



● Introduction

HTML merupakan standart umum yang digunakan untuk membangun sebuah halaman website. HTML merupakan kepanjangan dari *Hyper Text Markup Language*, merupakan:

- Menggambarkan struktur halaman website.
- Terdiri dari serangkaian element, dimana element HTML akan memberitahukan browser untuk menampilkan konten dari berbagai platform komputer manapun.
- Mengontrol tampilan dari halaman website dan kontennya. Seperti memberikan label potongan konten dengan menggunakan element HTML seperti "*bagian judul*", "*bagian paragraf*", "*bagian tautan*" dan lain sebagainya.

Sifat dari HTML ialah client script dimana dokumen tersebut dapat dibuka didalam komputer secara stand alone yang tidak membutuhkan sebuah server untuk dapat menampilkannya di dalam browser. Dokument HTML biasanya berekstensi .html atau .htm dimana bahasa HTML tersebut tersusun atas tag atau syntax yang berformat <tagname>...</tagname>. Aturan dalam penulisan HTML sebagai berikut:

- Dalam penulisan tag HTML diapit dengan menggunakan dua kurung siku (*angle bracket*) "<" dan ">"
- Tag HTML selalu berpasangan, dimana tag awal dalam suatu syntax disebut dengan tag awal dan tag kedua sebagai tag terakhir. contoh: <p>...</p>
- Penulisan tag HTML tidak case sensitive dimana tag dengan huruf kecil sama dengan tag huruf besar, contoh: <P>...</P> sama dengan <p>...</p>
Namun berdasarkan aturan pada W3C (HTML5.0 dan XHTML) penulisan tag HTML sebaiknya menggunakan huruf kecil.
- Jika memiliki tag nested atau terdapat suatu tag berisi tag lagi, maka penulisan tag akhir tidak boleh bersilang dan harus berurutan. Contoh: <p>Ini contoh Huruf Tebal dan <i>Huruf cetak miring</i> </p>
- Penulisan syntax dokumen HTML selalu diawali dengan tag <html> dan diakhiri dengan tag </html>

SIMPLE DOCUMENT HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```



I B I K

Matakuliah/Code : Lab. Pemrograman Web / TIFA2Q0
Dosen : Irvan Rizky Ariansyah / Raiyana Jan Winata
Kelas : TI-21-PA & TI-21-KA

Dari contoh penulisan dokumen HTML diatas dapat diartikan sebagai berikut:

- Deklarasi `<!DOCTYPE html>` mendefinisikan bahwa dokumen ini adalah dokumen HTML5
- Elemen `<html>` adalah elemen root dari halaman HTML
- Elemen `<head>` berisi informasi meta tentang halaman HTML
- Elemen `<title>` menentukan judul untuk halaman HTML (yang ditampilkan di bilah judul browser atau di tab halaman)
- Elemen `<body>` mendefinisikan tubuh dokumen, dan merupakan wadah untuk semua konten yang terlihat, seperti judul, paragraf, gambar, hyperlink, tabel, daftar, dll.
- Elemen `<h1>` mendefinisikan heading besar
- Elemen `<p>` mendefinisikan paragraf

STRUKTUR HTML



• Editor

Dalam membangun sebuah halaman website dengan menggunakan HTML, *text editor* yang dapat digunakan cukup sederhana dimana aplikasi *text editor* ini terdapat di semua operating sistem pada komputer. Jika menggunakan Windows OS dapat menggunakan aplikasi *Notepad*, sedangkan jika menggunakan UNIX OS dapat menggunakan aplikasi *TextEdit*.

Untuk membuat sebuah dokumen HTML tahap pertama ialah buka *text editor* yang dimiliki seperti *Notepad* dan masukan *script* dibawah ini, lalu simpan dengan nama `index.html` atau `index.htm`.

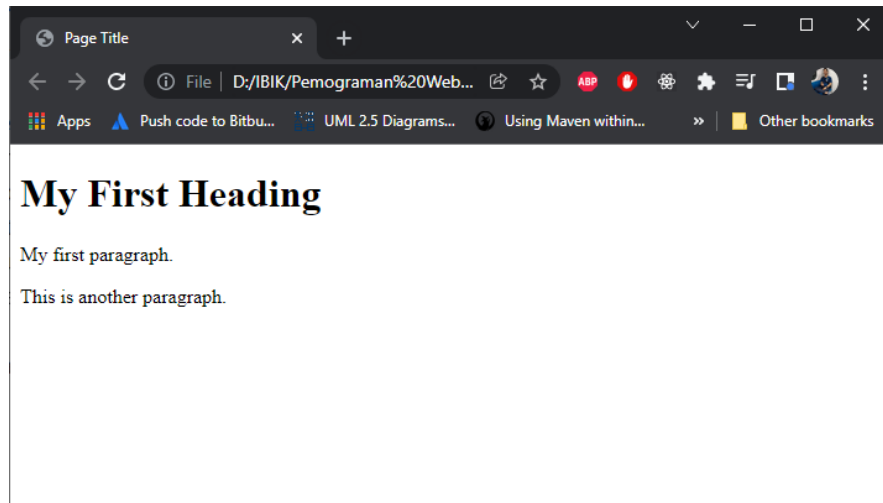


IBIK

Matakuliah/Code : Lab. Pemrograman Web / TIFA2Q0
Dosen : Irvan Rizky Ariansyah / Raiyana Jan Winata
Kelas : TI-21-PA & TI-21-KA

```
index.html - Notepad
File Edit Format View Help
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>
<h1>My First Heading</h1>
<p>My first paragraph.</p>
<p>This is another paragraph.</p>
</body>
</html>
Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

Setelah berhasil tersimpan dengan format extension .html atau .htm, maka file tersebut akan membentuk sebuah file dengan format icon browser. Jika dibuka file tersebut maka akan membuka sebuah aplikasi browser dan menampilkan sebuah tampilan berdasarkan tag HTML.



- **Document Head**

Berikut adalah tag atau elemen yang dapat digunakan dibagian `<head>...</head>` pada sebuah dokumen HTML:

1. Title

Elemen `<title>`, menentukan judul untuk halaman HTML (yang ditampilkan di bilah judul browser atau di tab halaman)

`<title>Page Title</title>`

2. Favicon

Elemen ini diperuntukan untuk menentukan icon pada halaman HTML (yang ditampilkan di bilah judul browser atau di tab halaman). Untuk menggunakan favicon, dikhususkan memiliki file gambar dengan extension .ico, .svg atau .png dan size yang digunakan ialah 16x16 pixel atau 32x32 pixel.



```
<link rel="icon" type="image/x-icon" href="/path/to/icons/favicon.ico">  
<link rel="icon" type="image/png" href="/path/to/icons/favicon.png">  
<link rel="icon" type="image/svg" href="/path/to/icons/favicon.svg">
```

3. Meta

Elemen `<meta>` biasanya digunakan untuk menentukan set karakter, deskripsi halaman, kata kunci, penulis dokumen, dan pengaturan *viewport*. Metadata tidak akan ditampilkan di halaman, tetapi digunakan oleh browser (cara menampilkan konten atau memuat ulang halaman), oleh mesin pencari (kata kunci), dan layanan web lainnya.

- Menentukan set karakter

```
<meta charset="UTF-8">
```

- Menentukan kata kunci untuk search engine

```
<meta name="keywords" content="HTML, CSS, JavaScript">
```

- Menentukan deskripsi halaman web

```
<meta name="description" content="Website IBIK - Pemograman Web">
```

- Menentukan penulis halaman

```
<meta name="author" content="Febry D Fairuz">
```

- Merefresh halaman setiap 30 detik

```
<meta http-equiv="refresh" content="30">
```

- Mengatur *viewport* agar situs website terlihat bagus di semua perangkat

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

4. Javascript

JavaScript membuat halaman HTML lebih dinamis dan interaktif. Tag HTML `<script>` digunakan untuk mendefinisikan skrip disisi klien (*JavaScript*). Elemen `<script>` berisi pernyataan skrip, atau menunjuk ke file skrip eksternal melalui atribut `src`.

Penggunaan umum untuk *JavaScript* adalah manipulasi gambar, validasi formulir, dan perubahan konten yang dinamis. Untuk memilih elemen HTML, *JavaScript* paling sering menggunakan metode `document.getElementById()`.

Contoh *JavaScript* ini menulis "Halo JavaScript!" menjadi elemen HTML dengan `id="demo"`:

```
<p id="demo"></p>
```

```
<script>
  document.getElementById("demo").innerHTML = "Hello JavaScript!";
</script>
```

5. Styles CSS

Cascading Style Sheets (CSS) digunakan untuk memformat tata letak halaman web. Dengan CSS, dapat mengontrol warna, font, ukuran teks, jarak antar elemen, bagaimana elemen diposisikan dan ditata, gambar latar belakang atau warna latar apa yang akan digunakan, tampilan berbeda untuk perangkat dan ukuran layar yang berbeda, dan lebih banyak!

```
<style>
  body {background-color: powderblue;}
  h1   {color: blue;}
  p    {color: red;}
</style>
```

- **Attribute Body**

Sedangkan berikut ini adalah tag atau elemen yang bisa digunakan dibagian `<body>...</body>` pada sebuah dokumen HTML:

- **Elements Contents HTML**

1. Headings

Headings adalah judul atau subjudul yang ingin ditampilkan di halaman web. *Headings* didefinisikan dengan tag `<h1>` hingga `<h6>`. `<h1>` mendefinisikan heading yang paling penting. `<h6>` mendefinisikan heading yang paling tidak penting.

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

2. Paragraphs

Sebuah paragraf selalu dimulai pada baris baru, dan biasanya berupa blok teks. Elemen HTML `<p>` mendefinisikan sebuah paragraf. Sebuah paragraf selalu dimulai pada baris baru, dan browser secara otomatis menambahkan beberapa spasi (margin) sebelum dan sesudah paragraf.

```
<p>This is a paragraph.</p>  
<p>This is another paragraph.</p>
```

3. Formatting

HTML berisi beberapa elemen untuk mendefinisikan teks dengan arti khusus. Elemen pemformatan dirancang untuk menampilkan jenis teks khusus:

Element	Description
<code></code>	Bold
<code></code>	Important
<code><i></code>	Italic
<code></code>	Emphasized
<code><mark></code>	Marked
<code><small></code>	Smaller
<code></code>	Deleted text
<code><sub></code>	Subscript text
<code><sup></code>	Superscript text

4. Quotations

Bagian ini kita akan membahas elemen HTML `<blockquote>`, `<q>`, `<abbr>`, `<address>`, `<cite>`, dan `<bdo>`.

Element	Description
<code><abbr></code>	singkatan atau akronim
<code><address></code>	informasi kontak untuk penulis/pemilik dokumen
<code><bdo></code>	arah teks
<code><blockquote></code>	bagian yang dikutip dari sumber lain
<code><cite></code>	judul sebuah karya
<code><q></code>	kutipan inline pendek

5. Comments

Untuk menambahkan komentar ke sumber HTML Anda dengan menggunakan sintaks berikut:

```
<!-- Write your comments here -->
```

6. List

List HTML memungkinkan programmer untuk mengelompokkan satu set item terkait dalam bentuk list. Daftar tidak berurutan dimulai dengan tag `` atau ``. Setiap item list dimulai dengan tag ``.

Syntax	Display
<pre> Coffee Tea Milk </pre>	<ul style="list-style-type: none">• Coffee• Tea• Milk
<pre> Coffee Tea Milk </pre>	<ol style="list-style-type: none">1. Coffee2. Tea3. Milk

7. Block and Inline

Setiap elemen HTML memiliki nilai tampilan default, tergantung pada jenis elemennya. Ada dua nilai tampilan: block dan inline.

- Block Element

Elemen level blok selalu dimulai pada baris baru. Elemen level blok selalu menggunakan lebar penuh yang tersedia (membentang ke kiri dan kanan sejauh mungkin). Elemen level blok memiliki margin atas dan bawah, sedangkan elemen inline tidak.

```
<div>Hello World</div>
```

- Inline Element

Elemen Inline tidak dimulai pada baris baru. Elemen Inline hanya membutuhkan lebar sebanyak yang diperlukan. Ini adalah elemen `` di dalam paragraf.

```
<span>Hello World</span>
```

8. Links

Links ditemukan di hampir semua halaman web. Links memungkinkan pengguna mengklik jalan mereka dari halaman ke halaman lain. Links HTML disebut dengan *hyperlink*. Untuk dapat mengklik tautan dan melompat ke halaman lain. Saat Anda menggerakkan mouse di atas tautan, panah mouse akan berubah menjadi tangan kecil.

Atribut terpenting dari elemen `<a>` adalah atribut `href`, yang menunjukkan tujuan tautan. Teks tautan adalah bagian yang akan terlihat oleh pembaca. Mengklik teks tautan, akan mengirim pembaca ke alamat URL yang ditentukan.

```
<a href="https://www.ibik.ac.id/"> Visit IBIK</a>
```

HTML Links - The target Attribute

Secara default, halaman tertaut akan ditampilkan di jendela browser saat ini. Untuk mengubahnya, Anda harus menentukan target lain untuk tautan tersebut.

Atribut `target` dapat memiliki salah satu dari nilai berikut:

Target	Description
<code><_self></code>	Membuka windows atau tab dalam satu halaman yang sama
<code><_blank></code>	Membuka windows atau tab baru
<code><_parent></code>	Membuka windows atau tab didalam elemen induk
<code><_top></code>	Membuka windows atau tab secara keseluruhan pada body

Create Bookmarks

Link HTML dapat digunakan untuk membuat bookmark, sehingga pembaca dapat melompat ke bagian tertentu dari halaman web.

```
<h2 id="C4">Chapter 4</h2>  
<a href="#C4">Jump to Chapter 4</a>
```

9. Images

Gambar dapat meningkatkan desain dan tampilan halaman web. Tag HTML `` digunakan untuk menyematkan gambar di halaman web.

Gambar secara teknis tidak dimasukkan ke dalam halaman web; gambar ditautkan ke halaman web. Tag `` membuat ruang penyimpanan untuk gambar yang direferensikan. Tag `` kosong, hanya berisi atribut, dan tidak memiliki tag penutup. Tag `` memiliki dua atribut yang diperlukan:

- `src` = Menentukan jalur ke gambar
- `alt` = Menentukan teks alternatif untuk gambar.
- `width` = Menentukan panjang
- `height` = Menentukan tinggi



```

```

10. Tables

Tabel HTML memungkinkan pengembang web untuk mengatur data ke dalam baris dan kolom.

Table Kolom

```
<table>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
</table>
```

Each table cell is defined by a `<td>` and a `</td>` tag. Segala sesuatu di antara `<td>` dan `</td>` adalah konten sel tabel.

Table Baris

```
<table>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
  <tr>
    <td>16</td>
    <td>14</td>
    <td>10</td>
  </tr>
</table>
```

Setiap baris tabel dimulai dengan `<tr>` dan diakhiri dengan tag `</tr>`. Berikut ini adalah tag elemen table yang dapat dipergunakan:

Element	Description
<code><table></code>	Mendefinisikan tabel
<code><th></code>	Mendefinisikan sel header dalam tabel
<code><tr></code>	Mendefinisikan baris dalam tabel
<code><td></code>	Mendefinisikan sel dalam tabel
<code><caption></code>	Mendefinisikan judul tabel



Matakuliah/Code : Lab. Pemrograman Web / TIFA2Q0
Dosen : Irvan Rizky Ariansyah / Raiyana Jan Winata
Kelas : TI-21-PA & TI-21-KA

<code><colgroup></code>	Menentukan grup dari satu atau lebih kolom dalam tabel untuk pemformatan
<code><col></code>	Menentukan properti kolom untuk setiap kolom dalam elemen <code><colgroup></code>
<code><thead></code>	Mengelompokkan konten header dalam sebuah tabel
<code><tbody></code>	Mengelompokkan konten tubuh dalam sebuah tabel
<code><tfoot></code>	Mengelompokkan konten footer dalam tabel

11. Iframe

Tag HTML `<iframe>` menentukan bingkai sebaris. Bingkai sebaris digunakan untuk menyematkan dokumen lain dalam dokumen HTML saat ini.

```
<iframe src="url" title="description" width="100px"
height="50px"></iframe>
```

12. Layouts

▪ Element Content Forms

Form HTML digunakan untuk mengumpulkan input data. Input data paling sering dikirim ke server untuk diproses.

Elemen HTML `<form>` digunakan untuk membuat formulir HTML untuk input pengguna:

First name: <input type="text" value="Febry"/>	<pre><form> <label for="fname">First name:</label>
 <input type="text" id="fname" name="fname">
 <label for="lname">Last name:</label>
 <input type="text" id="lname" name="lname"> </form></pre>
Last name: <input type="text" value="Fairuz"/>	
<input type="submit" value="Submit"/>	

1. Form Attributes

List atribut `<form>` yang dapat digunakan:

Attribute	Description
<code>accept-charset</code>	Menentukan pengkodean karakter yang digunakan untuk pengiriman formulir
<code>action</code>	Menentukan ke mana harus mengirim formulir-data saat formulir dikirimkan
<code>autocomplete</code>	Menentukan apakah formulir harus memiliki pelengkapan otomatis aktif atau nonaktif
<code>enctype</code>	Menentukan bagaimana formulir-data harus dikodekan saat mengirimkannya ke server (hanya untuk metode = "posting")
<code>method</code>	Menentukan metode HTTP yang akan digunakan saat mengirim data formulir
<code>name</code>	Menentukan nama formulir
<code>novalidate</code>	Menentukan bahwa formulir tidak boleh divalidasi saat dikirimkan
<code>rel</code>	Menentukan hubungan antara sumber daya yang ditautkan dan dokumen saat ini
<code>target</code>	Menentukan tempat untuk menampilkan respons yang diterima setelah mengirimkan formulir

2. Form Elements

Elemen `<form>` HTML dapat berisi satu atau lebih elemen form sebagai berikut:

- Elemen `<input>`

Salah satu elemen form yang paling sering digunakan adalah elemen `<input>`. Elemen `<input>` dapat ditampilkan dalam beberapa cara, tergantung pada atribut `type`.

```
<label for="fname">First name:</label>
<input type="text" id="fname" name="fname">
```

- Elemen `<label>`

Elemen `<label>` mendefinisikan label untuk beberapa elemen form. Elemen `<label>` berguna untuk pengguna pembaca layar, karena pembaca layar akan membacakan label dengan keras saat pengguna fokus pada elemen input.

Elemen `<label>` juga membantu pengguna yang mengalami kesulitan mengklik wilayah yang sangat kecil (seperti tombol radio atau kotak centang) - karena ketika pengguna mengklik teks di dalam elemen `<label>`, itu akan mengaktifkan tombol radio/kotak centang. Atribut `for` dari tag `<label>` harus sama dengan atribut `id` dari elemen `<input>` untuk mengikatnya bersama.

- Elemen `<textarea>`

Elemen `<textarea>` mendefinisikan bidang input multi-baris (area teks):

```
<textarea name="message" rows="10" cols="30">
Type something in here
</textarea>
```

Atribut `rows` menentukan jumlah baris yang terlihat di area teks. Atribut `cols` menentukan lebar yang terlihat dari area teks.

- Elemen `<button>`

Elemen `<button>` mendefinisikan tombol yang dapat diklik:

```
<button type="button">Click Me! Button</button>
<button type="submit">Click Me! Submit</button>
<button type="reset">Click Me! Reset</button>
```

- Elemen `<select>`

Elemen `<select>` mendefinisikan daftar drop-down:

```
<label for="cars">Choose a car:</label>
<select id="cars" name="cars">
  <option value="volvo">Volvo</option>
```

```
<option value="saab">Saab</option>
<option value="fiat">Fiat</option>
<option value="audi">Audi</option>
</select>
```

The `<option>` elements defines an option that can be selected. By default, the first item in the drop-down list is selected. To define a pre-selected option, add the selected attribute to the option:

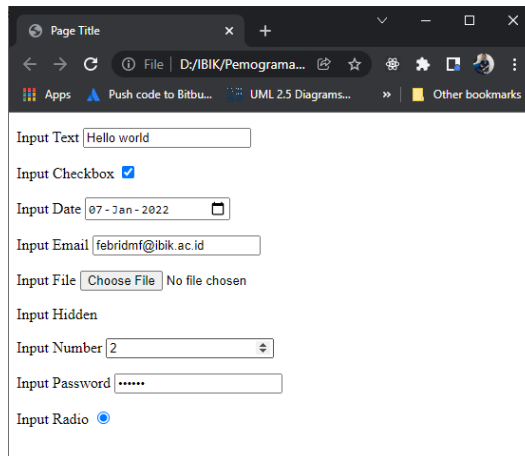
```
<option value="fiat" selected>Fiat</option>
```

3. Input Types

Berikut adalah berbagai jenis input yang dapat digunakan dalam HTML:

<pre><input type="text"> <input type="checkbox"> <input type="date"> <input type="email"> <input type="file"></pre>	<pre><input type="hidden"> <input type="number"> <input type="password"> <input type="radio"> <input type="range"></pre>
---	--

Display:



4. Input Attributes

Bagian ini menjelaskan atribut yang berbeda untuk elemen `<input>` HTML.

Attribute	Description
value	Menentukan nilai awal untuk bidang input
readonly	Menentukan bahwa bidang input hanya-baca
disabled	Menentukan bahwa bidang input harus dinonaktifkan
size	Menentukan lebar yang terlihat, dalam karakter, dari bidang input
maxlength	Menentukan jumlah karakter maksimum yang diizinkan dalam bidang input

min & mun	Menentukan nilai minimum dan maksimum untuk bidang input
pattern	Menentukan ekspresi reguler yang memeriksa nilai bidang input, saat formulir dikirimkan.
placeholder	Menentukan petunjuk singkat yang menjelaskan nilai yang diharapkan dari bidang input (nilai sampel atau deskripsi singkat tentang format yang diharapkan)
required	Menentukan bahwa bidang input harus diisi sebelum mengirimkan formulir

- **Styles CSS**

1. Inline

Atribut *style* digunakan untuk menambahkan gaya ke elemen, seperti warna, font, ukuran, dan lainnya. Pengaturan *style* pada suatu elemen, dapat dilakukan dengan atribut *style*. Atribut *style* HTML memiliki sintaks berikut:

```
<tagname style="property:value;">
```

Properti adalah properti CSS. *Value* adalah nilai CSS.

Contoh penggunaan *css inline*:

```
<h1 style="text-align:center;">Centered Heading</h1>
<p style="text-align:center;">Centered paragraph.</p>

<h1 style="font-size:300%;">This is a heading</h1>
<p style="font-size:160%;">This is a paragraph.</p>

<h1 style="font-family:verdana;">This is a heading</h1>
<p style="font-family:courier;">This is a paragraph.</p>

<h1 style="color:blue;">This is a heading</h1>
<p style="color:red;">This is a paragraph.</p>

<h1 style="background-color:blue;">This is a heading</h1>
<p style="background-color:tomato;">This is a paragraph.</p>
```

2. Internal

Kode CSS internal diletakkan di dalam bagian `<head>` pada halaman. *Class* dan *ID* bisa digunakan untuk merujuk pada kode CSS, namun hanya akan aktif pada halaman tersebut. Style CSS yang dipasang dengan metode ini akan di-download setiap kali halaman dipanggil, jadi ini akan meningkatkan kecepatan akses.

Namun, ada beberapa kasus dimana penggunaan internal *stylesheet* justru berguna. Salah satu contohnya adalah untuk mengirimkan template halaman ke seseorang – karena semuanya bisa



terlihat dalam 1 halaman, maka akan lebih mudah untuk melihat previewnya. CSS internal diletakkan di dalam tag `<style></style>`. Contohnya:

```
<head>
<style>
  body {background-color: blue;}
  h1   {color: blue; font-family:verdana; font-family:courier; }
  p    {color: red;}
  .center {text-align: "center";}
  #card {border: 2px solid blue; margin: 50px;}
  #card > h1 {color: tomato}
  #card > p {background-color: pink}
</style>
</head>
```

```
<body>
<h1>Centered Heading</h1>
<p>Centered paragraph.</p>

<h1 class="center">This is a heading</h1>
<p class="center">This is a paragraph.</p>

<div id="card">
  <h1>This is a heading</h1>
  <p>This is a paragraph.</p>
</div>
</body>
```

3. External

Salah satu cara yang paling nyaman untuk menambahkan CSS ke website adalah dengan menghubungkannya ke file .CSS eksternal. Dengan cara tersebut, perubahan apapun yang Anda buat pada file CSS akan tampil pada website Anda secara keseluruhan. File CSS eksternal biasanya diletakkan setelah bagian `<head>` pada halaman:

```
<head>
  <link rel="stylesheet" href="styles.css">
</head>
```

File `styles.css`



```
body {background-color: blue;}  
h1 {color: blue; font-family:verdana; font-family:courier; }  
p {color: red;}  
.center {text-align: "center";}  
#card {border: 2px solid blue; margin: 50px;}  
#card > h1 {color: tomato}  
#card > p {background-color: pink}
```

- **Javascript**

1. Basic

JavaScript membuat halaman HTML lebih dinamis dan interaktif. Tag HTML `<script>` digunakan untuk mendefinisikan skrip sisi klien (JavaScript). Elemen `<script>` berisi pernyataan skrip, atau menunjuk ke file skrip eksternal melalui atribut `src`.

Penggunaan umum untuk JavaScript adalah manipulasi gambar, validasi formulir, dan perubahan konten yang dinamis. Untuk memilih elemen HTML, JavaScript paling sering menggunakan metode `document.getElementById()`. Contoh JavaScript ini menulis "Halo JavaScript!" menjadi elemen HTML dengan `id="demo"`:

```
<script>  
document.getElementById("demo").innerHTML = "Hello JavaScript!";  
</script>
```

2. Inline

```
<button type="button" onclick="alert('Hello World!')">Click  
Me!</button>
```

3. Internal

Kode Javascript internal diletakkan di antara elemen `<head>` dan elemen `<body>` pada halaman. *Class* dan *ID* bisa digunakan untuk merujuk pada kode Javascript, namun hanya akan aktif pada halaman tersebut. Javascript yang dipasang dengan metode ini akan di-download setiap kali halaman dipanggil, jadi ini akan meningkatkan kecepatan akses.

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>Page Title</title>
```

```
<script>
    document.getElementById("demo").innerHTML = "Hello
JavaScript!";
</script>
</head>
<body>

<h1>My First Heading</h1>
<p id="demo"></p>

</body>
</html>
```

4. External

Salah satu cara yang paling nyaman untuk menambahkan *Javascript* ke website adalah dengan menghubungkannya ke file *.JS* eksternal. Dengan cara tersebut, perubahan apapun yang Anda buat pada file *Javascript* akan tampil pada website Anda secara keseluruhan. File JS eksternal biasanya diletakkan setelah bagian `<head>` pada halaman:

```
<head>
    <script type="text/javascript" src="my-script.js"></script>
</head>
```




IBIK

Matakuliah/Code : Lab. Pemrograman Web / TIFA2Q0
 Dosen : Irvan Rizky Ariansyah / Raiyana Jan Winata
 Kelas : TI-21-PA & TI-21-KA

❖ LATIHAN PRAKTIKUM

1. Latihan *Typography*

Buatlah dokumen HTML bernama *Latihan-1*.

Buatlah sebuah artikel yang berisi *Heading*, *Paragraph* dan *Formatting*. Minimum penggunaan paragraf sebanyak 4 buah.

2. Latihan *Image*

Buatlah dokumen HTML bernama *Latihan-2*.

Dari dokumen HTML *Latihan-1*, tambahkanlah elemen *Image*. Penggunaan *image* minimum 2 buah. Dan tambahkan atribut gambar untuk mengatur panjang dan lebar dari gambar tersebut.

3. Latihan *Table*

Buatlah dokumen HTML bernama *Latihan-3*.

Silakan implementasikan bentuk dibawah ini kedalam scripting HTML:

Oksigen mempunyai beberapa SIFAT FISIKA, diantaranya yang terdapat dalam table berikut:

Sifat Fisika	Oksigen
Massa atom <i>relative</i>	15,994
Nomor Atom	8
Konfigurasi <i>electron</i>	2s ² 2p ⁴
Jari-jari atom (nm)	0,074
Jari-jari X ²⁻ (nm)	0.140
Keelectronegatifan	3,5
Energy ionisasi I (kJ/mol)	1316
Energy ionisasi II (kJ/mol)	3396
Kerapatan (g/cm ³)	1.27 (padatan)
Titik leleh (°C)	+183
Titik beku (°C)	-219
Potensial elektroda (V)	+0.401
X _{2(g)} + 2e ⁺ _(aq) ⇌ 2X ⁻ _(aq)	-

4. Latihan List



Buatlah dokumen bernama Latihan-4.

Ubahlah bentuk dibawah ini kedalam scripting HTML:

The 10 Most Popular Programming Languages to Learn in 2022

There's no question that software programming is a hot career right now. The **U.S. Bureau of Labor Statistics** projects 21 percent growth for programming jobs from 2018 to 2028, which is more than four times the average for all occupations. What's more, the median annual pay for a software programmer is about \$106,000, which nearly three times the median pay for all U.S. workers.

Not all programming jobs are the same, however. Different roles, companies, and types of software require knowing and understanding different programming languages—and it's often beneficial to know multiple languages. Trying to break into the field of software programming can be a daunting experience, especially for professionals with no prior programming experience.

The [Master of Science in Computer Science Align program](#) at Northeastern University is specifically designed for students who want to transition into computer science from another field of study.

"Our aim is to transport students from a variety of different backgrounds and have them come out as software engineers," says Ian Gorton, PhD and director of the graduate computer science programs at Northeastern University—Seattle. *"We focus on math, programming, and a variety of computer science and engineering concepts."*

Whether you're new to programming or looking to brush up on your skills, it helps to know which languages are in high demand. Here are 10 of the most popular programming languages of 2020 based on the number of job postings listed on job search site Indeed, the average annual salary for those jobs, and factors such as ease of use and potential for growth.

Top 10 Most Popular Programming Languages

1. Python
Average annual salary: \$120.000
2. JavaScript
Average annual salary: \$118.000
3. Java
Average annual salary: \$104.000
4. C#
Average annual salary: \$97.000
5. C
Average annual salary: \$97.000
6. C++

Average annual salary: \$97.000

7. Go

Average annual salary: \$93.000

8. R

Average annual salary: \$93.000

9. Swift

Average annual salary: \$93.000

10. PHP

Average annual salary: \$81.000

7 Other Programming Languages to Consider





- **Web-based** startups are more likely to be programming in **Python** and **JavaScript**.
- Larger companies tend to develop their internal software applications using **C#** or **Java** and their **Web applications** using **PHP**.
- Programs for data *analytics typically* use the **R** and **MATLAB** programming languages.
- Embedded devices, such as those in the automotive and healthcare industries, run software written in **C**, **C++**, or **Rust**.
- Applications that run on the cloud are increasingly written in **Go** or **Scala**.
- **Mobile applications** are increasingly written in **Swift** or **Kotlin**.

5. Latihan Iframe, Block dan Inline

Buatlah dokumen HTML bernama Latihan-5.

a. Ubahlah bentuk dibawah ini menjadi script HTML:

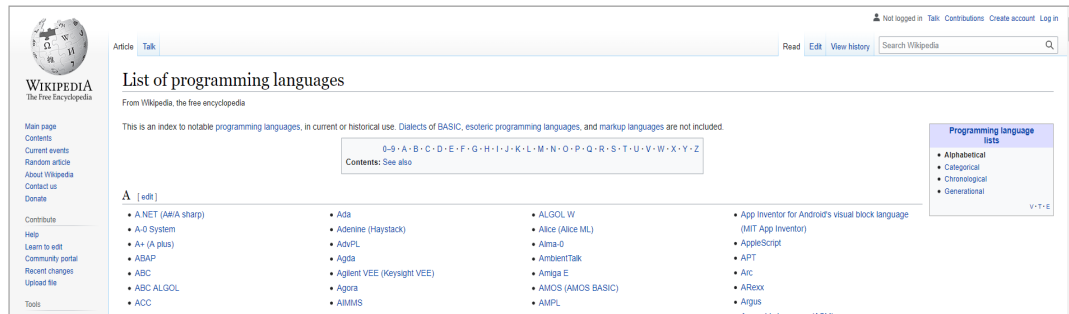
Programming Language Base on Career

Front-end Web Development	Back-end Web Development	Mobile Development
 JavaScript	 JavaScript	 JavaScript
 Elm	 Scala	 Java
 TypeScript	 Go	 Objective C
	 Ruby	 Swift
	 Python	

Top 10 Programming Language in 2022

...Iframe to Latihan-4...

List of programming languages from Wikipedia



- Buatlah 1 buah iframe yang memanggil file dokumen Latihan-4
- Buatlah 1 buah iframe yang memanggil sebuah url

6. Latihan Link

Buatlah dokumen HTML bernama Latihan-6.

- Implementasikan bentuk dibawah ini kedalam script HTML:

Top 10 Daftar Bahasa Pemograman

Daftar isi:

- [Python](#)
- [JavaScript](#)
- [Java](#)
- [C#](#)
- [C](#)
- [C++](#)
- [Go](#)
- [R](#)
- [Swift](#)
- [PHP](#)

Python

Python adalah ...



I B I K

Matakuliah/Code : Lab. Pemrograman Web / TIFA2Q0
Dosen : Irvan Rizky Ariansyah / Raiyana Jan Winata
Kelas : TI-21-PA & TI-21-KA

JavaScript

JavaScript adalah ...

...dan seterusnya...

Buatlah *hyperlink* pada elemen **Daftar Isi**, dimana link tersebut merujuk ke *diverent area*, ketika mengklik salah satu *link* tersebut maka akan merujuk ke area yang telah dituju. Contoh: jika mengklik *link Java*, maka akan dirujuk ke area *Java*.

b. Buatlah tiga buah link baru dimana:

- 1) Link "Go to Latihan-4" menuju ke dokumen HTML Latihan-4 dengan menggunakan target **_self**
- 2) Link "Go to Latihan-5" menuju ke dokumen HTML Latihan-5 dengan menggunakan target **_blank**
- 3) Link "Go to URL" menuju URL dengan menggunakan target **_parent**

7. Latihan Form

Buatlah dokumen HTML bernama *Latihan-7*.

Buatlah form yang berisi isian field sebagai berikut:

- NPM (input text)
- Fullname (input text)
- Email (input email)
- Password (input password)
- Place Birth (input text)
- Birthdate (input date)
- Gender (radio button, *Male and Female*)
- Address (textarea)

Dari form diatas memiliki dua buah button. Button submit dengan nama Save dan button reset dengan nama Clear.

8. Latihan Layout

Buatlah folder project dengan nama *my-portfolio*. Didalam folder tersebut terdapat dua buah folder yaitu folder *css* dan *images*. Dan terdapat satu buah file bernama *index.html*. Dimana file tersebut berisi script HTML yang akan menampilkan bentuk portfolio atau curriculum vitae pada *browser*. Silakan membuat bentuk portfolio dengan bentuk bebas namun mengimplementasikan bentuk elemen HTML yang telah diberikan minimal 5 buah elemen. Contoh: