi footer dalam sebuah table
9. <col> : Tag untuk menentukan properti kolom untuk setiap kolom dalam elemen.
   1. **Fungsi dari pemrograman HTML dan CSS**
10. Fungsi HTML

Fungsi HTML secara umum adalah untuk mengelola serangkaian data dan informasi sehingga suatu dokumen dapat diakses dan ditampilkan di Internet melalui layanan dan [halaman web](https://id.wikipedia.org/wiki/Laman" \t "_blank).

HTML sebenarnya bukan bahasa pemograman karena hanya berkumpulan teks yang memiliki [flowchart](http://www.fungsiklopedia.com/fungsi-flowchart/) dan instruksi. HTML merupakan dasar dari bahasa pemrograman web yang artinya anda harus mengusai HTML sebelum lanjut ke bahas pemerograman web.

Adapun Fungsi dan Kegunaan HTML adalah sebagai berikut :

1. Mengintegerasikan gambar dengan tulisan.
2. Membuat Pranala.
3. Mengintegerasikan berkas suara dan rekaman gambar hidup.
4. Membuat form interaktif.
5. Untuk menyunting format tulisan seperti **cetak tebal,** atau huruf miring dan garis bawah dll.
6. Menampilkan naskah dalam bentuk huruf yang mirip dengan hasil ketikan mesin ketik.
7. Mengubah-ubah ukuran tulisan untuk suatu karakter tertentu.
8. Fungsi CSS

Sebenarnya untuk fungsi CSS kamu sudah bisa mengetahuinya dari penjelasan sebelumnya. Pada dasarnya, CSS berfungsi untuk mendesain, membentuk, serta mengubah tampilan halaman sebuah *website*. CSS dapat bekerja dan berlaku melalui tag HTML. Dengan adanya CSS, tag HTML yang sederhana dapat diubah sehingga tampilan laman *website*pun menjadi terlihat lebih menarik dan efisien

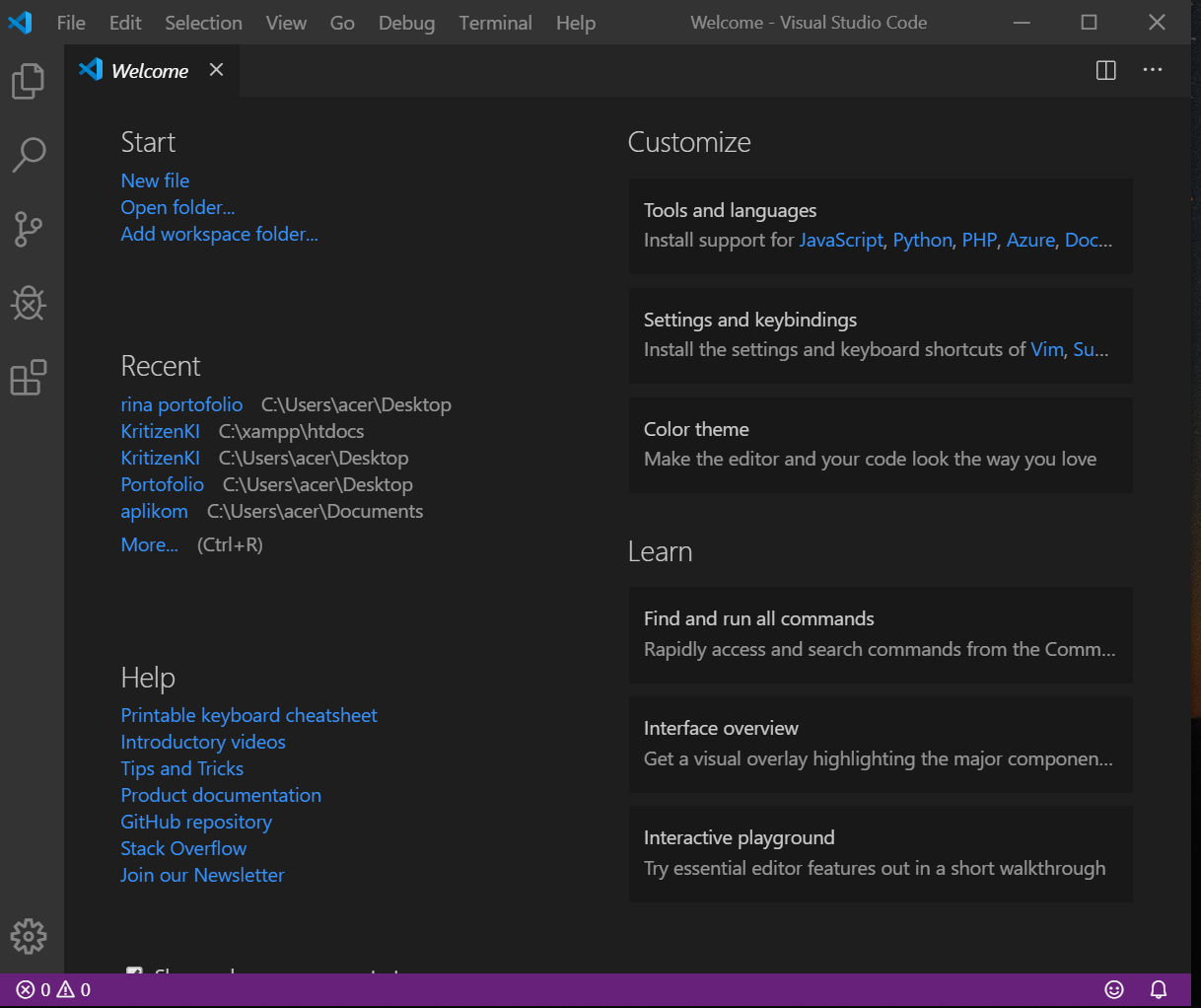
* 1. **Mengoperasikan HTML**

Ada beberapa langkah-langkah dalam mengoperasikan HTML menggunakan Visual Studio Code yaitu :

* + 1. Membuat file index HTML

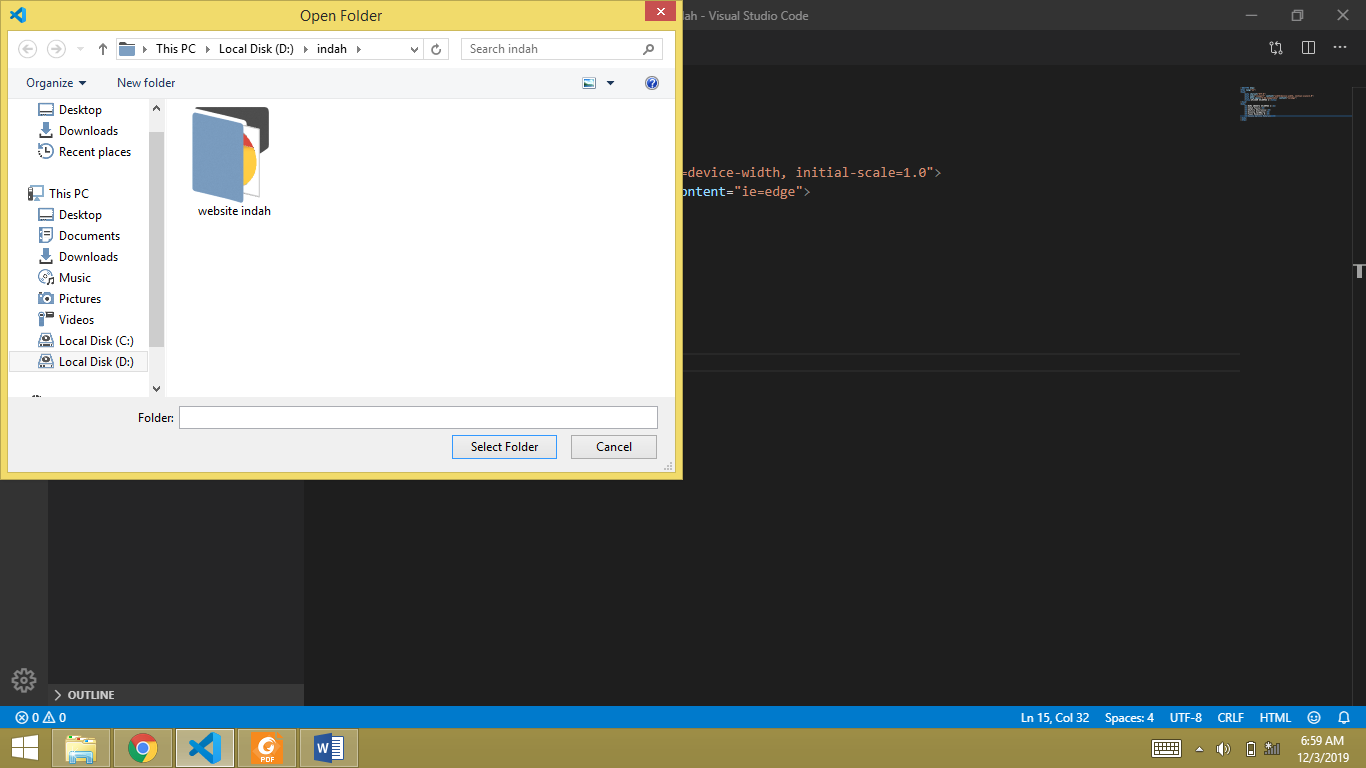
Adapun langkah-langkah dalam Membuat file index HTMLpada *software* visual code sebagai berikut :

1. Membuka aplikasi visual studio code, kemudian klik open folder



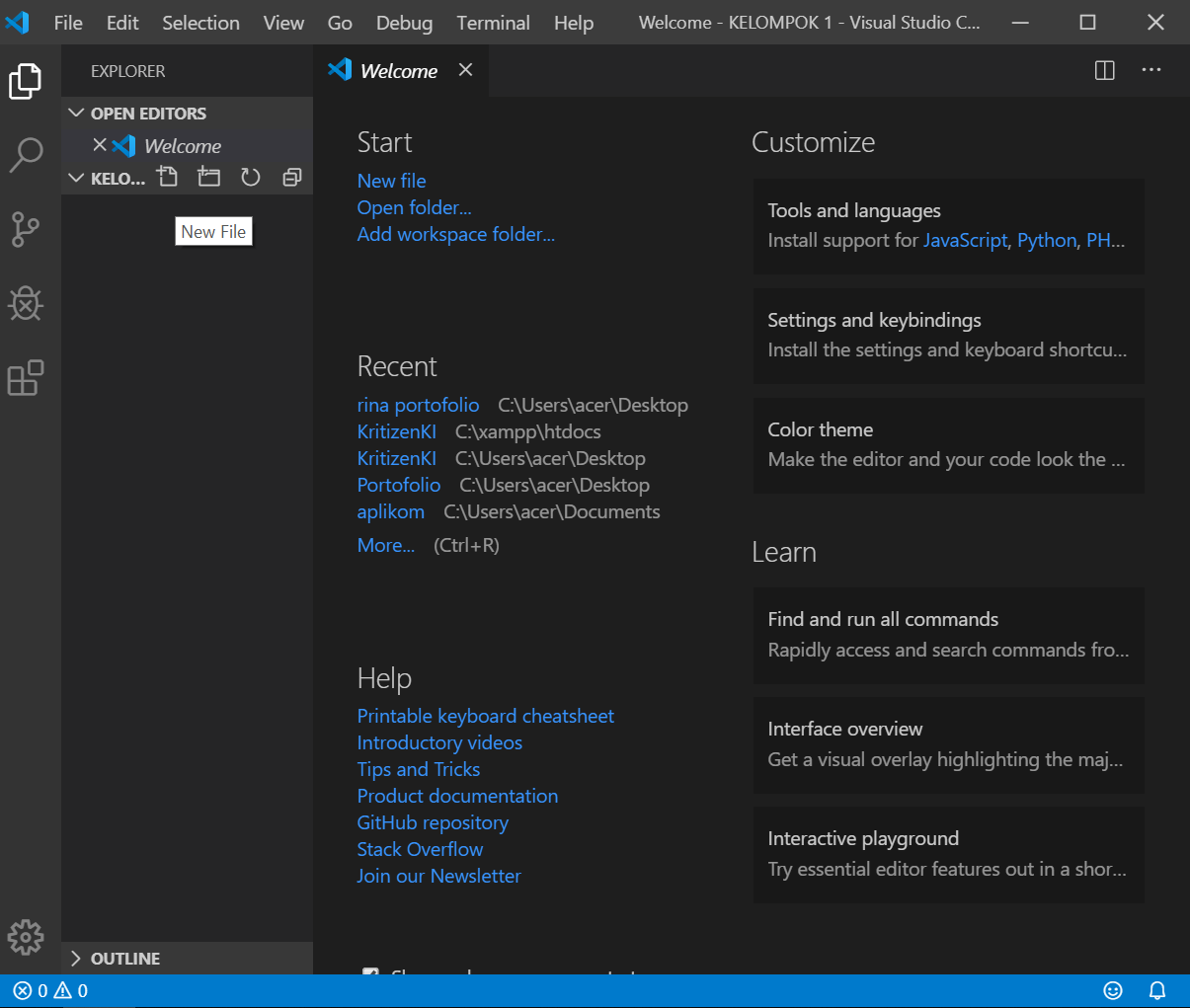
Gambar 3.1 Tampilan awal studio code

1. Kemudian pilih folder kosong yang sebelumnya telah dibuat, dan klik *select* folder.



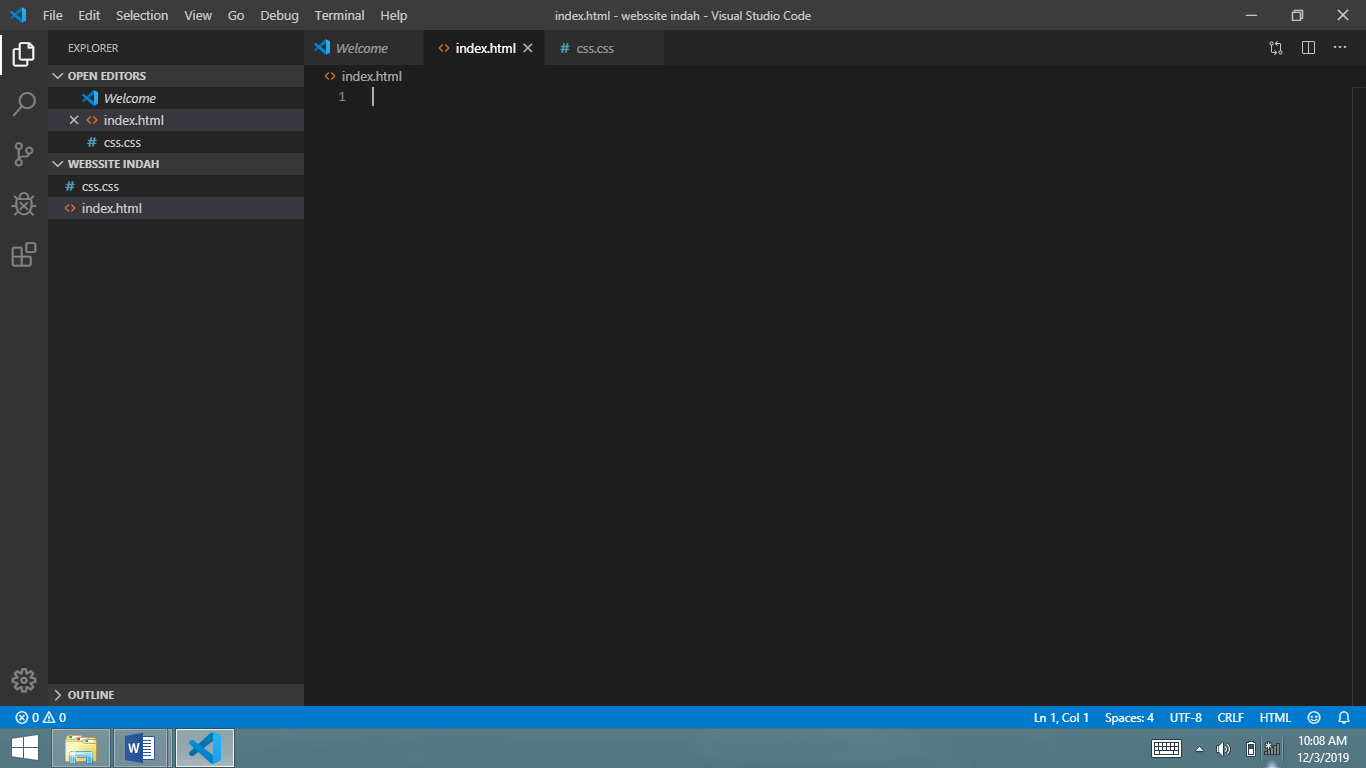
Gambar 3.2 Memilih folder kosong

1. Selanjutnya klik new file dan buat file pertama yang berisikan bahasa pemrograman HTML dan file kedua yang berisikan pemrograman CSS.



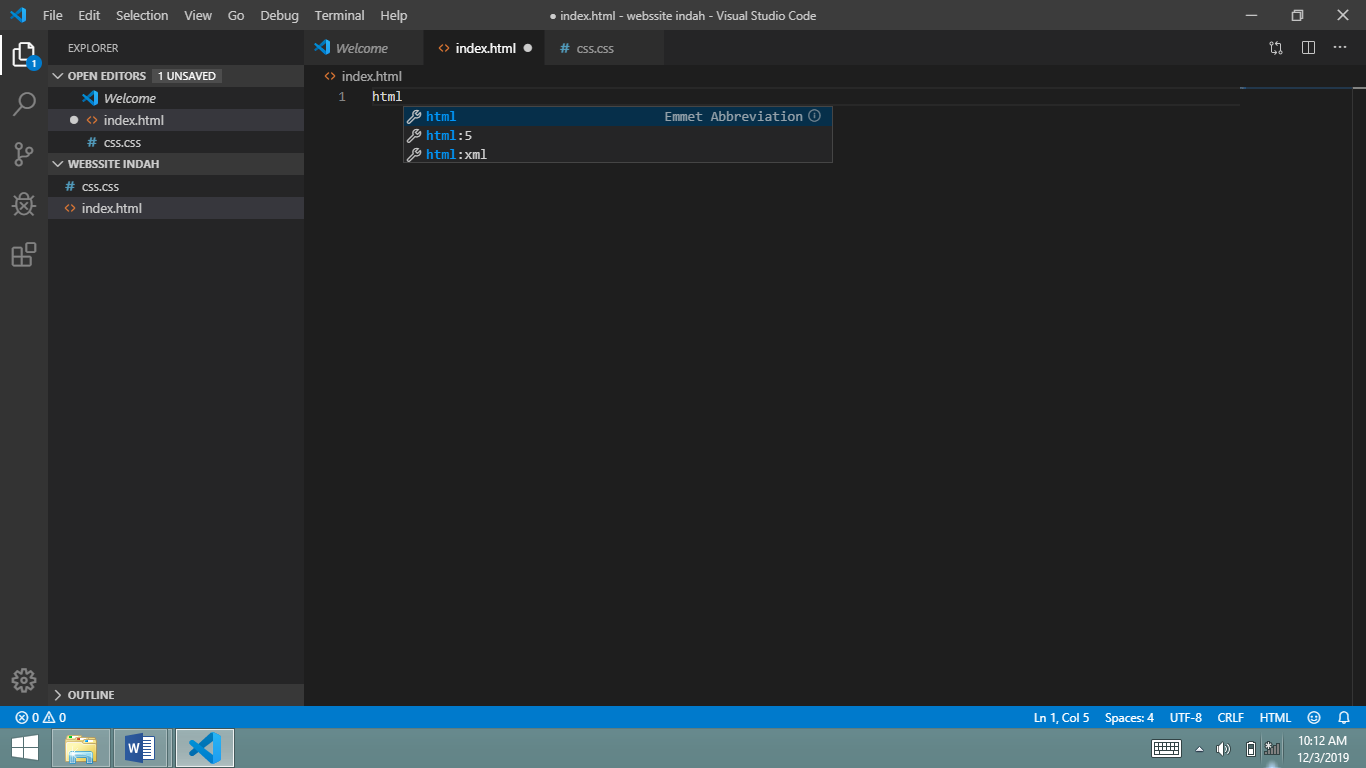
Gambar 3.3 Membuat file baru

1. Setelah itu akan muncul tampilan awal file yang telah dibuat sebelumnya dengan judul index.html dan css.css



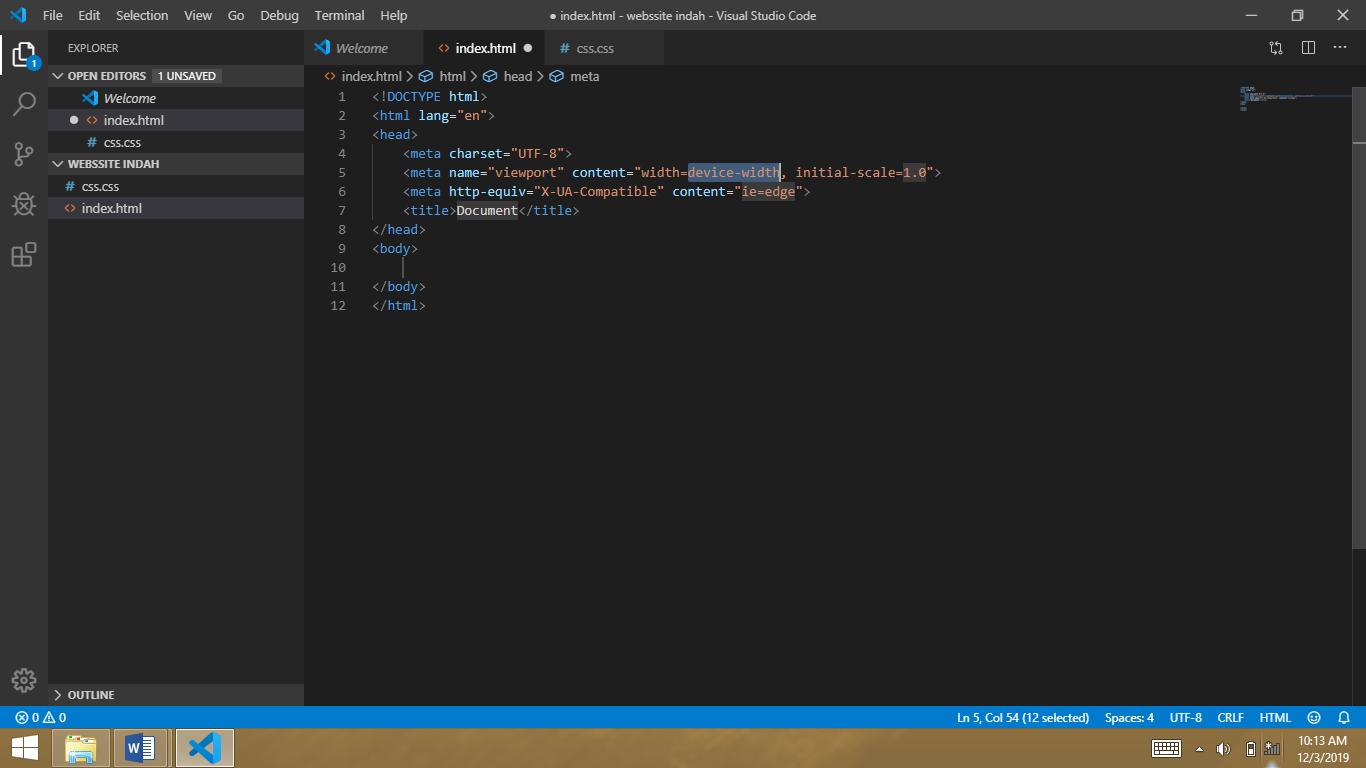
Gambar 3.4 Tampilan awal file baru

1. Setelah itu pada file index.html kita menuliskan html dan muncul pilihan html5, lalu enter.



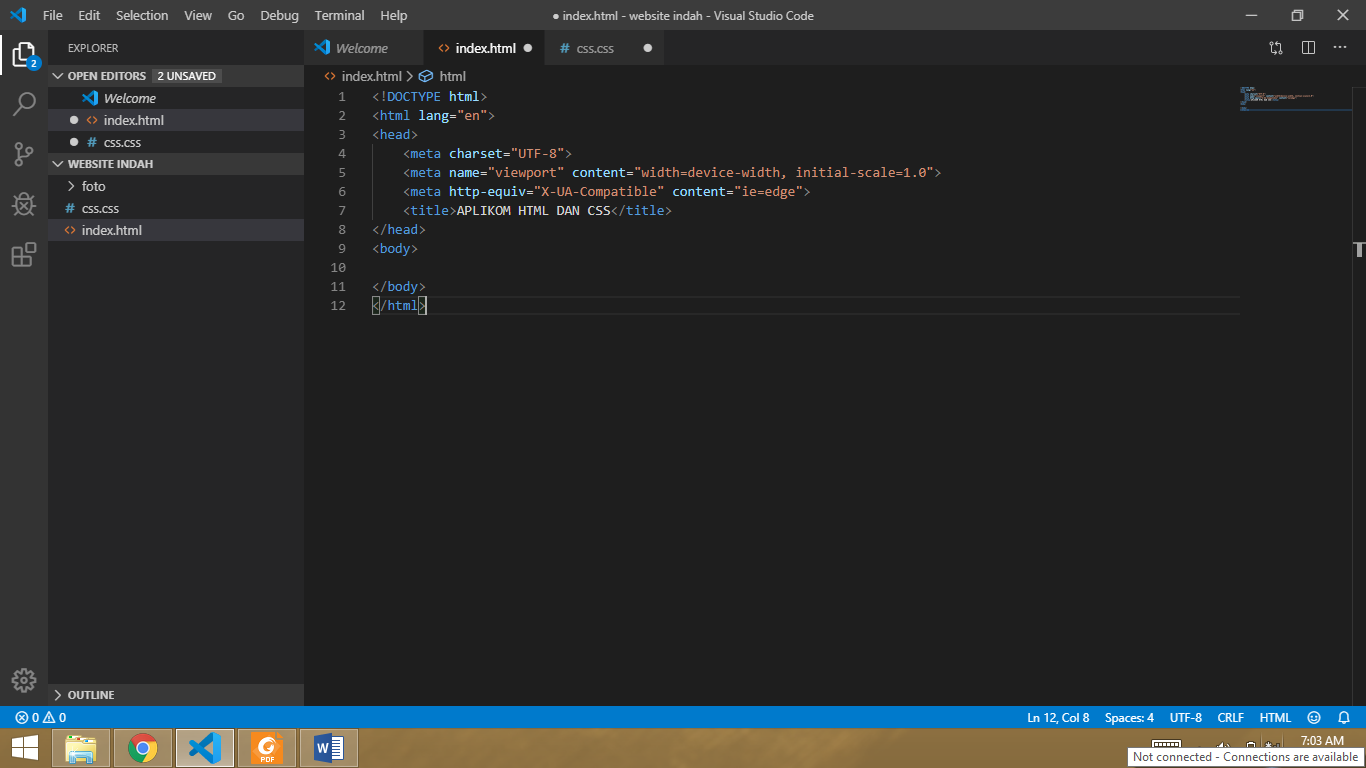
Gambar 3.5 Memilih tag html5

1. Kemudian akan muncul struktur dasar dari html 5.



Gambar 3.6 Struktur dasar HTML5

1. Dibagian tag <title> ganti nama document menjadi nama APLIKOM HTML DAN CSS ataupun sesuai keinginan masing-masing.

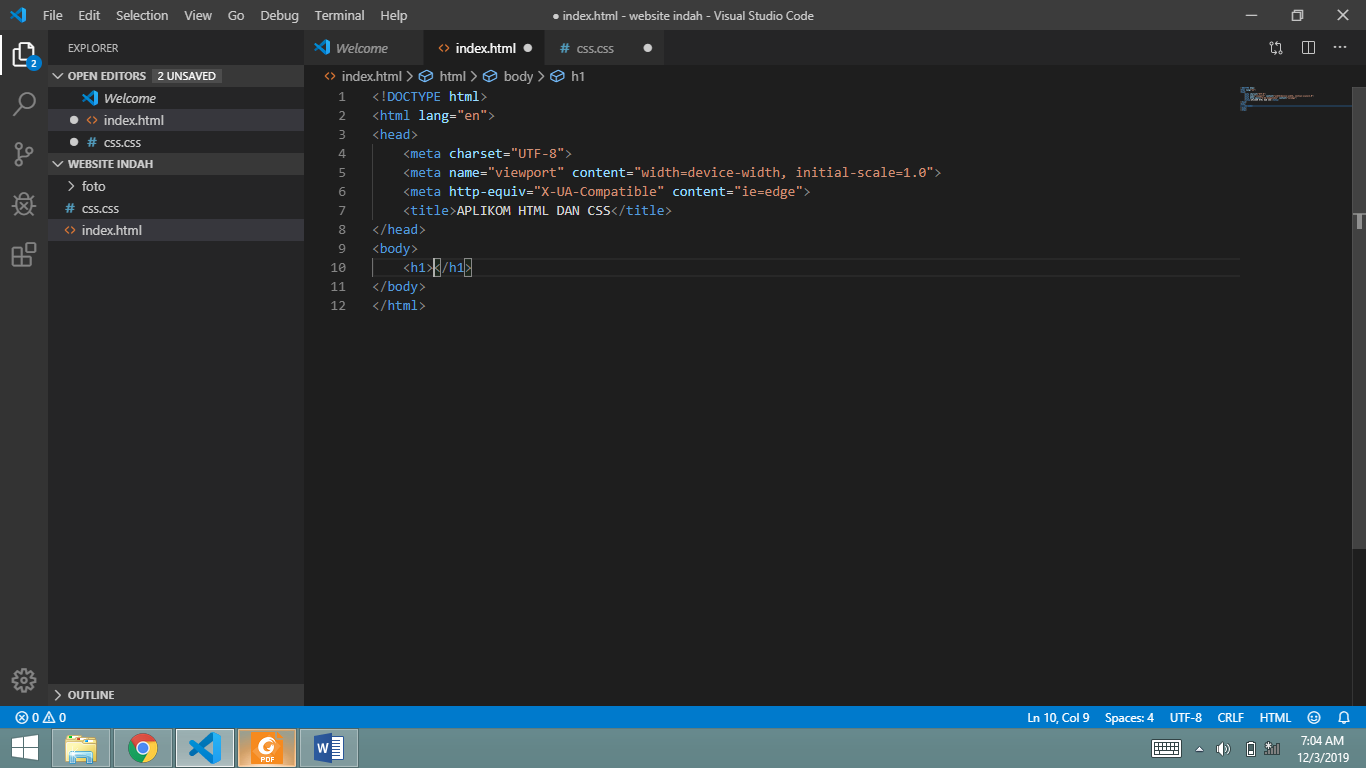


Gambar 3.7 Membuat judul website

* + 1. Membuat Heading HTML5

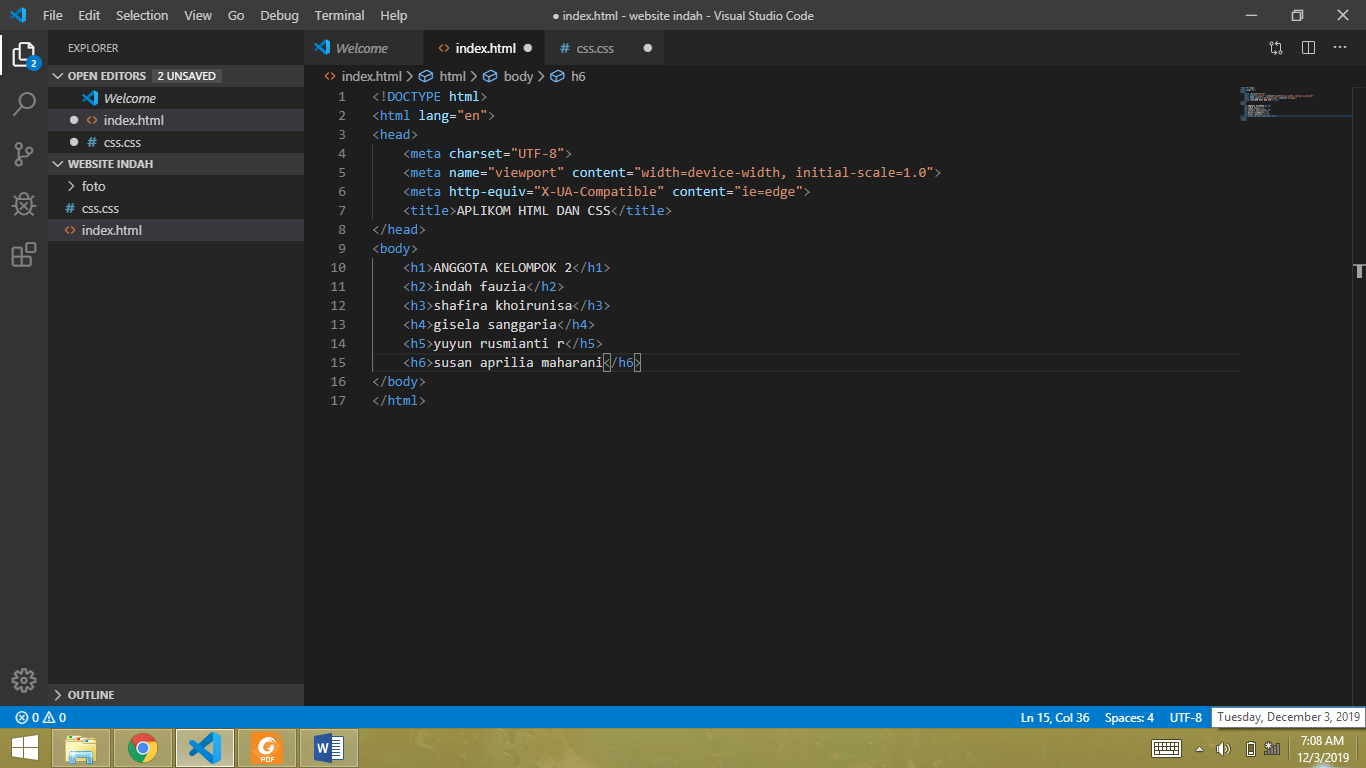
Adapun langkah-langkah dalam membuat heading pada HTML5sebagai berikut:

1. Membuat heading dengan tag <h1> yang dituliskan pada bagian bawah tag<body>

****

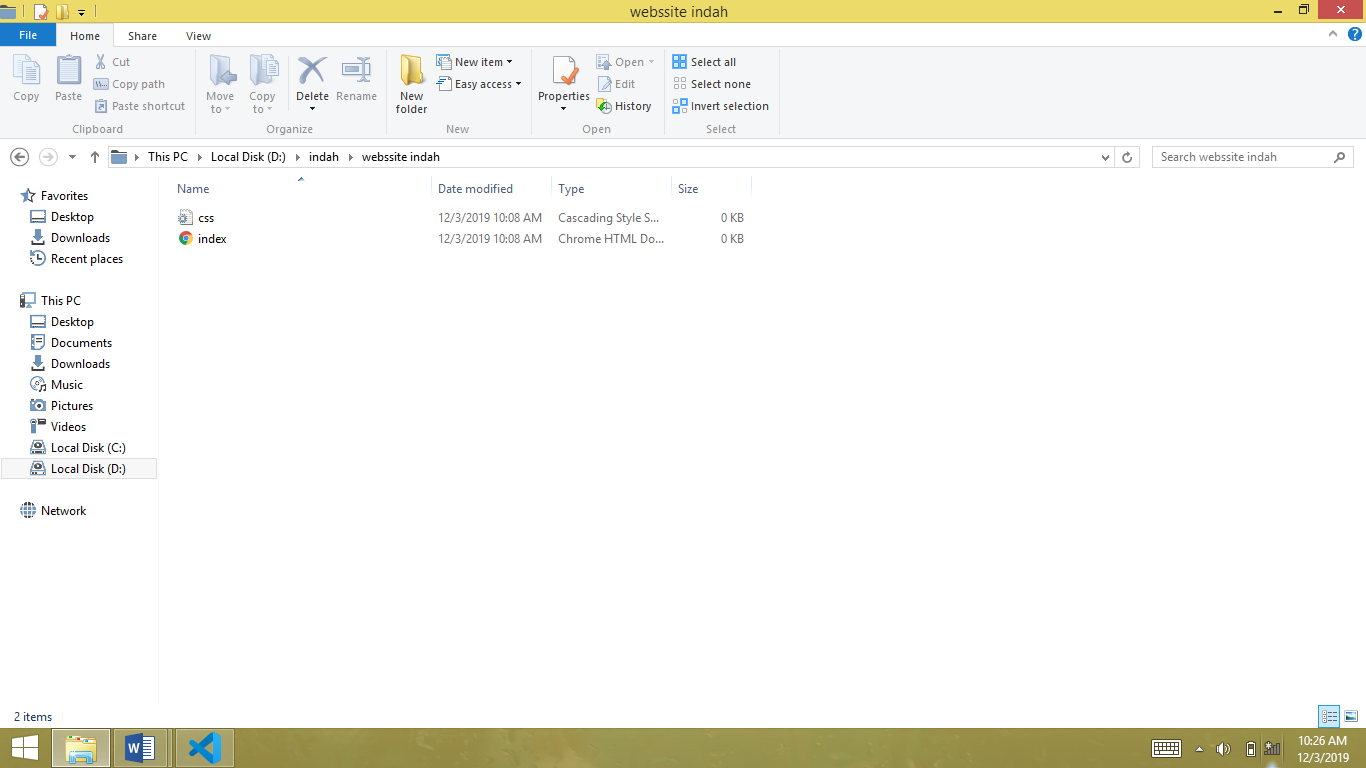
Gambar 3.8 Membuat heading

1. Kemudian tuliskan isi heading tersebut diantara dua tag heading pembuka dan penutupnya. Lalu *save* *file* tersebut



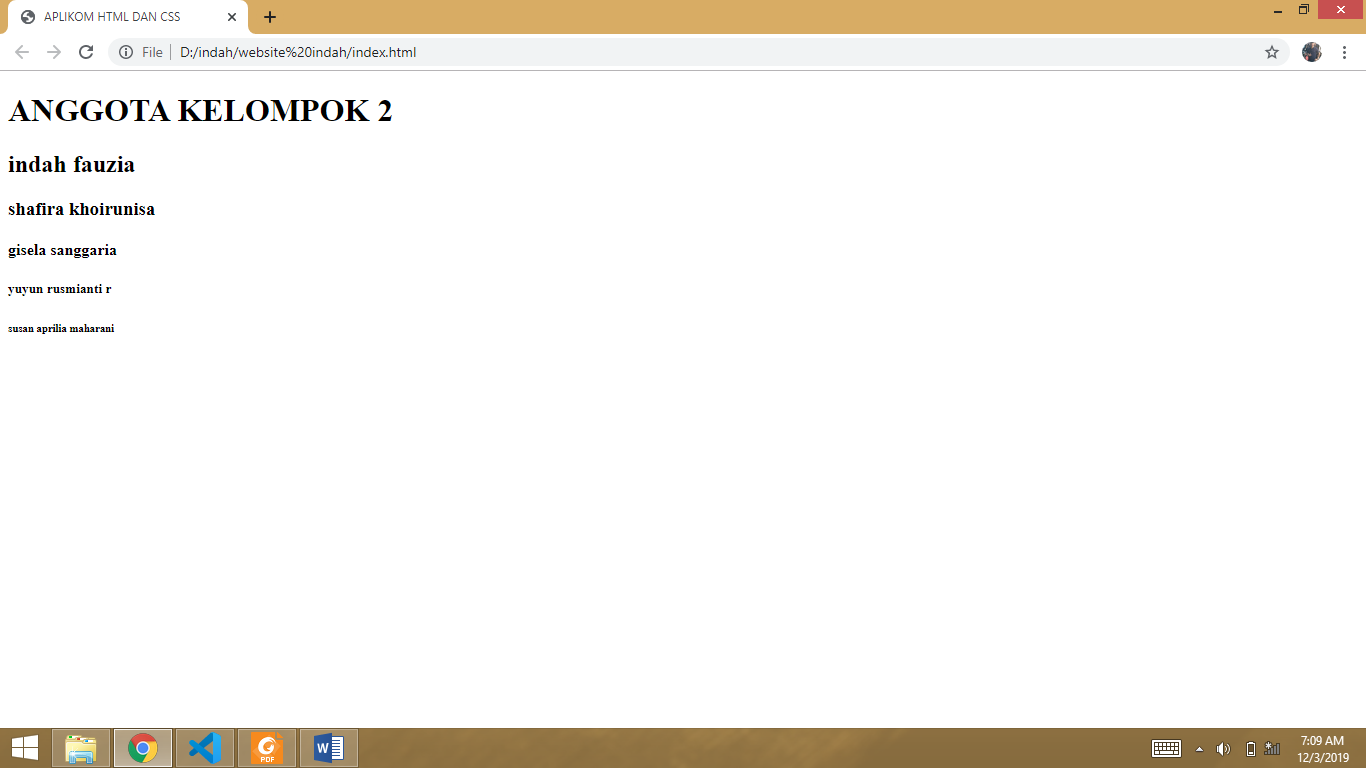
Gambar 3.9 Menuliskan isi heading

1. Setelah itu, buka folder yang dibuat tadi dan klik 2 kali file index tersebut



Gambar 3.10 Membuka folder

1. Kemudian akan muncul tampilan heding website yang telah dibuat sebelumnya.

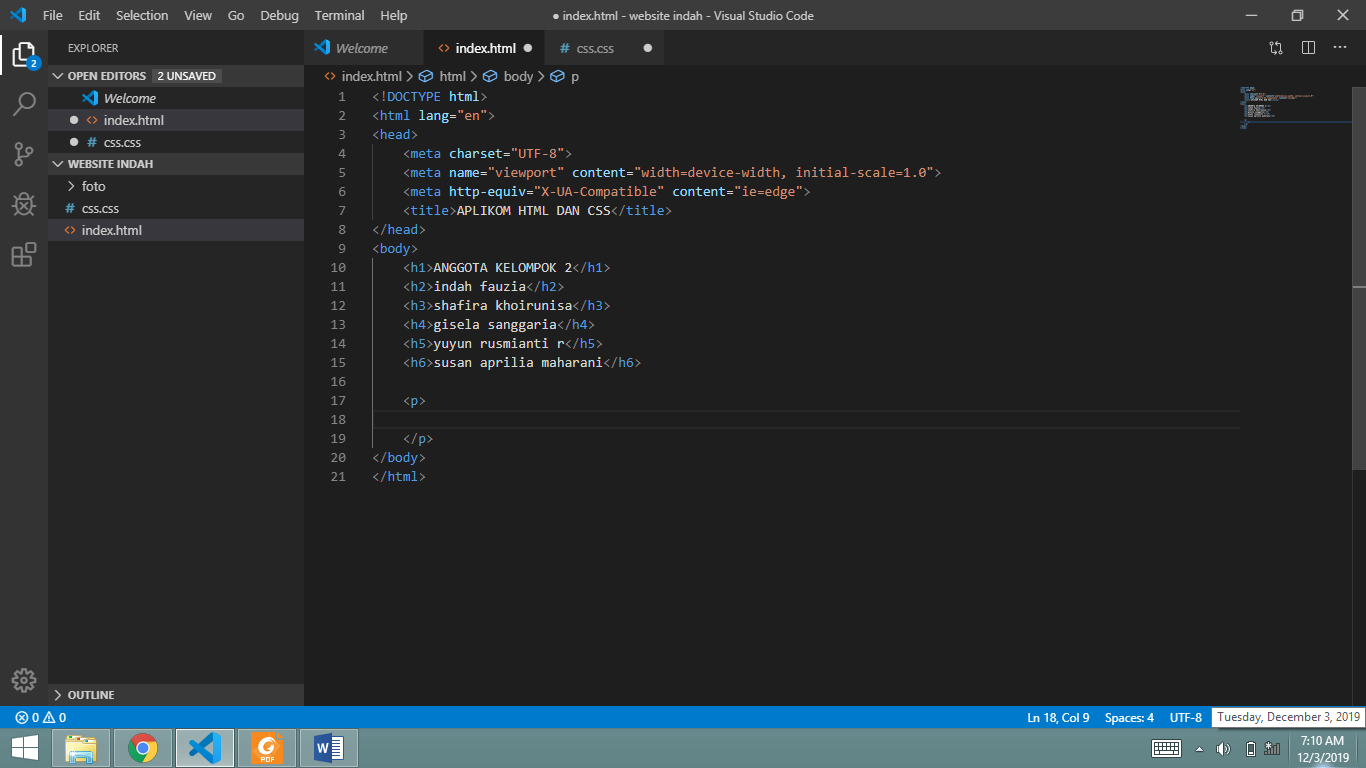


Gambar 3.11 Tampilan heading pada website yang telah dibuat

* + 1. Membuat Paragraf HTML5

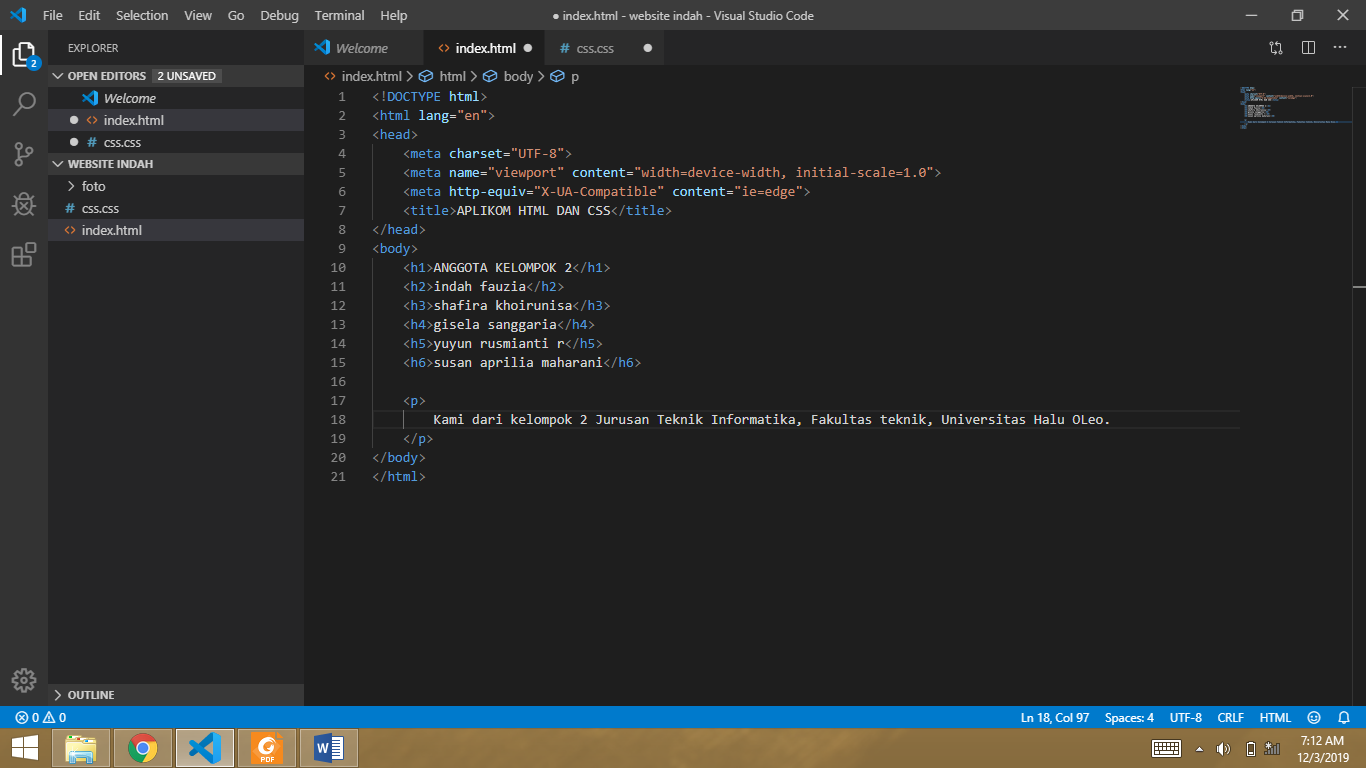
Adapun langkah-langkah dalam membuat paragraf pada HTML5sebagai berikut :

1. Membuat paragraf dengan tag <p> yang dituliskan setelah heading.

****

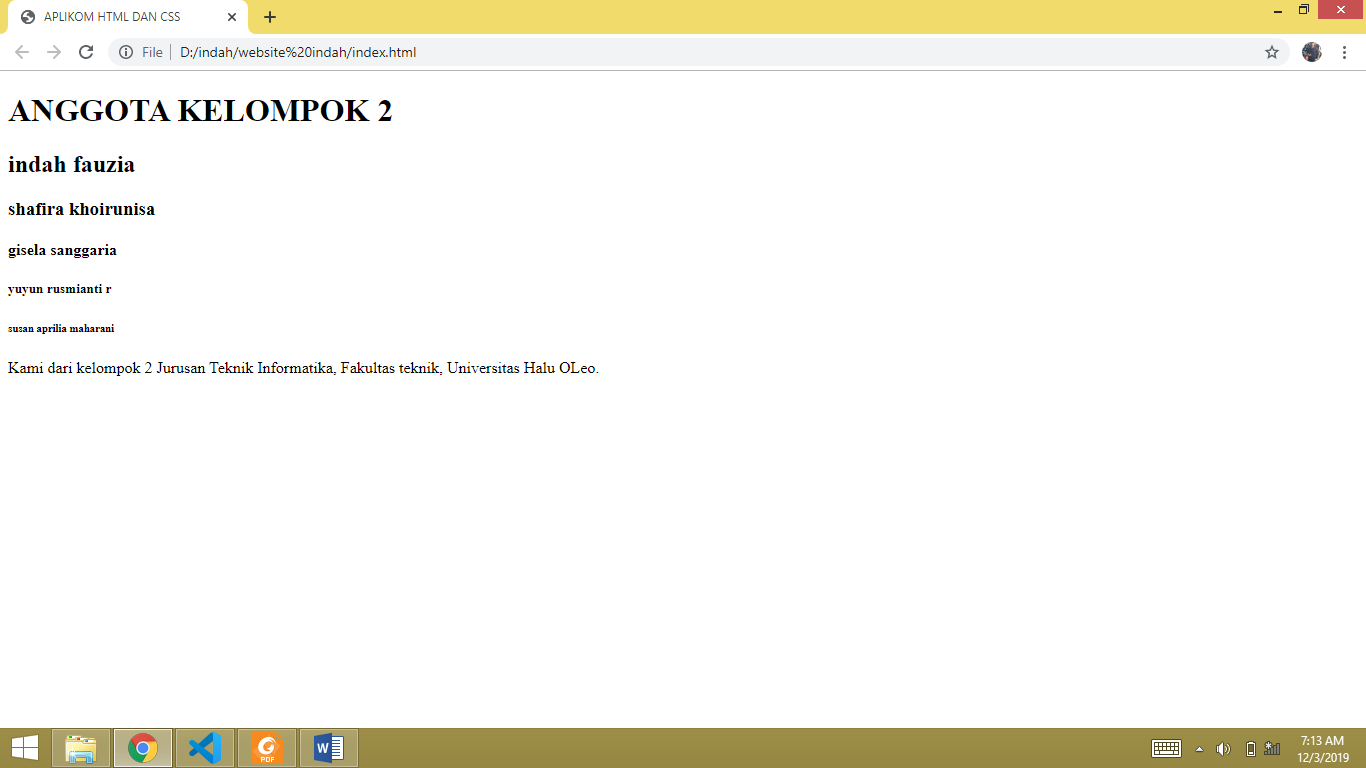
Gambar 3.12 Membuat paragraph

1. Kemudian tuliskan isi paragraf tersebut diantara dua tag paragraf pembuka dan penutupnya. Lalu *save* *file* tersebut



Gambar 3.13 Menuliskan isi paragraf

1. Kemudian refresh pada chrome untuk melihat perubahan penambahan paragraf yang akan muncul pada tampilan website yang telah dibuat sebelumnya.

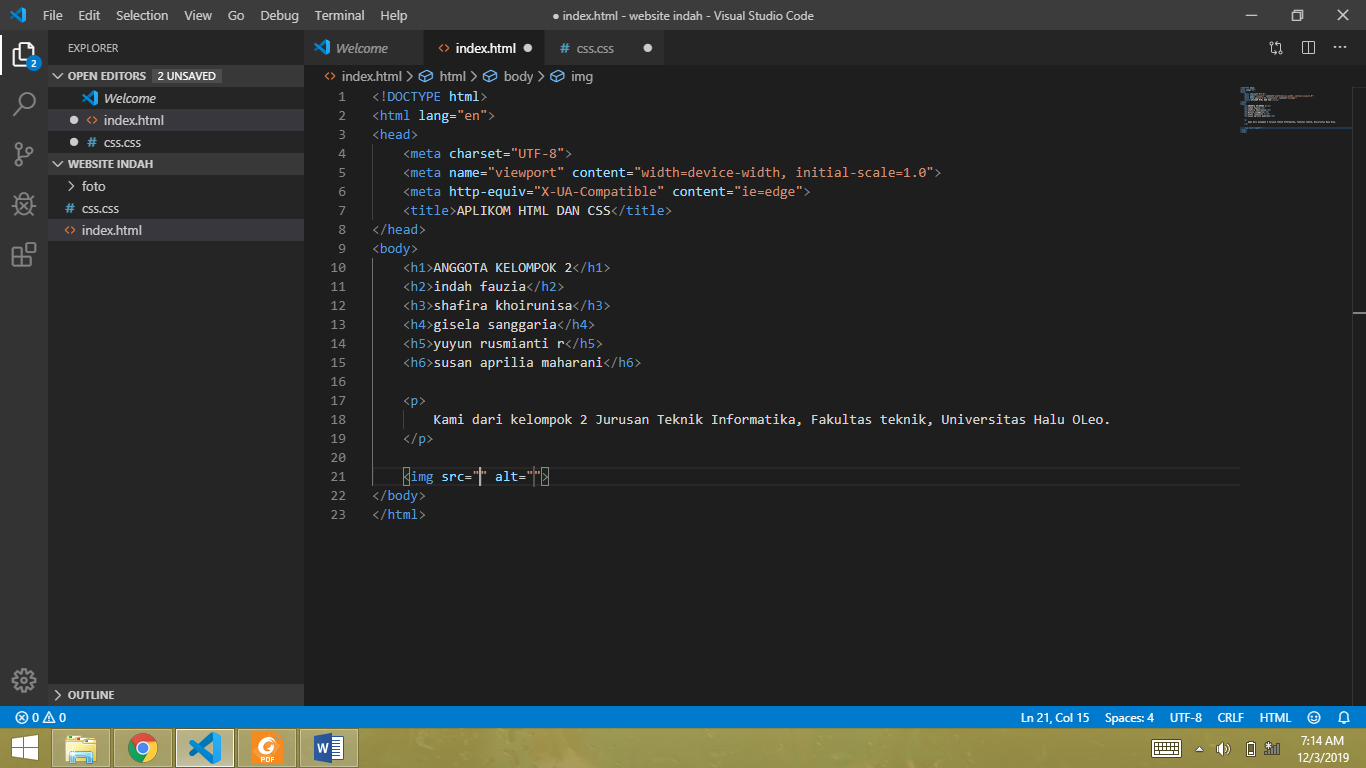


Gambar 3.14 Penambahan paragraf pada tampilan website

* + 1. Memasukan Gambar

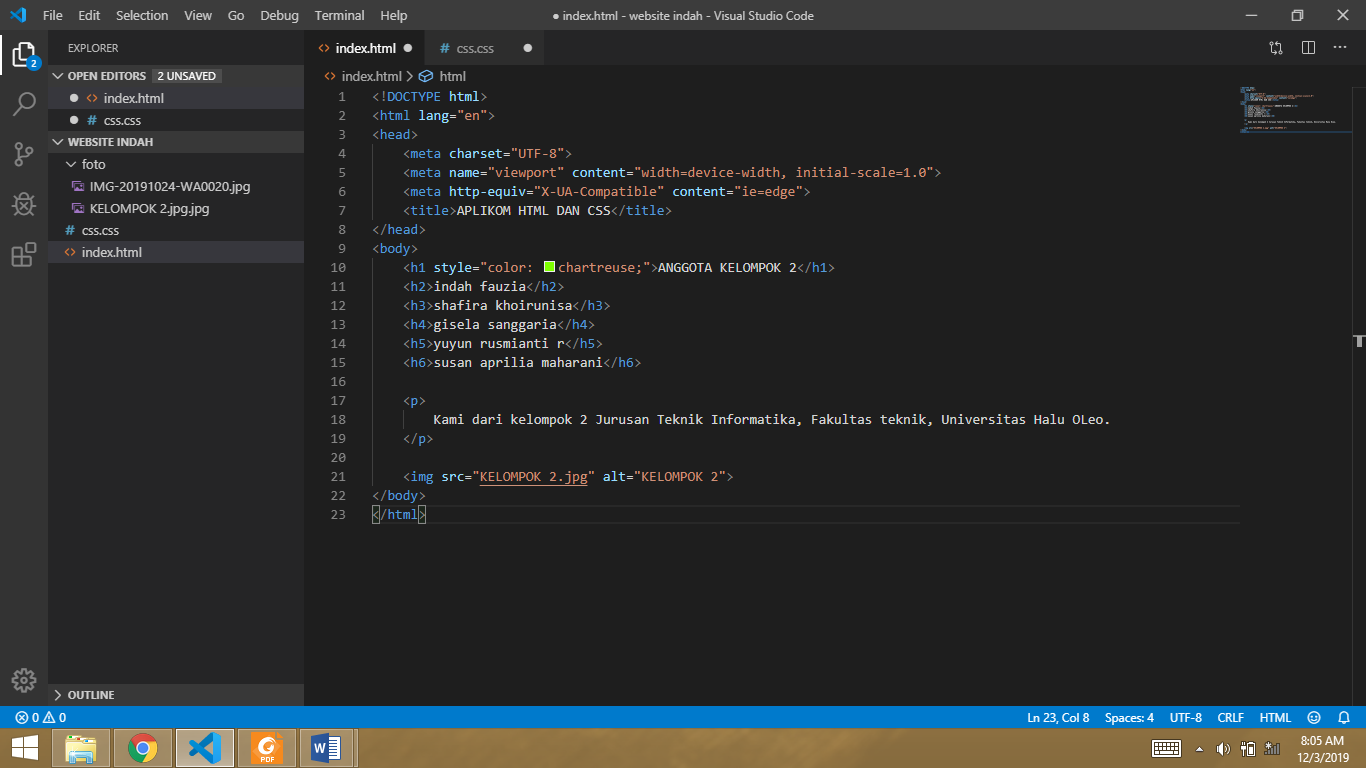
Adapun langkah-langkah dalam memasukan gambarsebagai berikut :

1. Memasukan gambar dengan menggunakan tag <img> yang dituliskan setelah tag paragraf.



Gambar 3.15 Menuliskan tag <img> untuk memasukan gambar

1. Kemudian masukan gambar diantara tanda petik dua tersebut.

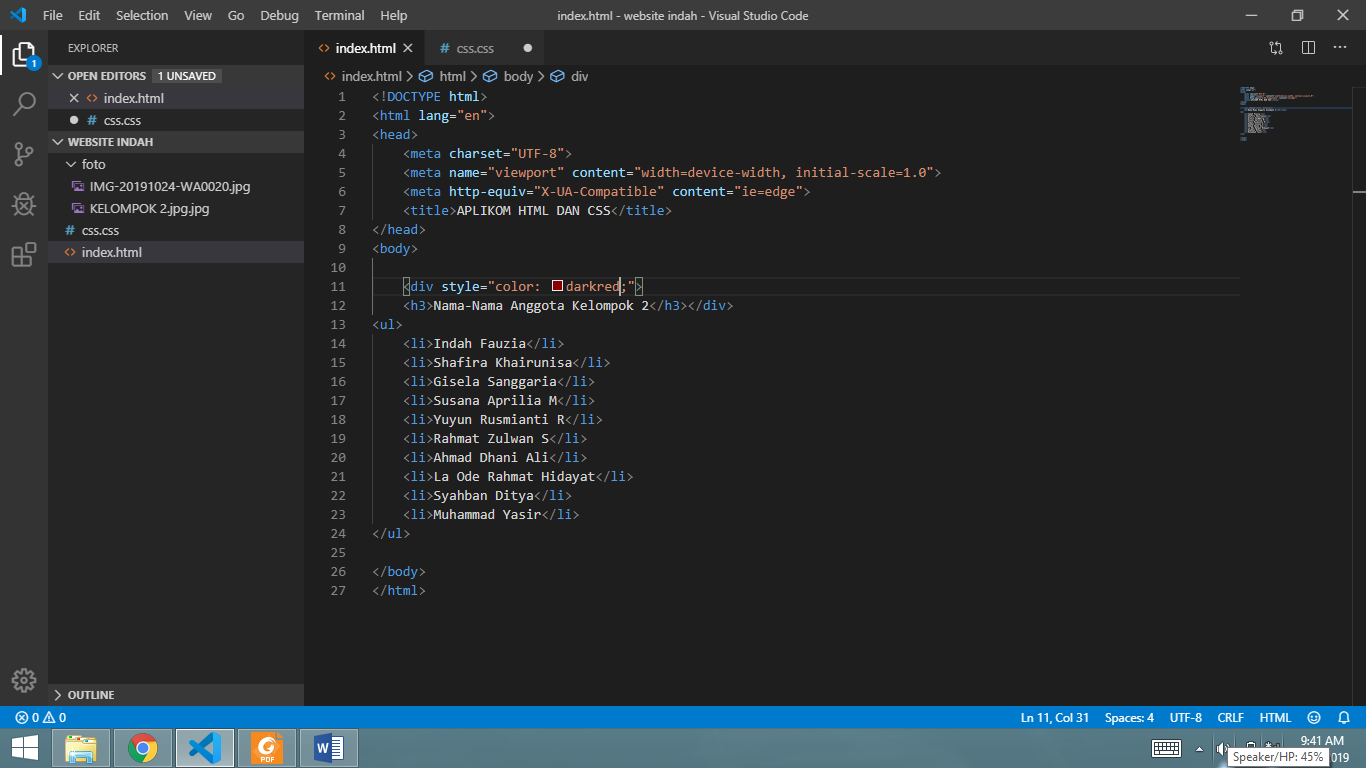


Gambar 3.16 Memasukan gambar

* + 1. Membuat Inline Style

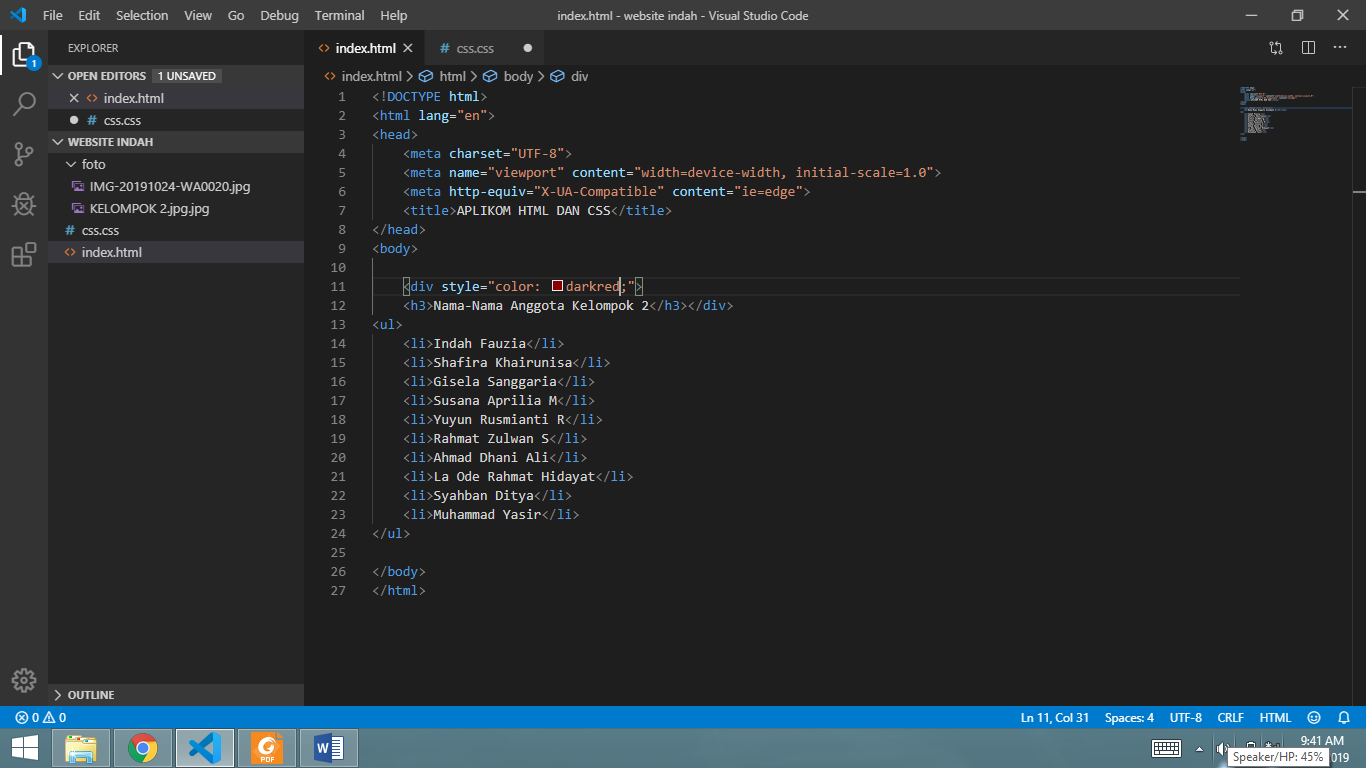
Adapun langkah-langkah dalam membuat inline stylesebagai berikut :

1. Menambahkan tag <style> pada h1

****

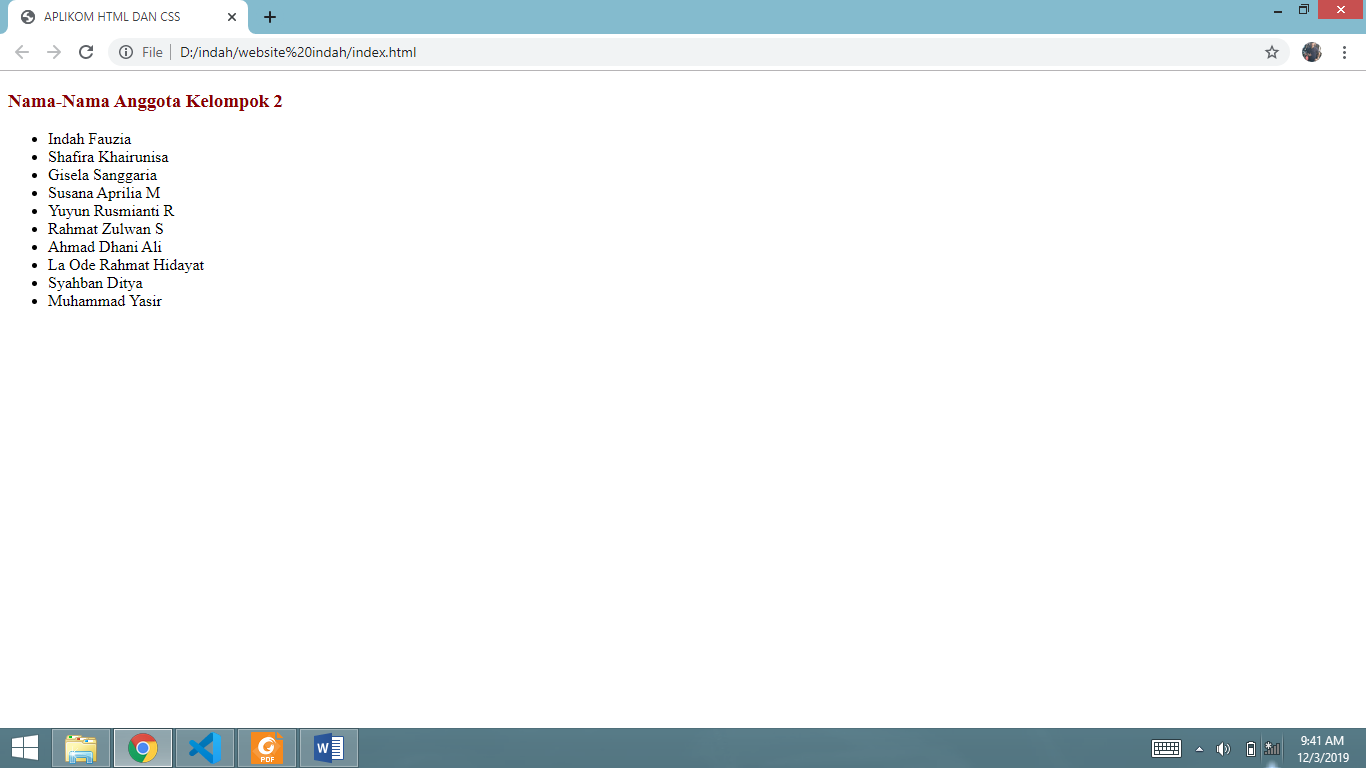
Gambar 3.18 Menambahkan tag style dalam tag <h1>

1. Kemudian pada pertengahan tanda kutip dua tambahakan warna (*color*) sesuai keinginan.

****

Gambar 3.19 Memberikan warna pada heading 1

1. Kemudian refresh pada chrome untuk melihat perubahan dari warna pada heading 1 yang muncul pada tampilan website yang telah dibuat sebelumnya.

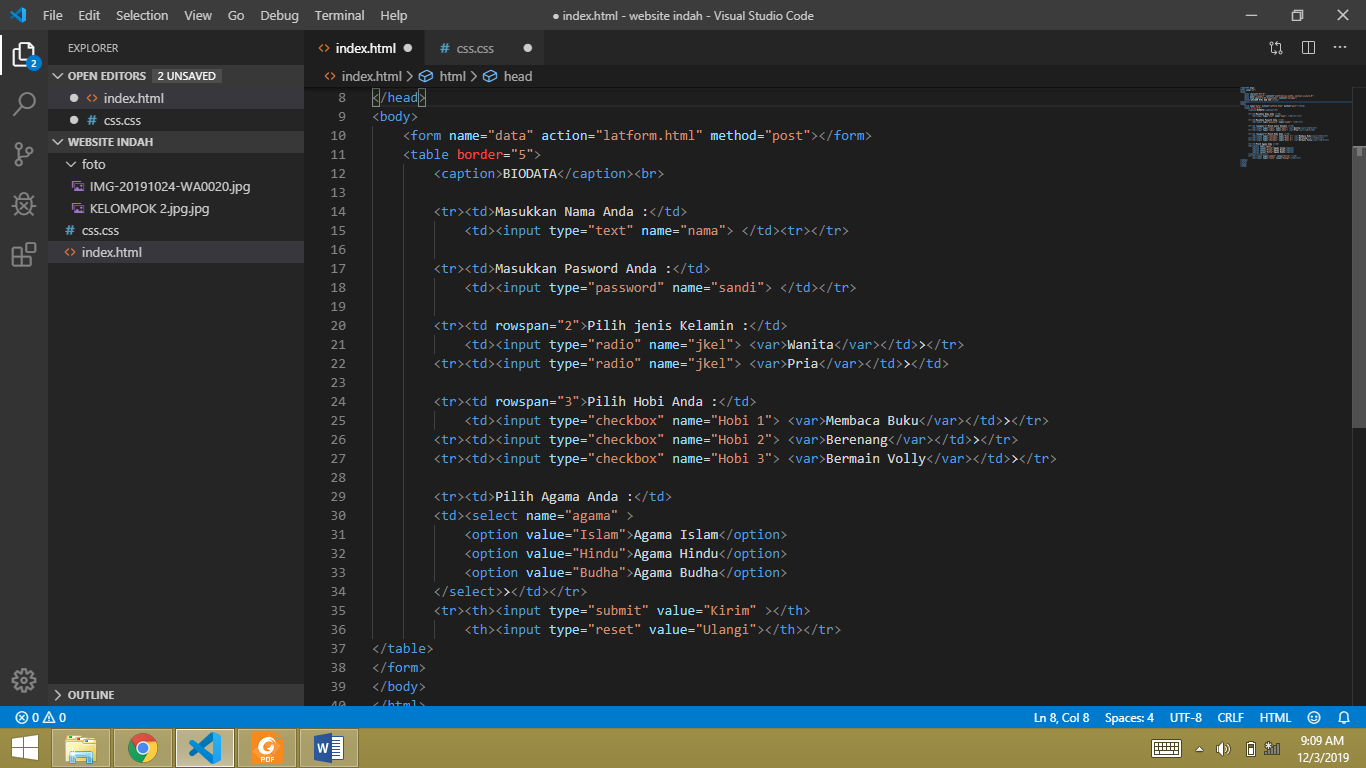
****

Gambar 3.20 Perubahan warna pada heading 1

* + 1. Membuat Tabel

Adapun langkah-langkah dalam membuat tabelsebagai berikut :

1. Pertama-tama klik tag form kemudian akan muncul secara otomatis name, action, dan method.
2. Klik tag table untuk membuat sebuat tabel. Kemudian pilih tabel border dan mengisi table sesuai dengan keinginan masing-masing.
3. Klik tag caption untuk menambahkan judul pada tabel tersebut.
4. Kemudian, buat tabel tersebut dengan meng-klik <tr><td>. Selanjutnya ikuti langkah langkah sesuai pada gambar di bawah ini.



Gambar 3.21. Membuat tabel

Struktur HTML untuk membuat tabel menggunakan beberapa jenis tag. Beberapa diatantaranya adalah <table>, <tr>, <th> dan <td>. Contohnya adalah sebagai berikut:

* 1. <table>

Menandakan akan dibuatnya tabel. Pada tag ini terdapat beberapa atribut yaitu border (untuk menentukan tebal garis tabel), cellpadding (untuk menentukan jarak dari pinggir sel ke tulisan di dalam sel), dan cellspacing (jarak tiap sel dalam tabel)

* 1. <tr>

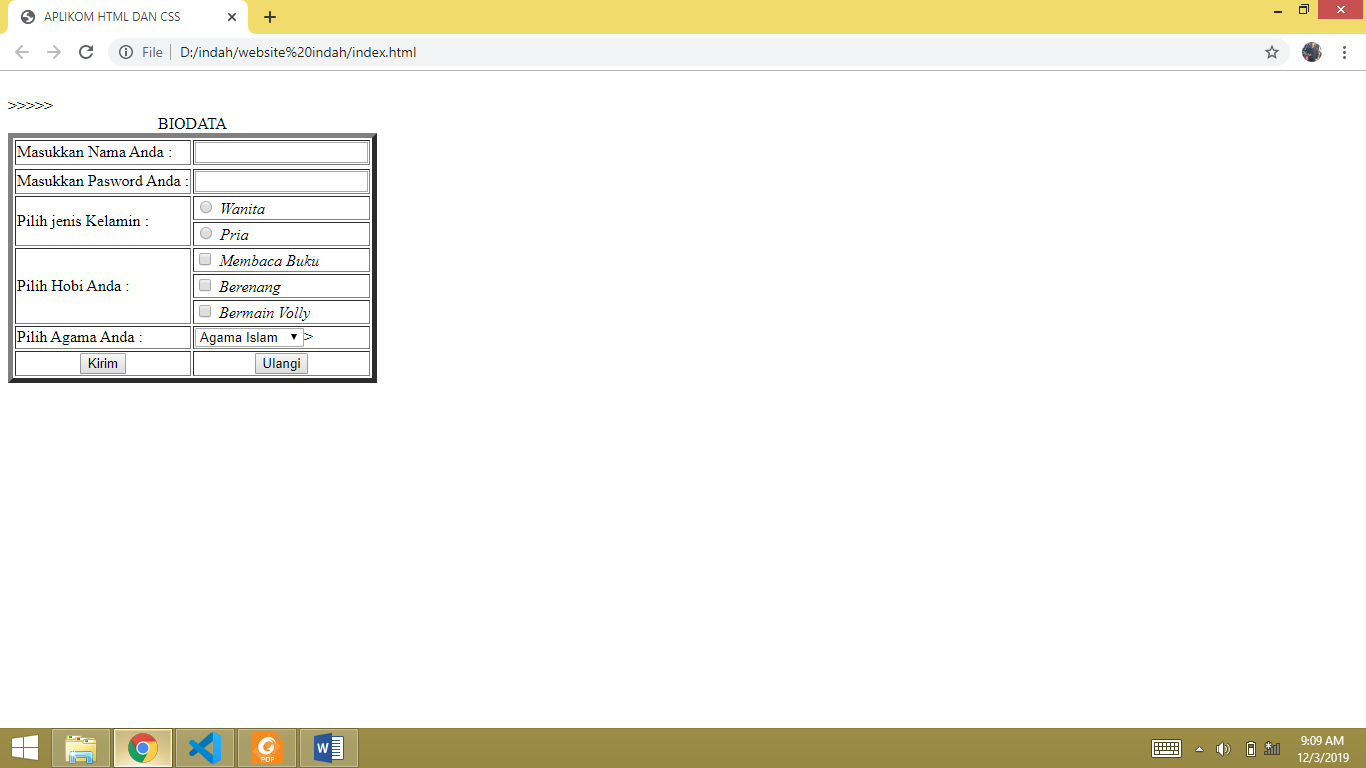
Menandakan sebuah baris dalam tabel. Pada tag ini terdapat atribut bgcolor yang berfungsi untuk mengubah warna (dalam hal ini baris) menjadi warna abu-abu. Atribut bgcolor juga dapat diletakan di tag lainnya seperti tag <table> untuk mengganti warna tabel

1. <th>

Menandakan sebuah sel utama (sel kepala) dari sebuah tabel. Setiap sel kepala akan di cetak tebal dan dirata tengahkan.

1. <td>

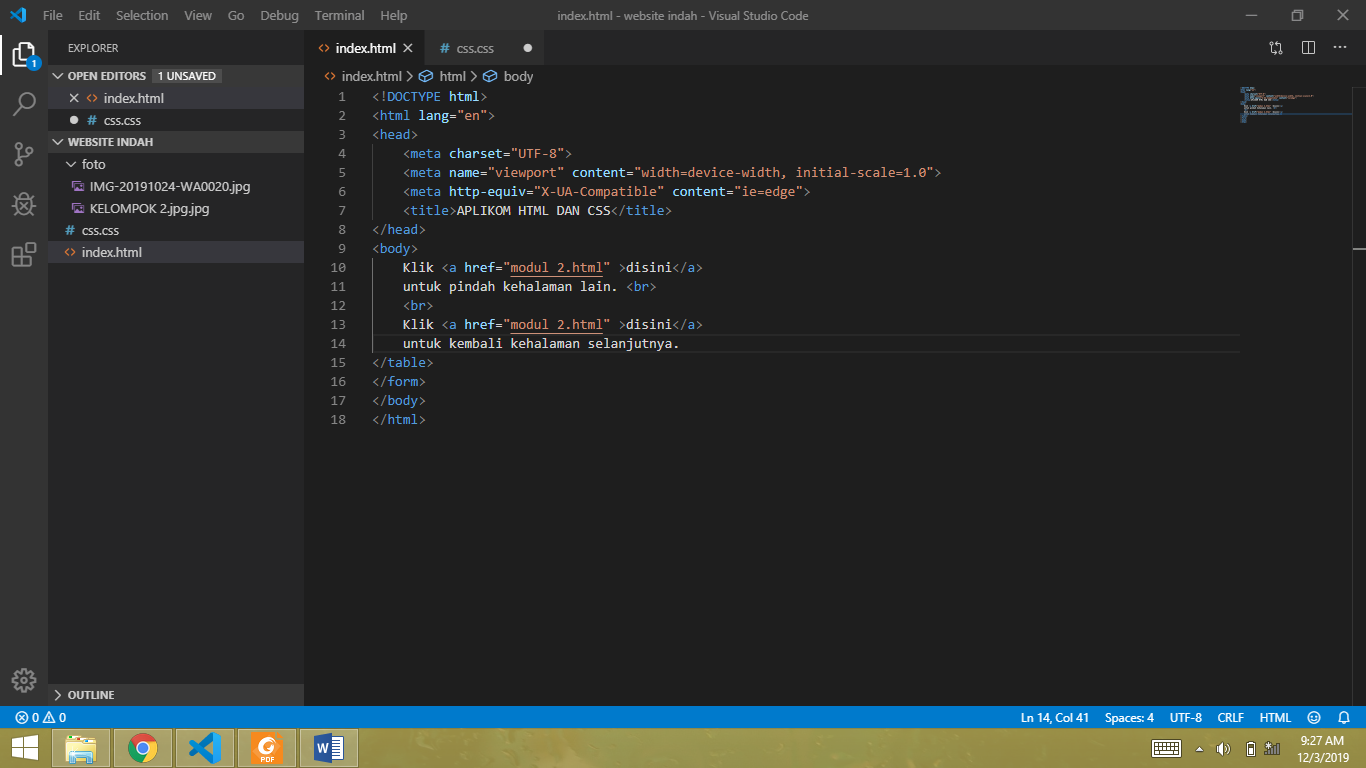
Menandakan sebuah sel dalam tabel.

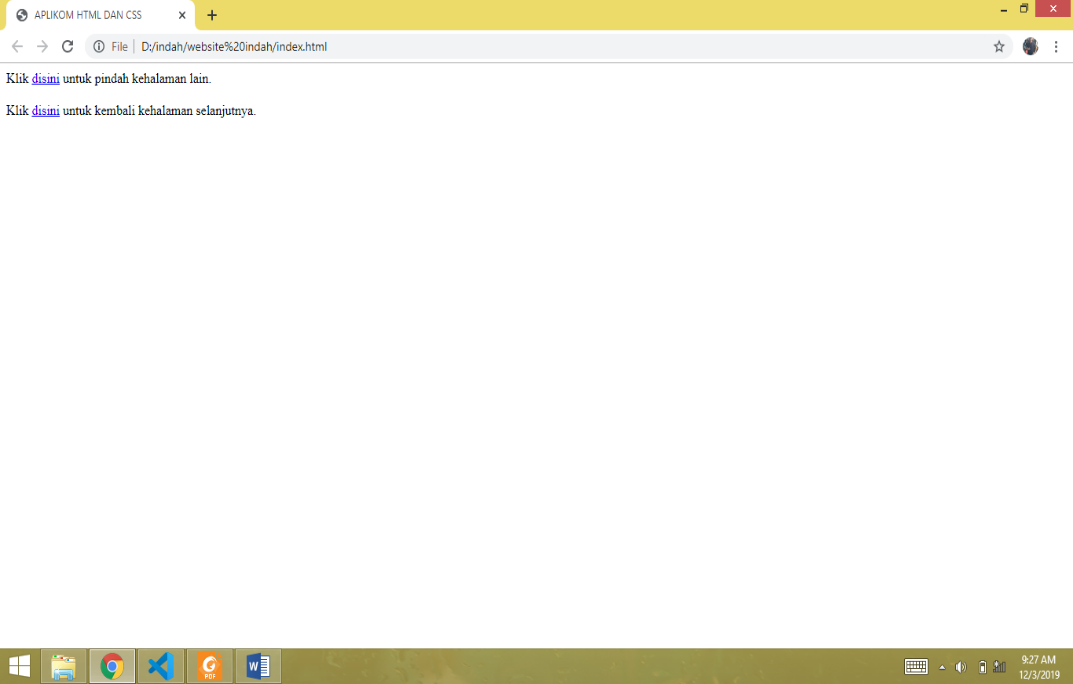


Gambar 3.21 Tampilan tabel

## Menambahkan Link ke Halaman Lain

Halaman web terasa kurang lengkap jika tidak memiliki link. Link adalah penghubung dari satu halaman ke halaman lain. Untuk menambahkan link ke halaman lain, dapat menggunakan tag <a>. Berikut contoh penggunaanya:

Gambar 3.22 Menambahkan Link ke Halaman Lain

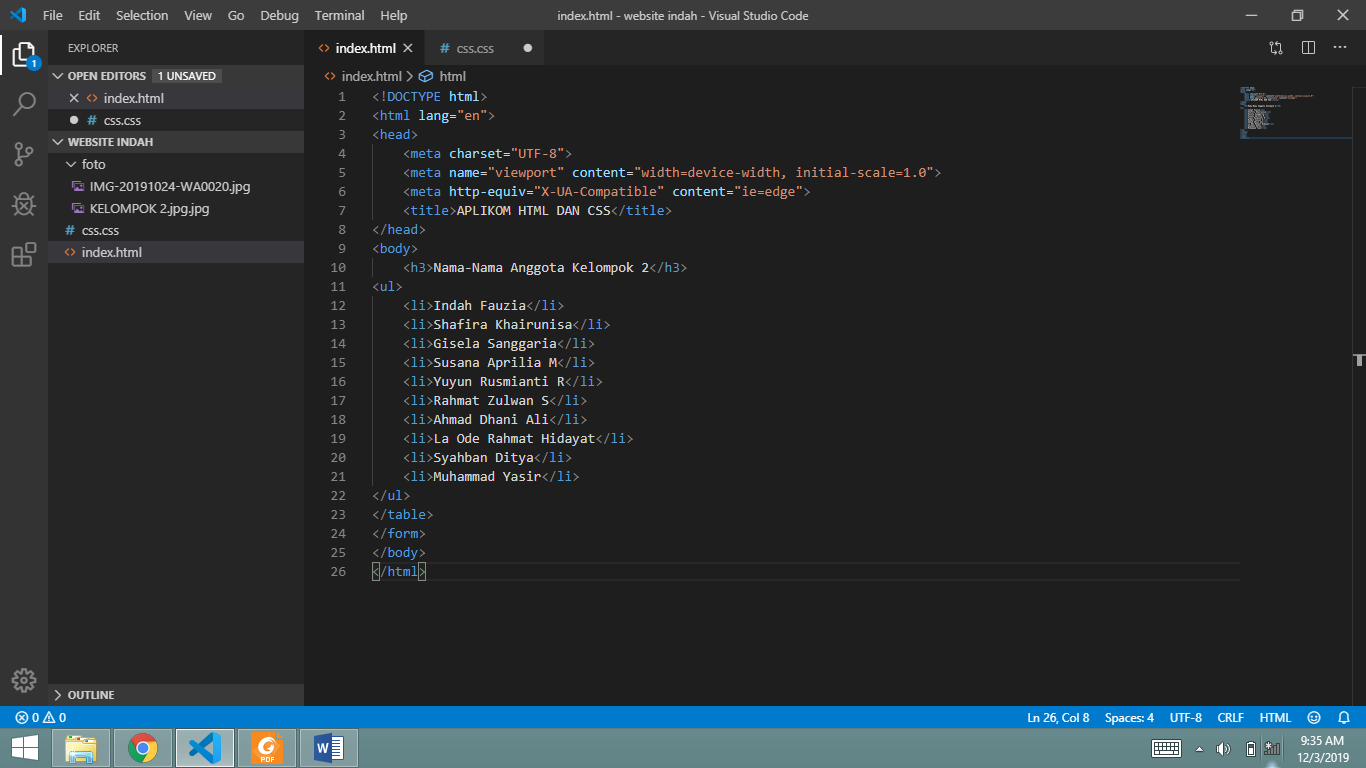


Gambar 3.23 Tampilan Hasil

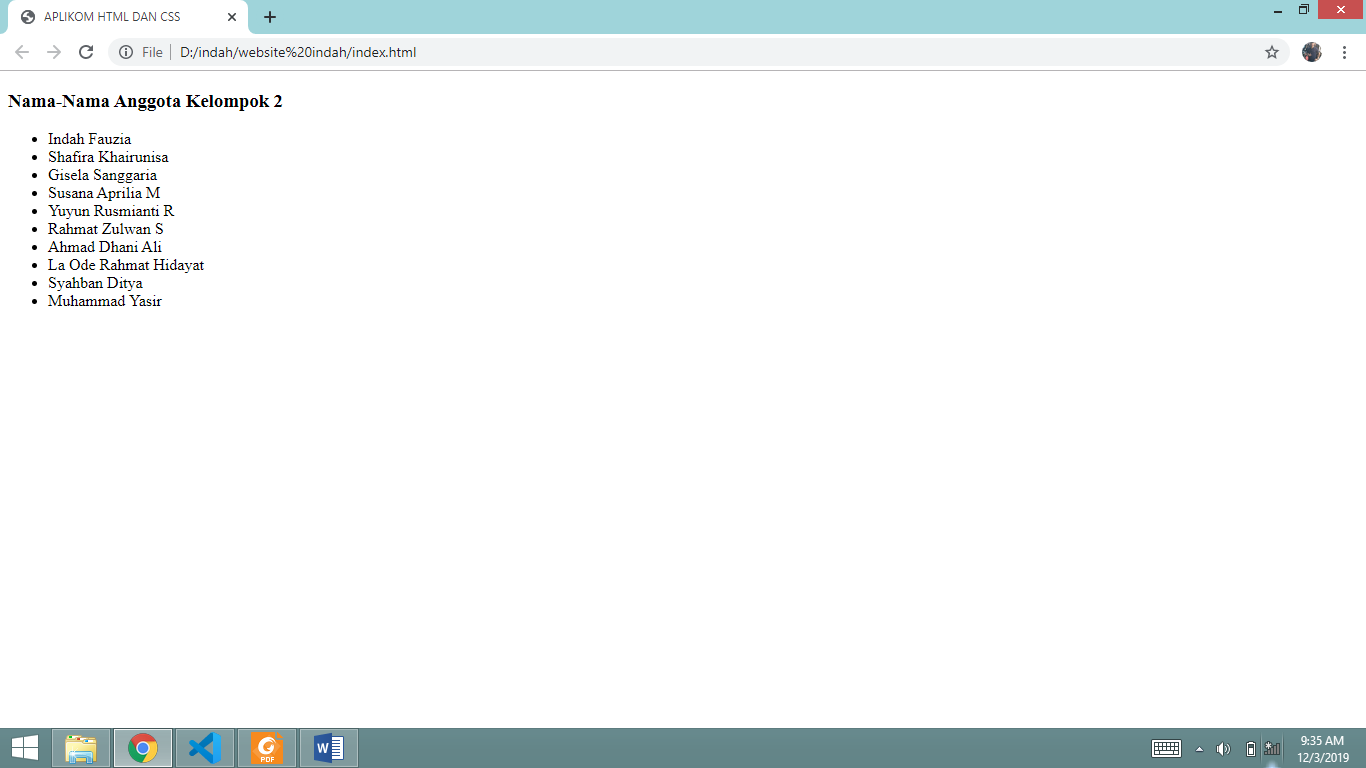
Penggunaan tag <a> sangat bergantung pada atribut *href*-nya. Fungsi atribut ini yaitu menunjukan letak alamat yang akan dituju. Apabila kita ingin membuka halaman baru di tab yang baru dengan atribut *target* (*target*=\_blank).

## Membuat List

Jenis list yang digunakan bisa termasuk *unordered list* (list yang tidak memperhatikan urutan) dan *ordered list* (list yang memperhatikan urutan). Contohnya yaitu:



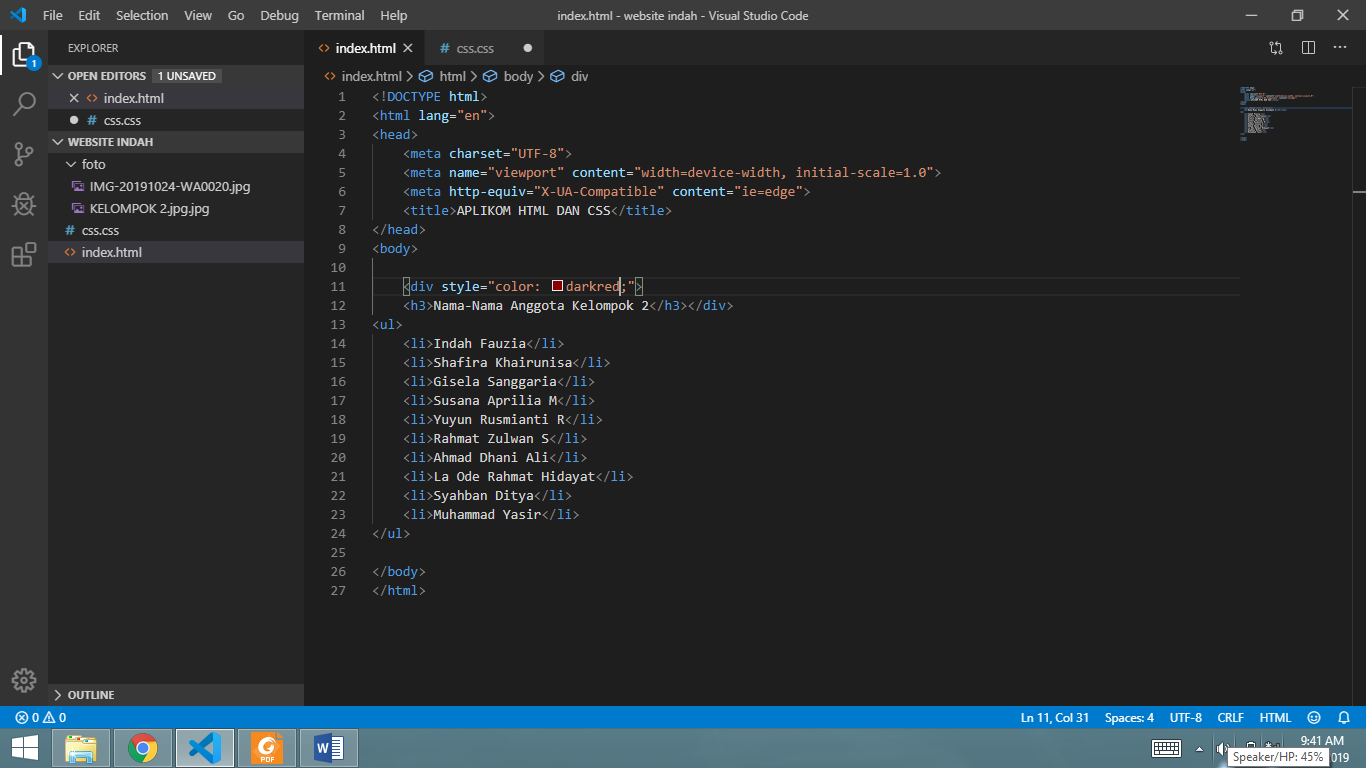
Gambar 26. Membuat list

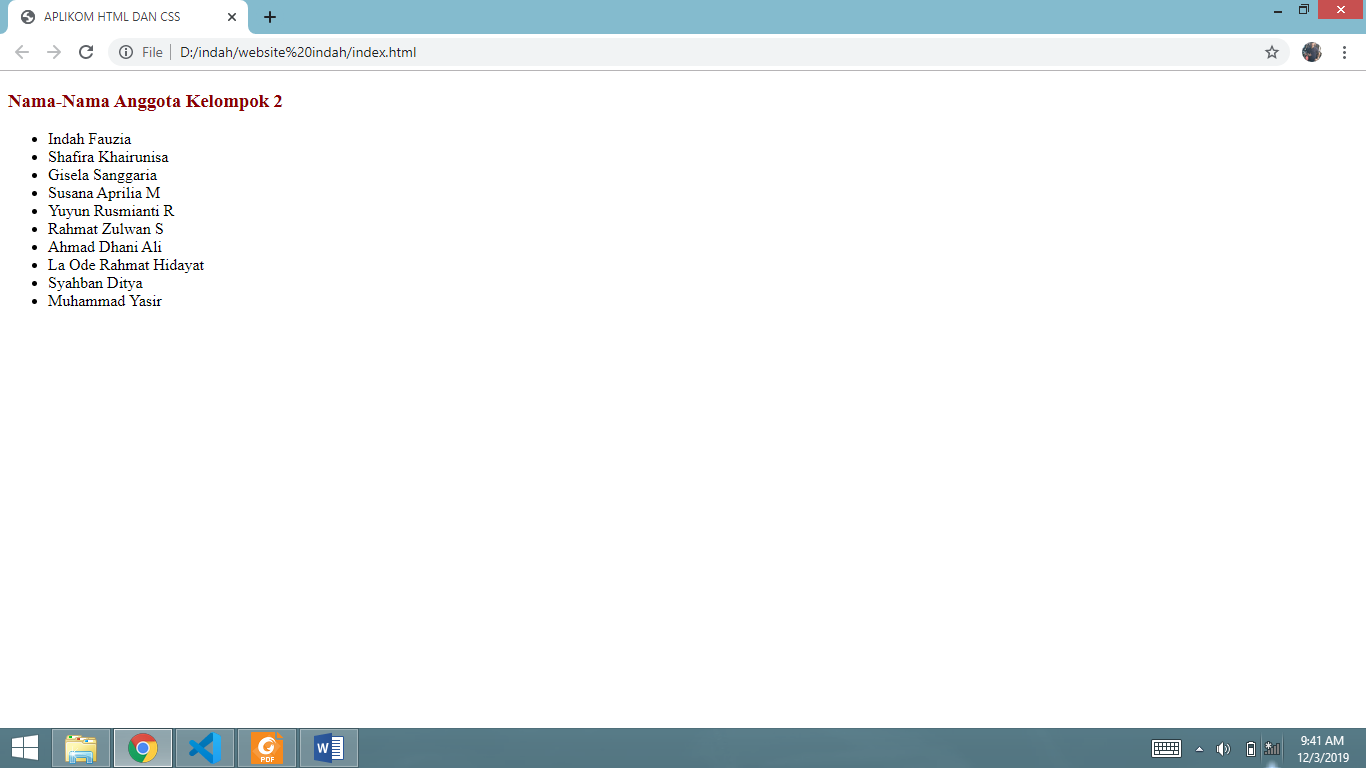
Gambar 3.27 Tampilan List

Untuk metode listing dari kedua jenis list tersebut kurang lebih sama, hanya yang membedakan adalah untuk *unordered list* menggunakan tag <ul> sedangkan untuk *ordered list* menggunakan <ol>. Untuk menunjukan poin dalam list tersebut kita menggunakan tag <li>.

## Mengenal tag <div>

Tag <div> adalah sebuah tag yang menandakan suatu bagian tertentu dalam dokumen HTML. Tag ini sering digunakan untuk menentukan suatu bagian tertentu pada sebuah website. Oleh karena itu, tag ini sering dipadukan dengan CSS untuk melakukan pemformatan (styling) sebuah dokumen HTML. Contoh penggunaan tag ini yaitu:

Gambar 3.28 Contoh tag <div>



Gambar 3.29 Tampilan hasil <div>

* 1. **Penerapan CSS Pada HTML**

CSS biasanya disimpan di dalam sebuah file berekstensi .css dan disematkan di dalam dokumen HTML untuk memberikan style pada halaman tersebut. Meski demikian, kita tetap dapat menambahkan sintaks CSS langsung di dalam dokumen HTML, di dalam tag <style></style> untuk mengakomodir penerapan style yang hanya digunakan di dalam dokumen tersebut. Kita juga dapat menyematkan sintaks CSS langsung di dalam atribut style pada suatu tag HTML untuk menerapkan style yang hanya digunakan pada tag itu saja. Berikut adalah contoh penerapan ketiga mode tersebut :

* + - 1. Dokumen yang disematkan file style.css:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<link rel="stylesheet" type="text/css" href="style.css" />

</head>

<body>

<h1>Selamat Datang, Programmer!</h1>

<p>Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

* + - 1. Dokumen dengan CSS di dalamnya:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<style>

h1 { text-align: center; color: lightsteelblue; }

</style>

</head>

<body>

<h1>Selamat Datang, Programmer!</h1>

<p>Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

* + - 1. Dokumen dengan CSS pada tag:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

</head>

<body>

<h1 style="text-align:center; color:lightsteelblue;">Selamat Datang, Programmer!</h1>

<p>Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

* 1. Sintaks CSS

Sekarang mari kita perhatikan sintaks penulisan kode CSS:

h1 { text-align: center; color: red; }

Pada contoh kode di atas:

h1 -> selector

{ text-align: center; color: red; } -> deklarasi style

text-align -> property

center -> value

color -> property

red-> value

Selector adalah bagian CSS untuk merujuk elemen HTML yang ingin dikenakan style. Ada beberapa jenis selector yang dapat kita gunakan untuk memudahkan pekerjaan styling kita pada CSS, seperti selector element, selector class, dan selector id.

Pada kode di atas elemen yang dirujuk oleh kode CSS kita adalah <h1>. Elemen <h1> ini kita buat supaya tulisannya rata tengah, maka kita beri property text-align dengan value center. Property dan value CSS dipisahkan dengan tanda titik dua (:). Kita dapat memberikan lebih dari satu property style dengan memisahkan antar property dengan tanda titik koma (;).  property dan value disimpan di dalam kurung kurawal { dan }.

* + - 1. Selector Element

Selector ini mengacu elemen HTML berdasarkan nama elemennya. Pada contoh berikut:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<style>

h1 { color: red; }

p { color: blue; }

</style>

</head>

<body>

<h1>Welcome</h1>

<h1>Selamat Datang, Programmer!</h1>

<p>Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

Pada contoh diatas itu berarti kita menerapkan style color : red pada semua elemen <h1> yang ada di dokumen dan menerapkan style color:blue pada semua elemen <p>.

* + - 1. Selector class

Selector ini mengacu elemen berdasarkan nama classnya. Nama class pada elemen diacu dengan nilai atribut class diawali oleh tanda titik (.). Pada contoh berikut:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<style>

.merah { color: red; }

</style>

</head>

<body>

<h1>Welcome</h1>

<h1 class="merah">Selamat Datang, Programmer!</h1>

<p class="merah">Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

pada dokumen di atas semua elemen yang memiliki atribut class merah yakni elemen <h1> yang kedua dan <p> akan berwarna merah. Elemen <h1> yang pertama tidak dikenai style karena tidak memiliki atribut class="merah".

Contoh lain:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<style>

h1.merah { color:red; }

</style>

</head>

<body>

<h1>Welcome!</h1>

<h1 class="merah">Selamat Datang, Programmer!</h1>

<p class="merah">Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

pada dokumen di atas, elemen yang teksnya berwarna merah hanyalah elemen <h1> yang kedua karena selector h1.merah berarti hanya memilih elemen <h1> yang memiliki atribut class="merah" saja. Elemen <p> juga tidak dikenai style meskipun memiliki atribut class merah.

Bandingkan contoh di atas dengan contoh di bawah ini:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<style>

h1 .merah { color:red; }

</style>

</head>

<body>

<h1>Welcome!</h1>

<h1 class="merah">Selamat Datang, <span class="merah">Programmer</span> !</h1>

<p class="merah">Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

Pada contoh terakhir ini, teks yang berwarna merah hanya yang bertuliskan Programmer saja. Perhatikan selector pada CSS di atas, terdapat spasi antara h1 dan .merah. Spasi antar selector berarti mengacu pada elemen turunannya. h1 .merah berarti memilih elemen berclass merah yang ada di dalam elemen h1.

* + - 1. Selector id

Selector ini mengacu elemen berdasarkan nama idnya. Nama id pada elemen diacu dengan nilai atribut id diawali oleh tanda pagar (#). Pada contoh berikut:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<style>

#judul-utama { color:red; }

#subjudul { color:blue; }

</style>

</head>

<body>

<h1 id="judul-utama">Welcome!</h1>

<h2 id="subjudul">Selamat Datang, Programmer!</h2>

<p>Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

elemen <h1> akan berwarna merah dan <h2> akan berwarna biru. Perhatikan lagi contoh ini:

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<style>

#hijau { color:green; }

</style>

</head>

<body>

<h1 id="hijau">Welcome!</h1>

<h1 id="hijau">Selamat Datang, Programmer!</h1>

<p id="hijau">Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

Pada contoh di atas, hanya elemen <h1> yang pertama saja yang berwarna hijau. Berbeda dengan selector class, selector id bersifat unik, artinya mesti hanya ada satu id spesifik dalam sebuah dokumen. Apabila ada dua atau lebih elemen yang ber-id sama seperti contoh di atas, maka hanya elemen yang ditemukan pertama kali (dari atas) pada dokumen saja yang dikenai style.

Untuk mengenakan style ke banyak elemen, gunakanlah class.

* + - 1. Menggabung Selector

Kita dapat mengenakan style yang sama kepada berbagai elemen, dengan memisah antar selector dengan tanda koma (,).

<!DOCTYPE html>

<html>

<head>

<title>HTML dengan CSS</title>

<style>

h1, p { color:#333; text-align:center; }

</style>

</head>

<body>

<h1>Selamat Datang, Programmer!</h1>

<p>Selamat belajar pemrograman di CodePolitan</p>

</body>

</html>

Pada contoh di atas,  semua elemen <h1> dan <p> pada dokumen akan dikenai style yang sama.

**BAB IV**

**PENUTUP**

* 1. **Kesimpulan**

Dari penjelasan materi tentang HTML dan berkekmbang menjadi HTML5 serta CSS yang berkembang menjadi CSS3. HTML lebih dikenal sebagai standar Bahasa yang membuat dokumen web sedangkan HTML versi 5 justru tidak dibuat untuk mempublikasikan informasi di web, namun oleh karena kesederhanaan dan kemudahan penggunanya, HTML kemudian dipilih orang untuk mendistribusikan informasi di web. Sedangkan untuk CSS, CSS bukan termasuk Bahasa pemrograman namun CSS berperan dalam mengendalikan ukuran gambar, warna, ukuran border serta desain lainnya. Namun CSS digunakan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur.

* 1. **Saran**

Adapun saran yang dapat kami berikan terhadap praktikum pemrograman web (HTML dan CSS) yaitu Sebaiknya sebelum Praktikum, masing-masing kelompok sudah mempraktekkan apa yang ada di Modul, jika ada kesalahan maka bisa ditanyakan saat praktikum.

**DAFTAR PUSTAKA**

Abror, muhamad. pengertian pemrograman menurut para ahli di <https://www.ayoksinau.com/pengertian-program-menurut-para-ahli-lengkap/>. Di akses pada tanggal 25 oktober 2019 pada pukul 19.41 WITA

Anonim, 2012. Pengertian dari pemrograman web <https://sevima.com/pengenalan-pemograman-web/>. Di akses pada 15 februari 2019 pada pukul 17.18 WITA

Anonim, 2013. Sejarah dan perkembangan CSS <https://cahyonex.wordpress.com/2013/11/19/sejarah-dan-perkembangan-css/>. Di akses pada 19 agustus 2018 pada pukul 18.46 WITA

Aji, Damar Yosa. sejarah dan perekembangan HTML <https://www.codepolitan.com/sejarah-dan-perkembangan-html5-belajar-html>. Di akses pada tanggal 25 mei 2018 pada pukul 19.34 WITA

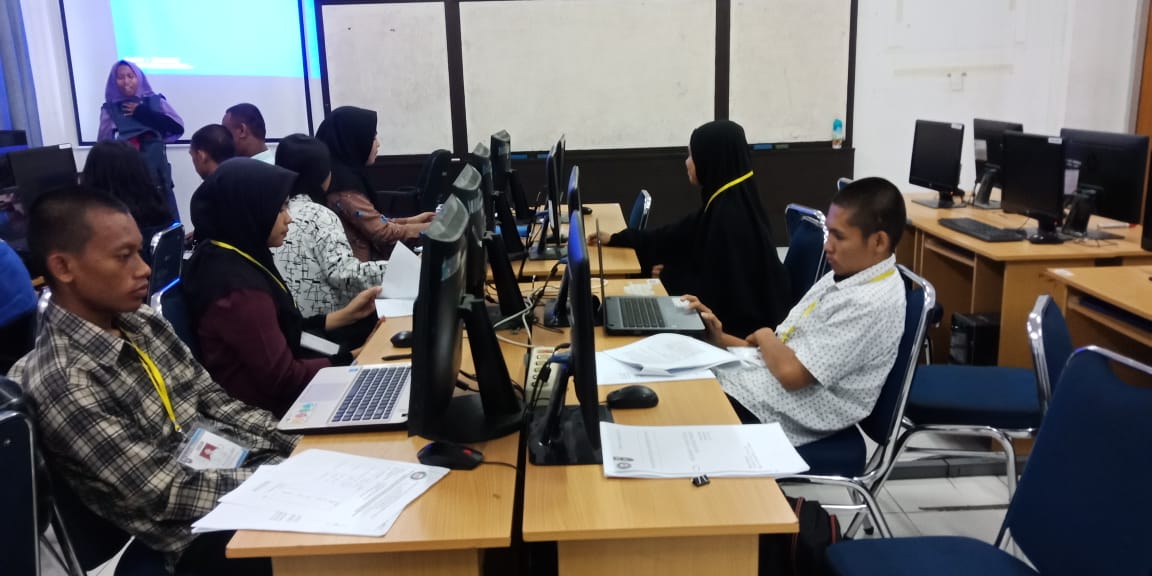
Indri. Sejarah HTML kelebijan dan kekurangannya <http://ngemasfizar.ilearning.me/2013/09/05/sejarah-html-dan-kelebihan-serta-kekurangannya/>. Di akses pada tanggal 5 april 2019 pada pukul 19.21 WITA

Eko. Fungsi HTML <http://www.fungsiklopedia.com/fungsi-html/>. Di akses pada tanggal 27 mei 2019 pada pukul 20.15 WITA

Anonim 2018 belajar CSS pengertian, peran, fungsi, dan macamnya <https://www.dewaweb.com/blog/belajar-css-pengertian-peran-fungsi-dan-macamnya/>. Pada tanggal 21 maret 2019 pada pukul 20.47 WITA

Haryanto, toni. Pengenalan dasar sintaks CSS <https://www.codepolitan.com/pengenalan-dan-sintaks-dasar-css>. Pada tanggal 13 juli 2018 pada pukul 21.07 WITA

**Lampiran**



Gambar pelaksaan praktikum aplikasi komputer pemrograman web (HTML dan CSS)

****

Gambar anggota kelompok 2

