|  |  |
| --- | --- |
|  | **DEPT OF COMPUTER AND COMMUNICATION ENGINEERING**  **TT KIẾN TRÚC VÀ GIAO THỨC IOT**  **ITAL328264** |
| **Group: Huỳnh Thiện Khải**  **Đỗ Bình Nguyên** | |

A collage of a person with glasses

Description automatically generated

# HTML (tag, features), CSS (style)

## HTML (30 tags)

### HTML HEADING

* Định nghĩa:
  + Dùng để viết các tiêu đề, các mục chính, phụ trong 1 trang web.
  + Có tất cả 6 loại tiêu đề như cấu trúc bên dưới
* Cấu trúc:

|  |  |
| --- | --- |
| **Syntax** | **Kết quả** |
| <h1>Tiêu đề 1</h1> | Tiêu đề 1 |
| <h2>Tiêu đề 2</h2> | Tiêu đề 2 |
| <h3>Tiêu đề 3</h3> | Tiêu đề 3 |
| <h4>Tiêu đề 4</h4> | Tiêu đề 4 |
| <h5>Tiêu đề 5</h5> | **Tiêu đề 5** |
| <h6>Tiêu đề 6</h6> | **Tiêu đề 6** |

#### Ví dụ:

<!DOCTYPE html>

<html lang="en">

    <head>

    </head>

    <body>

        <h1>Tiêu đề 1</h1>

        <h2>Tiêu đề 2</h2>

        <h3>Tiêu đề 3</h3>

        <h4>Tiêu đề 4</h4>

        <h5>Tiêu đề 5</h5>

        <h6>Tiêu đề 6</h6>

    </body>

</html>

### HTML Paragraphs:

#### Định nghĩa và cách dùng: xác định một đoạn văn (paragraph) Luôn bắt đầu với một dòng mới, trình duyệt sẽ tự thêm một số khoảng trắng trước và sau một đoạn văn

#### Syntax:

<p>Đây là một đoạn văn</p>

#### Ví dụ:

<!DOCTYPE html>

<html lang="en">

    <head>

    </head>

    <body>

        <p>Đây là một đoạn văn</p>

    </body>

</html>

### HTML Styles:

#### Định nghĩa:

##### Là một trong thuộc tính dùng để add styles cho một tag, element. Ví dụ như **màu sắc (color), font, kích cỡ (size)**

##### Thường là dùng trong <head> tag để chỉnh màu sắc trang hay kích cỡ, màu, …

#### Syntax:

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

#### Ví dụ: (Lưu ý: những thuộc tính trong style sẽ được viết trong file css)

<!DOCTYPE html>

<html lang="en">

    <head>

        <style>

            html {

                font-size: 22px;

            }

            body {

                background-color: lightslategray;

                color: black;

            }

        </style>

    </head>

    <body>…</body>

</html>

### HTML Formatting

#### Định nghĩa:

##### Trong HTML có một số các element để xác định các văn bản có ý nghĩa đặc biệt

##### Một số phần tử thường được dùng: <b>In đậm<\b>, <i>Chữ nghiêng</i>, …

#### Ví dụ:

<html *lang*="en">

<head>

    <meta *charset*="UTF-8">

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

    <title>Document</title>

</head>

<body>

    <b>Văn bản in đậm</b>

    <i>Văn bản in nghiêng</i>

    <sub>Chỉ số dưới (subscript)</sub>

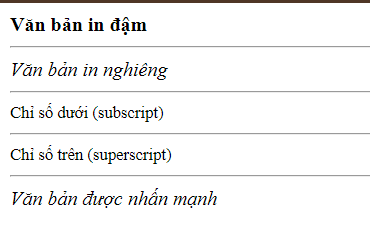
    <sup>Chỉ số trên (superscript)</sup>

    <em>Văn bản được nhấn mạnh</em>

</body>

</html>

#### Output:



### HTML Quotations:

#### Định nghĩa:

##### Dùng để viết các đoạn trích dẫn

#### Các loại thẻ và ví dụ:

##### <blockquote> dùng để xác định một phần được trích dẫn từ một nguồn khác

Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

    <p>Here is a quote from WWF's website:</p>

    <blockquote *cite*="http://www.worldwildlife.org/who/index.html">

        For 60 years, WWF has worked to help people and nature thrive. As the world's leading conservation organization,

        WWF works in nearly 100 countries. At every level, we collaborate with people around the world to develop and

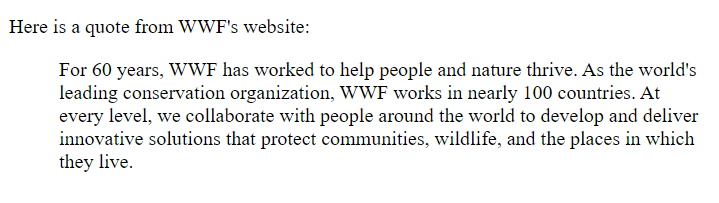
        deliver innovative solutions that protect communities, wildlife, and the places in which they live.

    </blockquote>

</body>

</html>

##### **Output**:



##### < q> Dùng với các trích dẫn ngắn hoặc các trích dẫn có dấu ngoặc kép

Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

    <p>Browsers usually insert quotation marks around the q element.</p>

    <p>WWF's goal is to: <q>Build a future where people live in harmony with nature.</q></p>

</body>

</html>

##### <abbr> Dùng để xác định các từ viết tắt

Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

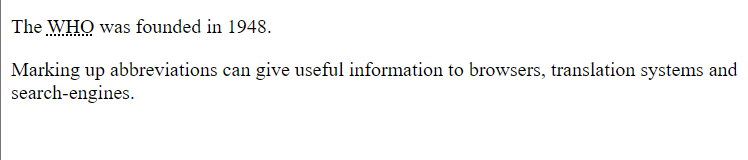
    <p>The <abbr *title*="World Health Organization">WHO</abbr> was founded in 1948.</p>

    <p>Marking up abbreviations can give useful information to browsers, translation systems and search-engines.</p>

</body>

</html>

##### Output: Khi trỏ vào từ “WHO” thì sẽ hiển thị “World Health Organization”.



##### <address> dùng để xác định thông tin của tác giả: nơi ở, liên lạc, …

Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

    <p>The HTML address element defines contact information (author/owner) of a document or article.</p>

    <address>

        Written by John Doe.<br>

        Visit us at:<br>

        Example.com<br>

        Box 564, Disneyland<br>

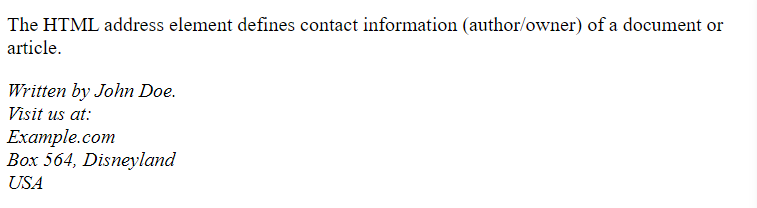
        USA

    </address>

</body>

</html>

Output:



##### <cite> dùng để xác định tiêu đề tác phẩm (sách, thơ, bài hát, …) và văn bản trong tag này thường là in nghiêng.

Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

    <p>The HTML cite element defines the title of a work.</p>

    <p>Browsers usually display cite elements in italic.</p>

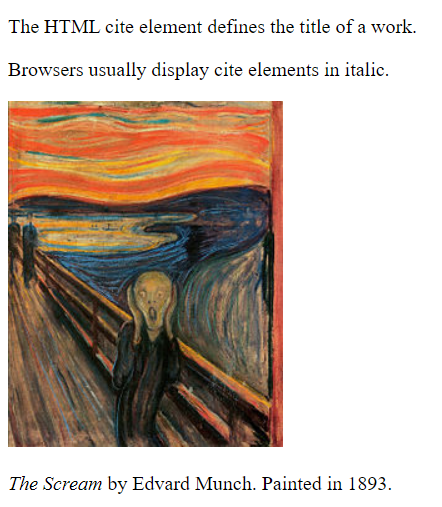
    <img *src*="img\_the\_scream.jpg" *width*="220" *height*="277" *alt*="The Scream">

    <p><cite>The Scream</cite> by Edvard Munch. Painted in 1893.</p>

</body>

</html>

Output: (**Lưu ý: phải có file ảnh thì mới truy cập vào được và hiện ảnh lên không thì chỉ hiện nội dung của thuộc tính *alt* trong <img> tag ! ! !**)



##### <bdo> là viết tắt của Bi-Directional Override dùng để ghi đè văn bản theo hai chiều, nghĩa là viết từ trái qua phải hoặc phải qua trái

Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

    <p>If your browser supports bi-directional override (bdo), the next line will be written from right to left (rtl):

    </p>

    <bdo *dir*="rtl">This line will be written from right to left</bdo>

    <hr>

    <p>If your browser supports bi-directional override (bdo), the next line will be written from right to left (rtl):

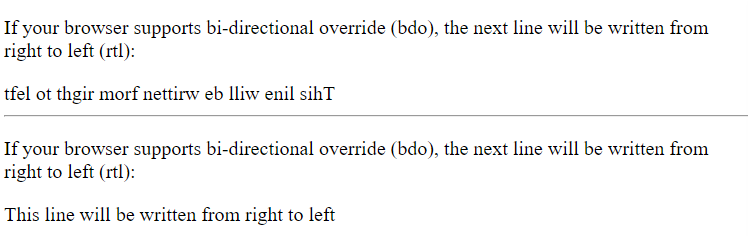
    </p>

    <bdo *dir*="rtl">tfel ot thgir morf nettirw eb lliw enil sihT</bdo>

</body>

</html>

Output:



### HTML Colors

#### Định nghĩa:

##### Màu trong HTML có thể được xác định bằng tên màu (VD: red, blue, …) hoặc dùng mã màu (VD: rgb, rgba, hex, hsl, hsla và **rgb thường được dùng nhiều**)

##### Dùng để chỉnh màu nền (background-color), chỉnh màu văn bản (color), màu viền (border)

#### Syntax và ví dụ:

|  |  |  |
| --- | --- | --- |
| Màu nền | <h1 style="background-color:DodgerBlue;">Hello World</h1>  Hoặc:  <h1 style="background-color: rgb(30, 144, 255);">Hello World</h1>  Hoặc:  <h1 style="background-color:# 1e90ff;">Hello World</h1> | Hello World |
| <h1 style="background-color:Tomato;">Hello World</h1>  Hoặc:  <h1 style="background-color: rgb(255, 99, 71);">Hello World</h1>  Hoặc:  <h1 style="background-color: #ff6347;">Hello World</h1> | Hello World |
| Màu văn bản | <h1 style="color: DodgerBlue;">Hello World</h1>  Hoặc:  <h1 style="color: rgb(30, 144, 255);">Hello World</h1>  Hoặc:  <h1 style="color: #1e90ff;">Hello World</h1> | Hello World |
| <h1 style="color: Tomato;">Hello World</h1>  Hoặc:  <h1 style="color: rgb(255, 99, 71);">Hello World</h1>  Hoặc:  <h1 style="color: #ff6347;">Hello World</h1> | Hello World |
| Màu viền | <h1 style="border: 2px solid DodgerBlue;">Hello World</h1>  Hoặc:  <h1 style="border: 2px solid rgb(30, 144, 255);">Hello World</h1>  Hoặc:  <h1 style="border: 2px solid #1e90ff">Hello World</h1> | Hello World |
| <h1 style="border: 2px solid Tomato;">Hello World</h1>  Hoặc:  <h1 style="border: 2px solid rgb(255, 99, 71);">Hello World</h1>  Hoặc:  <h1 style="border: 2px solid #ff6347;">Hello World</h1> | Hello World |

### HTML CSS

#### Sơ lược CSS:

##### CSS – Cascading Style Sheets: Mục đích CSS ra đời là để giúp tạo ra các trang web đẹp mắt, dễ nhìn và tương tác tốt hơn với người dùng qua việc mô tả cách trình bày các tài liệu được viết bằng HTML hoặc XML.

##### Cho phép định dạng giao diện web page qua các thuộc tính như: màu sắc, phông chữ, bố cục, khoảng cách giữa các phần tử, …

##### Ngoài ra, bằng cách sử dụng CSS, làm cho cách biệt giữa nội dung (HTML) và cách trình bày (CSS) 🡪 giúp dễ sửa chữa,bảo trì web page.

#### Cách sử dụng CSS: được thêm vào HTML doc theo 3 cách sau:

##### **Inline**: Bằng cách dùng style attribute bên trong các phần tử HTML.

##### **Internal**: Bằng cách dùng <style> element trong phần <head>.

##### **External**: Bằng cách dùng <link> element để liên kết đến tệp CSS bên ngoài. (Cách phổ biến).

#### Inline CSS:

##### Được dùng để áp dụng một kiểu duy nhất cho một HTML element

##### Inline CSS sử dụng thuộc tính style của một HTML element

##### Ví dụ:

Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

    <h1 *style*="color:blue;">A Blue Heading</h1>

    <p *style*="color:red;">A red paragraph.</p>

</body>

</html>

Output:



#### Internal CSS:

##### Được sử dụng để xác định style cho một trang HTML (single HTML page).

##### Được xác định trong phần <head> của một trang HTML, bên trong phần tử <style>.

##### Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>

…

    <style>

        body {

            background-color: powderblue;

        }

        h1 {

            color: blue;

        }

        p {

            color: red;

        }

    </style>

</head>

<body>

    <h1>A Blue Heading</h1>

    <p>A red paragraph.</p>

</body>

</html>

Output:



#### External CSS:

##### Được dùng để xác định kiểu cho nhiều trang HTML.

##### Thêm liên kết đến phần <head> trong mỗi HTML page.

##### **HTML Code**: Sử dụng HTML Code để link tới file CSS code

<!DOCTYPE *html*>

<html *lang*="en">

<head>

    <meta *charset*="UTF-8">

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

    <title>Document</title>

    <!-- Link với file style.css -->

    <link *rel*="stylesheet" *href*="style.css">

</head>

<body>

    <h1>A Blue Heading</h1>

    <p>A red paragraph.</p>

</body>

</html>

**CSS Code**: Tinh chỉnh kiểu cách của từng phần để hiển thị ra trang web

body {

    background-color: powderblue; /\* Màu nền chung của trang web \*/

}

h1 {

    color: blue; /\* Màu chữ <h1> tag: xanh \*/

}

p {

    color: red; /\* Màu chữ paragraphs: đỏ \*/

}

Output:



### HTML Links:

#### Định nghĩa:

##### HTML links là các siêu liên kết (hyperlinks)

##### Khi nhấp vào liên kết sẽ chuyển đến page liên kết đó trỏ tới

##### Liên kết có thể là văn bản, hình ảnh, âm thanh, hay bất kỳ phần tử HTML nào khác, …

#### Syntax:

<a *href*="url">link text</a>

##### Thuộc tính bắt buộc phải có: href

##### Phần ***link text:*** phần người dùng nhìn thấy và khi nhấp vào sẽ truy cập đến ***url***

Code:

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

    <p>Go to <a *href*="https://www.google.com">Google</a></p>

</body>

</html>

Output:

|  |
| --- |
| Go to [Google](https://www.google.com/) |

#### Những thuộc tính cần phải biết trong <a> tag:

##### href: Xác định địa chỉ liên kết

##### target: Xác định cách mở liên kết đó:

###### \_self : Mở đường dẫn trong cùng một frame nơi mà bạn nhấp vào địa chỉ (Mặc định)

###### \_blank: Mở trong tab mới hoặc cửa sổ mới

###### \_parent: Mở trong frame cha (web ở trước.)

###### \_top: Mở liên kết trong toàn bộ phần body của window

###### Framename: Mở ở trong tên của iframe.

##### Sử dụng <img> tag (bỏ bên trong <a> tag) để sử dụng hình ảnh như một liên kết

###### Ví dụ:

**HTML Code:**

<!DOCTYPE *html*>

<html *lang*="en">

<head>

    <meta *charset*="UTF-8">

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

    <title>Document</title>

    <!-- Link với file style.css -->

    <link *rel*="stylesheet" *href*="style.css">

</head>

<body>

    <p>

        Go to

<a *href*="https://www.google.com" *target*="\_blank">

            <img *src*="google.png" *alt*="google image">

        </a>

    </p>

</body>

</html>

**CSS Code:**

img {

    vertical-align: middle;

    width: 23px;

}

**Output:**

|  |
| --- |
| Go to |

### HTML Images:

#### Dùng để đưa hình ảnh vào trang web.

#### Về mặt kỹ thuật: ảnh không được chèn vào web page mà được liên kết đến các trang web.

#### Chỉ chứa các thuộc tính, không có thẻ đóng.

#### Syntax:

<img src="url" alt="alternatetext">

#### Thuộc tính src:

##### (Bắt buộc), chỉ đường dẫn (URL) đến ảnh.

##### Hình ảnh phải ở cùng một vị trí với web page, nếu không biểu tượng liên kết sẽ hỏng và ***alternatetext*** hiện lên để thay thế.

#### Thuộc tính alt:

##### Cung cấp văn bản thay thế cho hình ảnh nếu (kết nối chậm, lỗi trong thuộc tính src, …)

##### Ví dụ:

**HTML Code:**

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

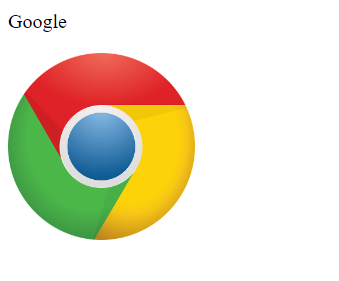
    <p>Google</p>

    <img *src*="google.png" *alt*="The google icon" *width*="25%">

</body>

</html>

**Output:**

****

#### Hình ảnh trên máy chủ khác hoặc website khác:

**HTML Code:**

<!DOCTYPE *html*>

<html *lang*="en">

<head>…</head>

<body>

    <h1>Đường Thái Tông</h1>

    <img *src*="https://upload.wikimedia.org/wikipedia/commons/thumb/b/b6/TangTaizongP.jpg/800px-TangTaizongP.jpg"

*alt*="TangTaizong" *style*="width: 300px;">

</body>

</html>

**Output:**

****

#### Ảnh động (GIFs)

**HTML Code:**

<!DOCTYPE html>

<html lang="en">

<head>…</head>

<body>

    <p><strong>Minion</strong></p>

    <img src="https://miro.medium.com/v2/resize:fit:640/format:webp/1\*AmI9wRbXrfIWGESx6eEiTw.gif" alt="Meme Minion">

</body>

</html>

**Output:**

|  |
| --- |
| **Minion** |

#### Images Floating:

**HTML Code:**

<!DOCTYPE html>

<html lang="en">

<head>…</head>

<body>

    <p>

        <img src="https://www.edigitalagency.com.au/wp-content/uploads/google-logo-icon-PNG-Transparent-Background-letter-G-multiple-colors.png" alt="Google icon" style="float: right; width: 50px; height: 50px;">

            The image will float to the right of the text.

    </p>

    <p>

        <img src="https://www.edigitalagency.com.au/wp-content/uploads/google-logo-icon-PNG-Transparent-Background-letter-G-multiple-colors.png" alt="Google icon" style="float: left; width: 50px; height: 50px;">

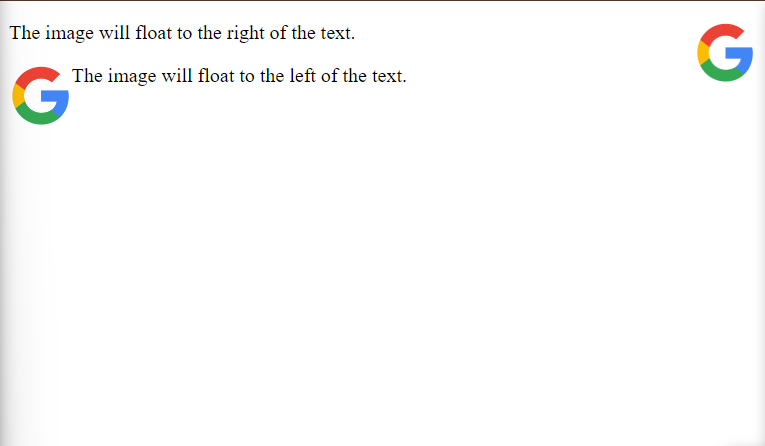
            The image will float to the left of the text.

    </p>

</body>

</html>

**Output:**



#### Image Maps:

##### Là một hình ảnh có các khu vực có thể nhấp vào. Các khu vực được xác định bằng các thẻ <area>.

##### Ví dụ:

**HTML Code:**

<!DOCTYPE html>

<html lang="en">

<head>…</head>

<body>

    <img src="img/caribbean.jpg" alt="Caribbean image" title="Caribbean"

    usemap="#Caribbean">

    <map name="Caribbean">

        <area shape="poly" coords="1,4,0,218,229,224,188,147,166,80,159,36,149,15,145,0" href="https://vi.wikipedia.org/wiki/Bi%E1%BB%83n" alt="beach">

        <area shape="circle" coords="230,28,36" href="https://vi.wikipedia.org/wiki/D%E1%BB%ABa" alt="tree">

        <area shape="poly" coords="267,54,311,0,399,-1,400,29,304,94,313,111,292,127,271,99" href="https://vi.wikipedia.org/wiki/D%E1%BB%ABa" alt="tree">

        <area shape="circle" coords="366,185,47" href="https://vi.wikipedia.org/wiki/D%E1%BB%ABa" alt="tree">

        <area shape="poly" coords="235,223,334,224,306,172,318,164,327,145,343,134,330,118,322,108,312,118,303,130,282,128,269,115,259,94,251,69,221,69,181,47,194,0,155,8,178,75,199,137,214,161,219,172" href="https://vi.wikipedia.org/wiki/C%C3%A1t" alt="sa">

    </map>

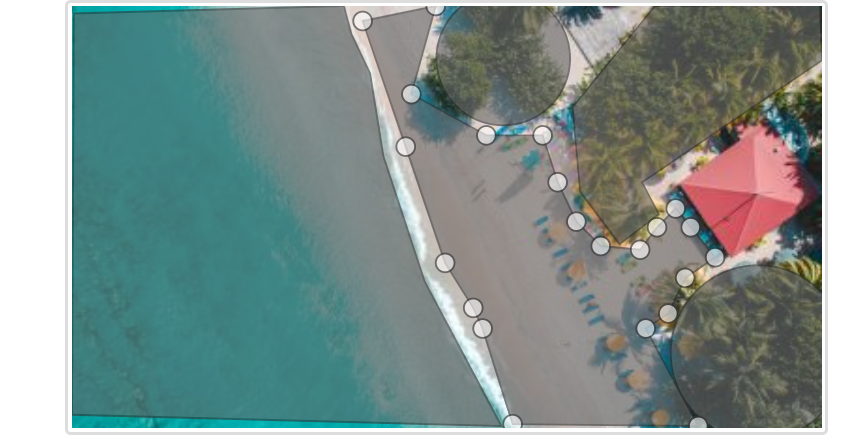
</body>

</html>

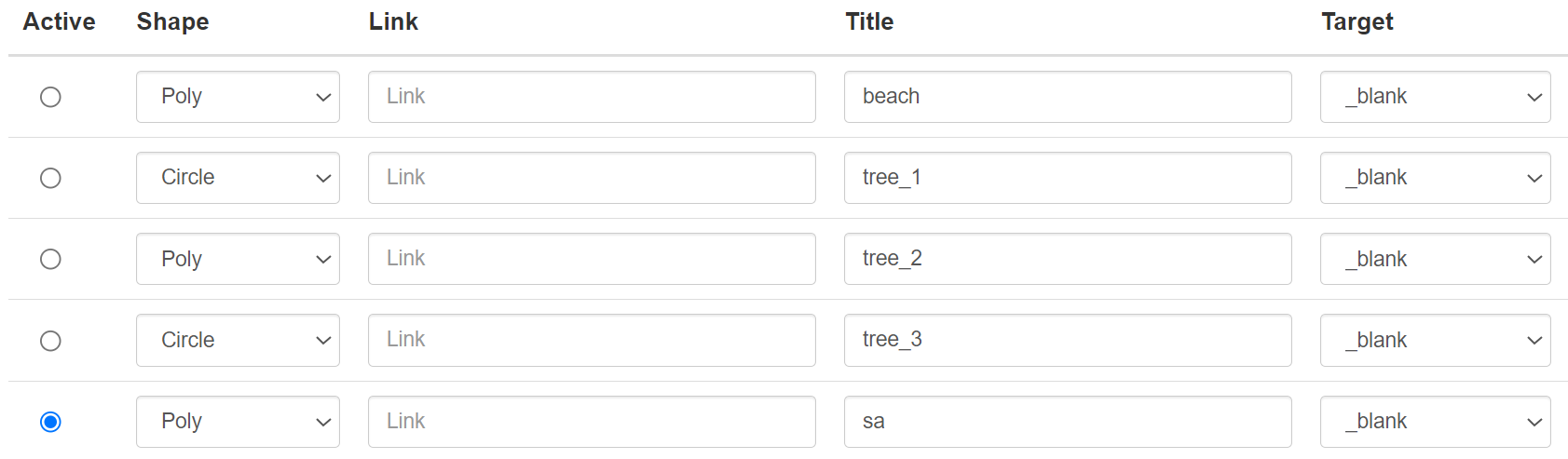
**Hình ảnh được sử dụng:**



**Các vùng được chọn như sau:**









##### **Với HTML Code trên nếu click vào:**

###### Beach: 🡪 đưa tới link <https://vi.wikipedia.org/wiki/Bi%E1%BB%83n>

###### Tree\_1, tree\_2, tree\_3 🡪 <https://vi.wikipedia.org/wiki/D%E1%BB%ABa>

###### Sa 🡪 <https://vi.wikipedia.org/wiki/C%C3%A1t>

### HTML Tables

#### Định nghĩa: Dùng để cho các nhà phát triển web có thể sắp xếp các dữ liệu thành hang và cột

**HTML Code:**

<!DOCTYPE html>

<html>

<style>

table, th, td {

  border:1px solid black;

}

</style>

<body>

<h2>A basic HTML table</h2>

<table style="width:100%">

  <tr>

    <th>Company</th>

    <th>Contact</th>

    <th>Country</th>

  </tr>

  <tr>

    <td>Alfreds Futterkiste</td>

    <td>Maria Anders</td>

    <td>Germany</td>

  </tr>

  <tr>

    <td>Centro comercial Moctezuma</td>

    <td>Francisco Chang</td>

    <td>Mexico</td>

  </tr>

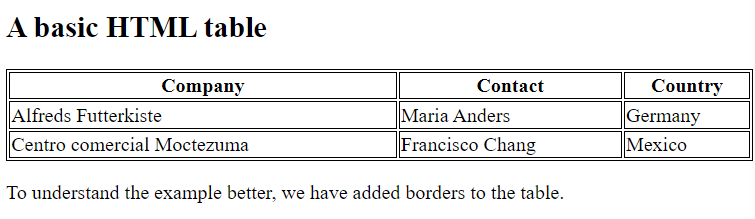
</table>

<p>To understand the example better, we have added borders to the table.</p>

</body>

</html>

**Output:**

****

### HTML Lists

#### Định nghĩa:

##### Cho phép các nhà phát triển web nhóm một tập hợp có các mục liên quan lại với nhau thành một danh sách.

##### Các loại danh sách: Không thứ tự, thứ tự, danh sách mô tả.

#### Ví dụ:

##### Unordered HTML List (Không có thứ tự):

**HTML Code:**

<!DOCTYPE html>

<html>

<body>

<h2>An unordered HTML list</h2>

<ul>

  <li>Coffee</li>

  <li>Tea</li>

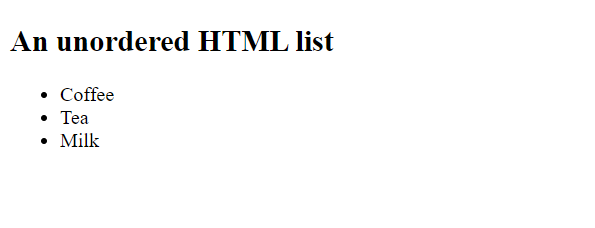
  <li>Milk</li>

</ul>

</body>

</html>

**Output:**



##### Ordered HTML List (Có thứ tự):

**HTML Code:**

<!DOCTYPE html>

<html>

<body>

<h2>An ordered HTML list</h2>

<ol>

  <li>Coffee</li>

  <li>Tea</li>

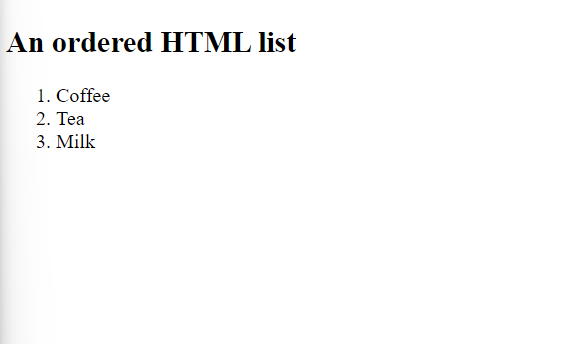
  <li>Milk</li>

</ol>

</body>

</html>

**Output:**



##### HTML Other Lists:

###### HTML Description Lists (Một danh sách có mô tả):

**HTML Code:**

<!DOCTYPE html>

<html>

<body>

<h2>A Description List</h2>

<dl>

  <dt>Coffee</dt>

  <dd>- black hot drink</dd>

  <dt>Milk</dt>

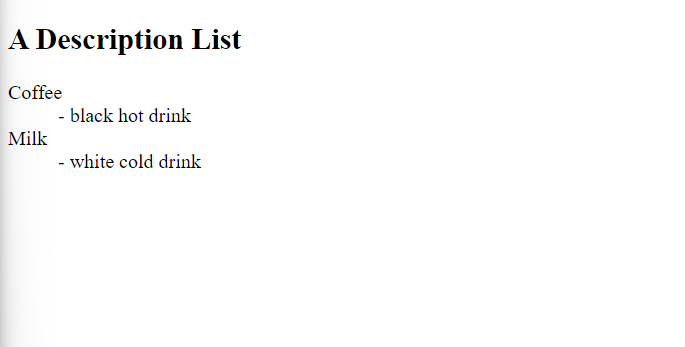
  <dd>- white cold drink</dd>

</dl>

</body>

</html>

**Output:**



Với: <dl> - Xác định danh sách mô tả

<dt> - Xác định thuật ngữ và tên thẻ

<dd> - Mô tả từng thuật ngữ

### HTML Block and Inline

#### Định nghĩa:

##### Mọi phần tử HTML đều có giá trị hiển thị mặc định, tùy vào loại phần tử đó. Và hai giá trị hiển thị phổ biến là block và inline.

##### **Block-level Elements**:

###### luôn bắt đầu trên một dòng mới, và trình duyệt sẽ tự động thêm một số khoản trắng trước và sau phần tử.

###### Luôn chiếm toàn bộ chiều rộng có sẵn (kéo dài sang trái hoặc phải hết mức có thể).

##### **Hai phần tử khối thường dùng:** <p> và <div>

###### <p> - Xác định đoạn văn trong các tài liệu HTML

###### <div> - Xác định một phần trong tài liệu HTML.

###### Ví dụ:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<style>

    table,

    th,

    td {

        border: 1px solid black;

    }

</style>

<body>

    <p *style*="border: 1px solid black">Hello World</p>

    <div *style*="border: 1px solid black">Hello World</div>

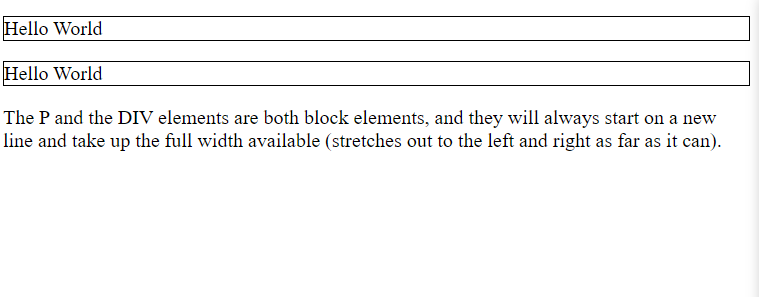
    <p>The P and the DIV elements are both block elements, and they will always start on a new line and take up the full

        width available (stretches out to the left and right as far as it can).</p>

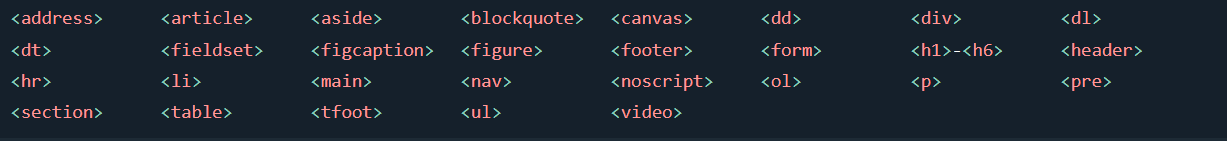
</body>

</html>

**Output:**

****

###### Một số block-level elements trong HTML:



##### **Inline Elemnets**:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<style>

    table,

    th,

    td {

        border: 1px solid black;

    }

</style>

<body>

    <p>This is an inline span <span *style*="border: 1px solid black">Hello World</span> element inside a paragraph.</p>

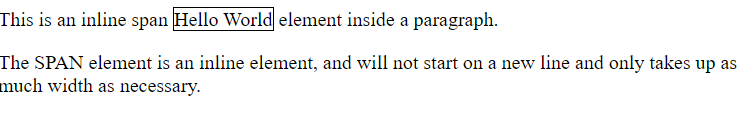
    <p>The SPAN element is an inline element, and will not start on a new line and only takes up as much width as

        necessary.</p>

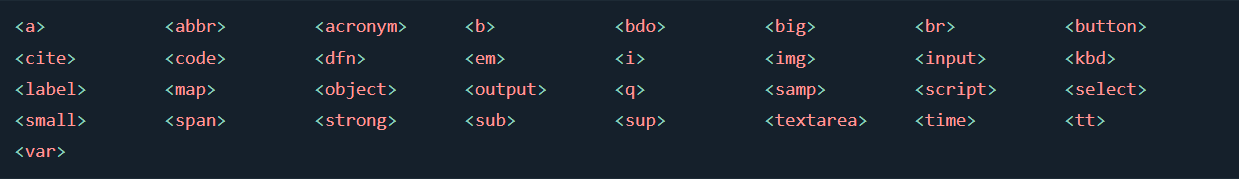
</body>

</html>

**Output:**

****

##### Một số Inline Elements trong HTML:



### HTML Iframes

#### Định nghĩa:

##### Sử dụng để hiển thị một trang web bên trong một trang web.

#### Syntax:

##### Chỉ định khung nội tuyến: khung nội tuyến dùng để nhúng tài liệu khác trong tài liệu HML hiện tại.

<iframe src="url" title="description"></iframe

##### Tip: Cách tốt nhất là luôn bao gồm thuộc tính title cho <iframe>. Điều này được trình đọc màn hình sử dụng để đọc nội dung của iframe.

#### Ví dụ:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h2>HTML Iframes</h2>

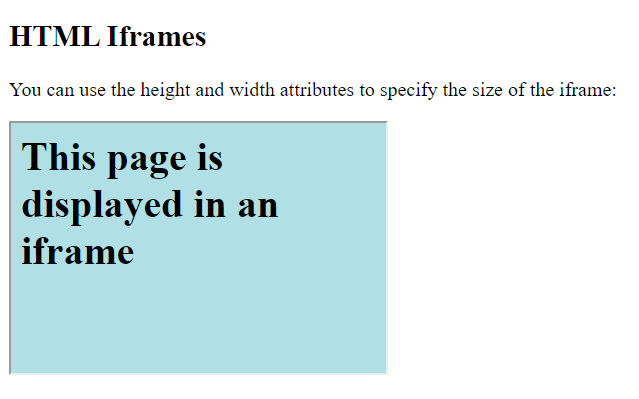
    <p>You can use the height and width attributes to specify the size of the iframe:</p>

    <iframe *src*="demo\_iframe.htm" *height*="200" *width*="300" *title*="Iframe Example"></iframe>

</body>

</html>

**Output:**

****

#### Iframe - Target for a Link

##### Một iframe có thể được sử dụng làm khung đích cho một liên kết.

##### Thuộc tính target của liên kết phải tham chiếu đến thuộc tính name của iframe:

##### Ví dụ:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h2>Iframe - Target for a Link</h2>

    <iframe *src*="demo\_iframe.htm" *name*="iframe\_a" *height*="300px" *width*="100%" *title*="Iframe Example"></iframe>

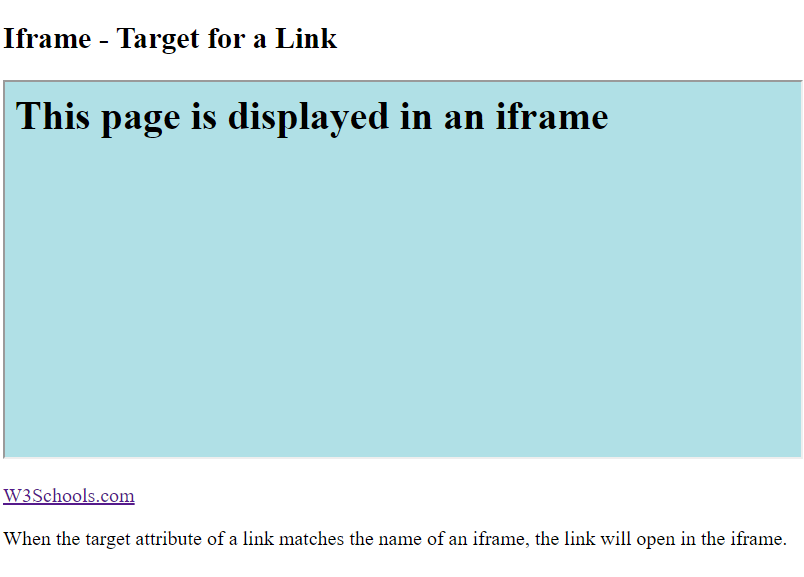
    <p><a *href*="https://www.w3schools.com" *target*="iframe\_a">W3Schools.com</a></p>

    <p>When the target attribute of a link matches the name of an iframe, the link will open in the iframe.</p>

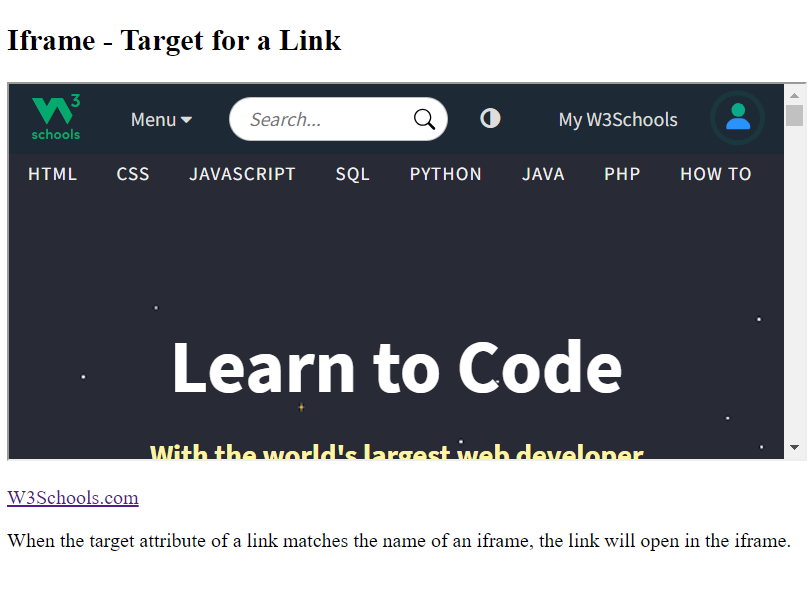
</body>

</html>

**Output:**

****

**Khi click vào link thì sẽ hiển thị trang web W3Schools trong Iframe như dưới đây:**

****

### HTML JavaScript:

#### Định nghĩa:

##### Làm cho các trang HTML trở nên năng động và tương tác hơn.

#### The HTML <script> Tag

##### Sử sụng để xác định tập lệnh phía máy khách (JavaScript)

##### Phần tử ***<script>*** chứa các câu lệnh script hoặc nó trỏ đến tệp script bên ngoài thông qua thuộc tính ***src***.

##### Các cách sử dụng phổ biến của JavaScript là thao tác hình ảnh, xác thực biểu mẫu và thay đổi nội dung động.

##### Để chọn một phần tử HTML, JavaScript thường sử dụng phương thức document.getElementById().

##### **HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h2>Use JavaScript to Change Text</h2>

    <p>This example writes "Hello JavaScript!" into an HTML element with id="demo":</p>

    <p *id*="demo"></p>

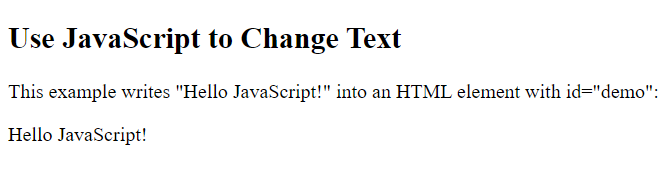
    <script>

        document.getElementById("demo").innerHTML = "Hello JavaScript!";

    </script>

</body>

</html>

**Output:  
**

#### Dưới đây là một số ví dụ về những gì JavaScript có thể làm:

##### **Code 1: Thay đổi nội dung**

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h1>My First JavaScript</h1>

    <p>JavaScript can change the content of an HTML element:</p>

    <button *type*="button" *onclick*="myFunction()">Click Me!</button>

    <p *id*="demo">This is a demonstration.</p>

    <script>

        function myFunction() {

            document.getElementById("demo").innerHTML = "Hello JavaScript!";

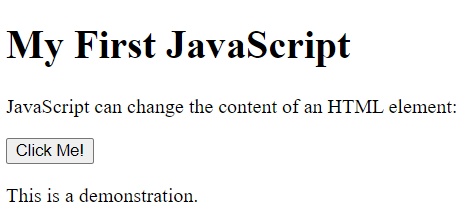
        }

    </script>

</body>

</html>

**Output trước khi bấm nút:**

****

**Output sau khi bấm nút:**

****

##### **Code 2: Thay đổi styles**

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h1>My First JavaScript</h1>

    <p *id*="demo">JavaScript can change the style of an HTML element.</p>

    <script>

        function myFunction() {

            document.getElementById("demo").style.fontSize = "25px";

            document.getElementById("demo").style.color = "red";

            document.getElementById("demo").style.backgroundColor = "yellow";

        }

    </script>

    <button *type*="button" *onclick*="myFunction()">Click Me!</button>

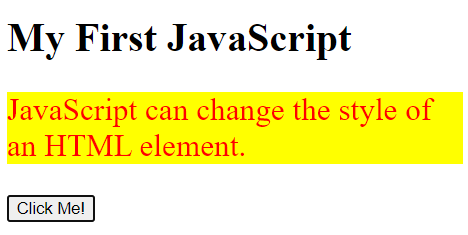
</body>

</html>

**Output trước khi bấm nút:**

****

**Output sau khi bấm nút:**

****

##### **Code 3: Thay đổi thuộc tính (attributes):**

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h1>My First JavaScript</h1>

    <p>Here, a JavaScript changes the value of the src (source) attribute of an image.</p>

    <script>

        function light(*sw*) {

            var pic;

            if (*sw* == 0) {

                pic = "pic\_bulboff.gif"

            } else {

                pic = "pic\_bulbon.gif"

            }

            document.getElementById('myImage').src = pic;

        }

    </script>

    <img *id*="myImage" *src*="pic\_bulboff.gif" *width*="100" *height*="180">

    <p>

        <button *type*="button" *onclick*="light(1)">Light On</button>

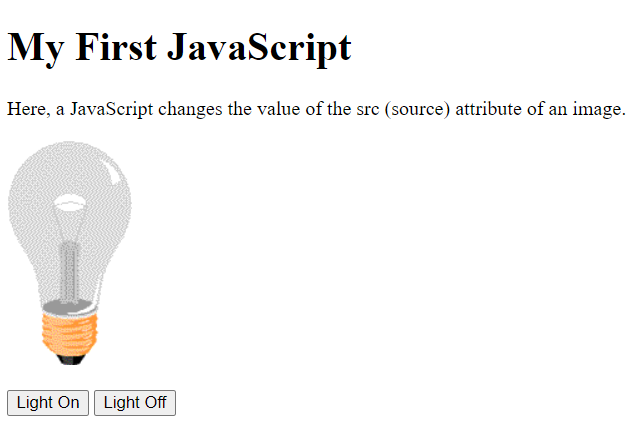
        <button *type*="button" *onclick*="light(0)">Light Off</button>

    </p>

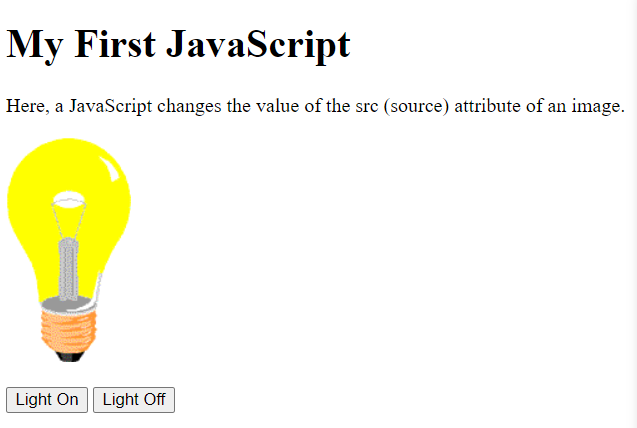
</body>

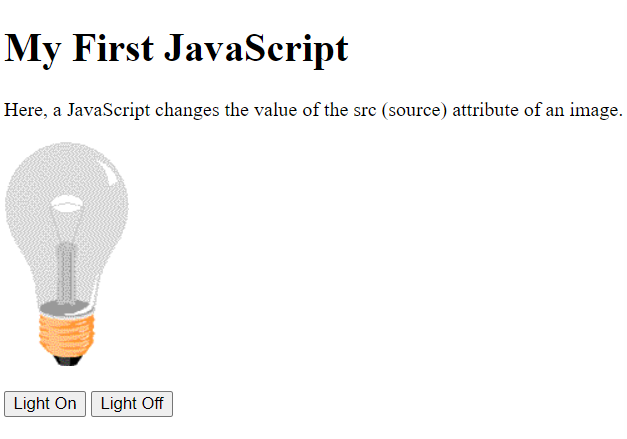
</html>

**Output: ban đầu là đèn tắt:**

****

**Nhấp vào nút light on:**

****

**Nhấp vào nút light off:  
 **

#### The HTML <noscript> Tag

##### xác định nội dung thay thế sẽ được hiển thị cho người dùng vô hiệu hóa ***script*** trong trình duyệt của họ hoặc có trình duyệt không hỗ trợ ***script.***

##### Code:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <p *id*="demo"></p>

    <script>

        document.getElementById("demo").innerHTML = "Hello JavaScript!";

    </script>

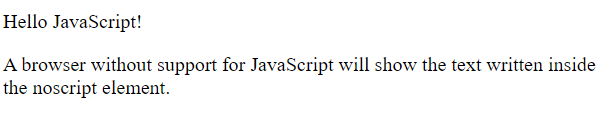
    <noscript>Sorry, your browser does not support JavaScript!</noscript>

    <p>A browser without support for JavaScript will show the text written inside the noscript element.</p>

</body>

</html>

**Output: Vì trình duyệt đang dùng là Google (hoặc Edge, …) thì sẽ không hiển thị đoạn văn trong *noscript.***

****

### HTML Head

#### Định nghĩa:

##### **Là nơi chứa các phần tử:** <title>, <style>, <meta>, <link>, <script>, <base>

##### Là nơi chứa siêu dữ liệu và được đặt giữa thẻ <html> và thẻ <body>

##### Siêu dữ liệu (metadata) HTML là dữ liệu về tài liệu HTML. Siêu dữ liệu không được hiển thị.

##### Siêu dữ liệu thường xác định document title, character set¸ styles, scripts, và thông tin meta khác.

#### The HTML <title> Element

##### xác định tiêu đề của tài liệu. Tiêu đề phải chỉ ở dạng văn bản và được hiển thị trên thanh tiêu đề của trình duyệt hoặc trong tab của trang.

##### là bắt buộc trong tài liệu HTML!

##### Nội dung của tiêu đề trang rất quan trọng để tối ưu hóa công cụ tìm kiếm (SEO)! Tiêu đề trang được thuật toán của công cụ tìm kiếm sử dụng để quyết định thứ tự khi liệt kê các trang trong kết quả tìm kiếm.

##### The <title> element:

###### xác định tiêu đề trên thanh công cụ của trình duyệt

###### cung cấp tiêu đề cho trang khi nó được thêm vào mục yêu thích

###### hiển thị tiêu đề cho trang trong kết quả của công cụ tìm kiếm

**HTML Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Page Title</title>

    <style>

        body {

            background-color: powderblue;

        }

        h1 {

            color: red;

        }

        p {

            color: blue;

        }

    </style>

</head>

<body>

    <h1>This is a Heading</h1>

    <p>This is a paragraph.</p>

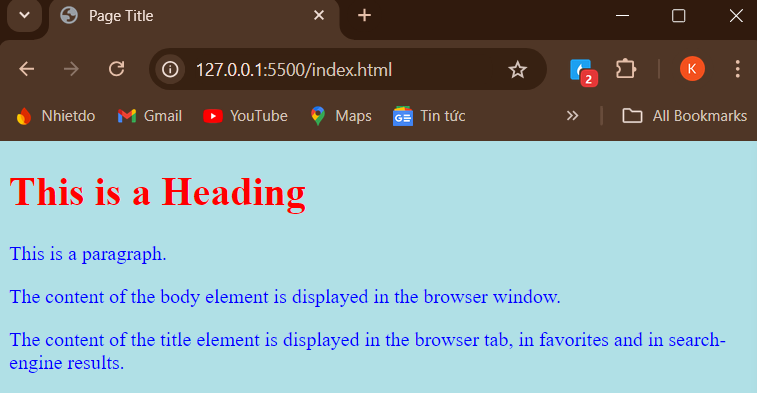
    <p>The content of the body element is displayed in the browser window.</p>

    <p>The content of the title element is displayed in the browser tab, in favorites and in search-engine results.</p>

</body>

</html>

**Output:**

****

#### The HTML <style> Element

##### xác định thông tin style cho một trang HTML

**HTML Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Page Title</title>

    <style>

        body {

            background-color: powderblue;

        }

        h1 {

            color: red;

        }

        p {

            color: blue;

        }

    </style>

</head>

<body>

    <h1>This is a Heading</h1>

    <p>This is a paragraph.</p>

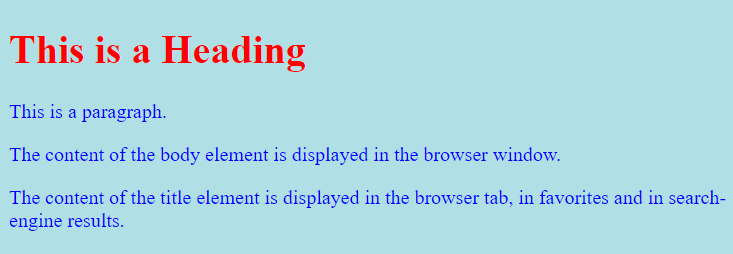
    <p>The content of the body element is displayed in the browser window.</p>

    <p>The content of the title element is displayed in the browser tab, in favorites and in search-engine results.</p>

</body>

</html>

**Output:**

****

#### The HTML <link> Element

##### xác định liên kết giữa tài liệu hiện tại và tài nguyên bên ngoài.

##### thường được sử dụng nhất để liên kết đến các style sheets bên ngoài.

**HTML Code: được liên kết với style.css**

<!DOCTYPE *html*>

<html>

<head>

    <title>Page Title</title>

    <link *rel*="stylesheet" *href*="style.css">

</head>

<body>

    <h1>This is a Heading</h1>

    <p>This is a paragraph.</p>

</body>

</html>

**CSS Code:**

body {

    background-color: #ffff00;

}

p {

    color: #0000ff;

}

**Output:**

****

#### The HTML <meta> Element

##### thường được sử dụng để chỉ định bộ ký tự, mô tả trang, từ khóa, tác giả của tài liệu và cài đặt chế độ xem.

##### sẽ không được hiển thị trên trang nhưng được sử dụng bởi các trình duyệt (cách hiển thị nội dung hoặc tải lại trang), bởi các công cụ tìm kiếm (từ khóa) và các dịch vụ web khác.

**HTML Code:**

<!DOCTYPE *html*>

<html>

<head>

    <!-- Xác định bộ ký tự được sử dụng: -->

    <meta *charset*="UTF-8">

    <!-- Xác định mô tả trang web của bạn: -->

    <meta *name*="description" *content*="Free Web tutorials">

    <!-- Xác định từ khóa cho công cụ tìm kiếm: -->

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

    <!-- Xác định tác giả của một trang -->

    <meta *name*="author" *content*="Khai Thien">

    <!-- Làm mới tài liệu cứ sau 30 giây -->

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

    <!-- Đặt chế độ xem để làm cho trang web của bạn trông đẹp trên tất cả các thiết bị: -->

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

</head>

<body>…</body>

</html>

### HTML Layout

#### Định nghĩa:

Trang web thường hiển thị nội dung theo nhiều cột (như tạp chí hoặc báo)

Xét ví dụ dưới đây:

**HTML Code:**

<!DOCTYPE *html*>

<html *lang*="en">

<head>

   <title>CSS Template</title>

   <meta *charset*="utf-8">

   <meta *name*="viewport" *content*="width=device-width, initial-scale=1">

   <style>

      \* {

         box-sizing: border-box;

      }

      body {

         font-family: Arial, Helvetica, sans-serif;

      }

      /\* Style the header \*/

      header {

         background-color: #666;

         padding: 30px;

         text-align: center;

         font-size: 35px;

         color: white;

      }

      /\* Create two columns/boxes that floats next to each other \*/

      nav {

         float: left;

         width: 30%;

         height: 300px;

         /\* only for demonstration, should be removed \*/

         background: #ccc;

         padding: 20px;

      }

      /\* Style the list inside the menu \*/

      nav ul {

         list-style-type: none;

         padding: 0;

      }

      article {

         float: left;

         padding: 20px;

         width: 70%;

         background-color: #f1f1f1;

         height: 300px;

         /\* only for demonstration, should be removed \*/

      }

      /\* Clear floats after the columns \*/

      section*::after* {

         content: "";

         display: table;

         clear: both;

      }

      /\* Style the footer \*/

      footer {

         background-color: #777;

         padding: 10px;

         text-align: center;

         color: white;

      }

      /\* Responsive layout - makes the two columns/boxes stack on top of each other instead of next to each other, on small screens \*/

      @media (max-width: 600px) {

         nav,

         article {

            width: 100%;

            height: auto;

         }

      }

   </style>

</head>

<body>

   <h2>CSS Layout Float</h2>

   <p>In this example, we have created a header, two columns/boxes and a footer. On smaller screens, the columns will

      stack on top of each other.</p>

   <p>Resize the browser window to see the responsive effect (you will learn more about this in our next chapter - HTML

      Responsive.)</p>

   <header>

      <h2>Cities</h2>

   </header>

   <section>

      <nav>

         <ul>

            <li><a *href*="#">London</a></li>

            <li><a *href*="#">Paris</a></li>

            <li><a *href*="#">Tokyo</a></li>

         </ul>

      </nav>

      <article>

         <h1>London</h1>

         <p>London is the capital city of England. It is the most populous city in the United Kingdom, with a

            metropolitan area of over 13 million inhabitants.</p>

         <p>Standing on the River Thames, London has been a major settlement for two millennia, its history going back

            to its founding by the Romans, who named it Londinium.</p>

      </article>

   </section>

   <footer>

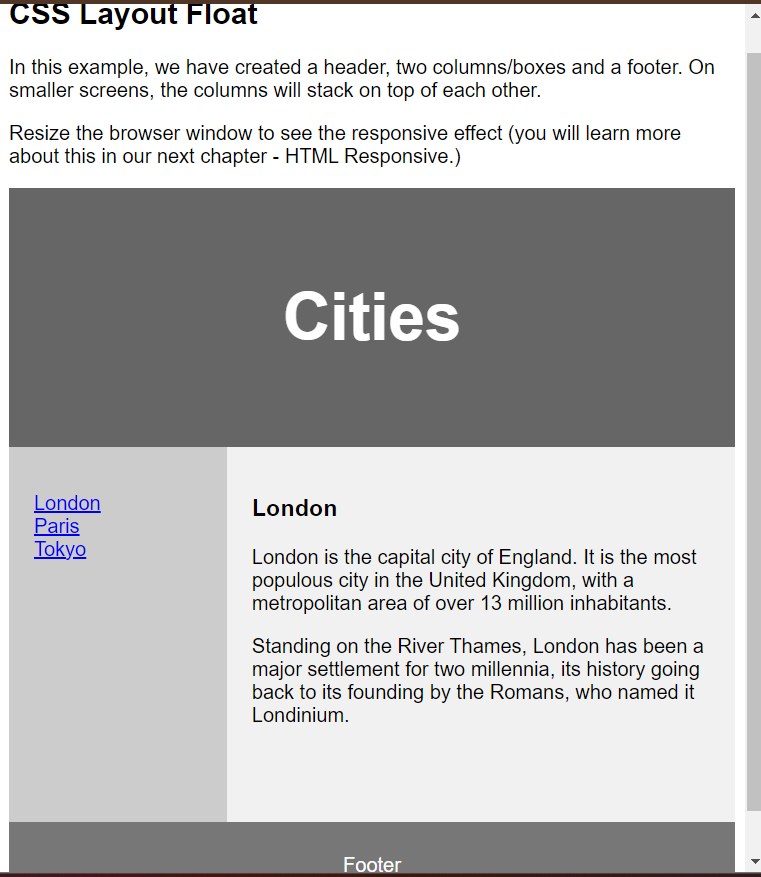
      <p>Footer</p>

   </footer>

</body>

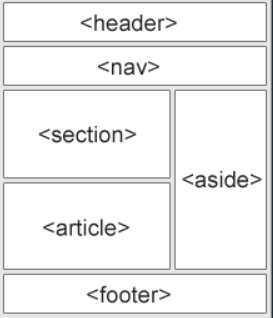
</html>

**Output: Xem phần hiển thị của trang web phía bên dưới**

****

#### HTML Layout Elements:

HTML có một số thành phần ngữ nghĩa xác định các phần khác nhau của trang web:



##### <header> - Xác định tiêu đề cho tài liệu hoặc phần

##### <nav> - Xác định một tập hợp các liên kết điều hướng

##### <section> - Xác định một phần trong tài liệu

##### <article> - Xác định nội dung độc lập, khép kín

##### <aside> - Xác định nội dung ngoài nội dung (như sidebar)

##### <footer> - Xác định chân trang cho tài liệu hoặc section

##### <details> - Xác định các chi tiết bổ sung mà người dùng có thể mở và đóng theo yêu cầu

##### <summary> - Xác định tiêu đề cho phần tử <details>

**HTML Code:**

<!DOCTYPE *html*>

<html *lang*="en">

<head>

    <style>

        body {

            padding: 0;

            margin: 0;

            background-color: #040b26;

            color: #b7b6b6;

        }

*.title* {

            text-align: center;

        }

        header {

            border: none;

            padding: none;

            margin-bottom: 20px;

            height: 50px;

            width: 100%;

            box-shadow: 2px 0px 10px 5px #071445;

            background-color: #07144558;

        }

        nav {

            border: none;

            margin-bottom: 20px;

            padding: none;

            height: 50px;

            width: 100%;

            box-shadow: 2px 0px 10px 5px #071445;

            background-color: #07144558;

        }

        div {

            display: grid;

            grid-template-columns: 1fr 1fr 1fr;

            grid-template-rows: 1fr 1fr;

            grid-template-areas:

                "section section aside"

                "article article aside";

        }

        section {

            grid-area: section;

            border: none;

            margin-bottom: 20px;

            padding: none;

            box-shadow: 2px 0px 10px 5px #071445;

            background-color: #07144558;

        }

        article {

            grid-area: article;

            border: none;

            padding: none;

            box-shadow: 2px 0px 10px 5px #071445;

            background-color: #07144558;

        }

        aside {

            grid-area: aside;

            border: none;

            margin-left: 20px;

            padding: none;

            box-shadow: 2px 0px 10px 5px #071445;

            background-color: #07144558;

        }

        footer {

            border: 0;

            margin-top: 20px;

            padding: 0;

            height: 50px;

            width: 100%;

            box-shadow: 2px 0px 10px 5px #071445;

            background-color: #07144558;

        }

    </style>

</head>

<body>

    <header>

        <h1 *class*="title" *id*="#home">The header</h1>

    </header>

    <nav *class*="title">

        <h1>The nav Element</h1>

        <ul *style*="list-style-type: none;">

    </nav>

    <div>

        <section>

            <h1 *class*="title">Sections</h1>

        </section>

        <article>

            <h1 *class*="title">Article</h1>

        </article>

        <aside>

            <h1 *class*="title">Aside</h1>

        </aside>

    </div>

    <footer>

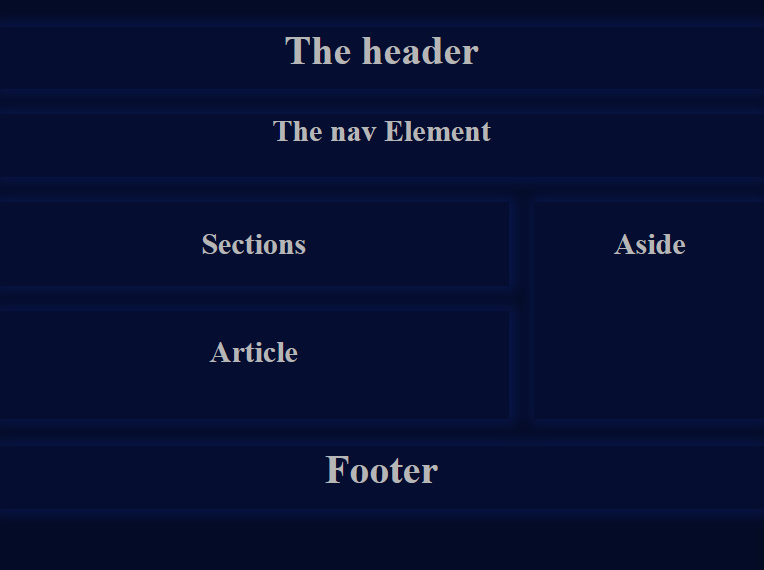
        <h1 *class*="title">Footer</h1>

    </footer>

</body>

</html>

**Output:**

****

#### HTML Layout Techniques (Các kỹ thuật)

Có bốn kỹ thuật khác nhau để tạo bố cục nhiều cột. Mỗi kỹ thuật đều có ưu và nhược điểm:

##### CSS framework

##### CSS float property

##### CSS flexbox

##### CSS grid

**Cụ thể từng kĩ thuật:**

#### CSS Frameworks:

##### Nếu muốn tạo nhanh bố cục của trang thì sử dụng cách này là nhanh nhất.

##### Dùng các thư viện có sẵn hay được lập trình sẵn để thiết kế cho web.

##### Cung cấp cấu trúc cơ bản để mang lại giải pháp thống nhất trong việc xây dựng giao diện web.

#### CSS Float Layout:

##### Sử dụng thuộc tính float trong CSS để tạo bố cục nhiều cột.

##### Ban đầu đươc thiết kế để căn chỉnh hình ảnh, nhưng cũng là một giải pháp được sử dụng để bố trí phần tử của trang.

**HTML Code:**

<!DOCTYPE *html*>

<html *lang*="en">

<head>

    <title>CSS Template</title>

    <meta *charset*="utf-8">

    <meta *name*="viewport" *content*="width=device-width, initial-scale=1">

    <style>

        \* {

            box-sizing: border-box;

        }

        body {

            font-family: Arial, Helvetica, sans-serif;

        }

        /\* Style the header \*/

        header {

            background-color: #666;

            padding: 30px;

            text-align: center;

            font-size: 35px;

            color: white;

        }

        /\* Create two columns/boxes that floats next to each other \*/

        nav {

            float: left;

            width: 30%;

            height: 300px;

            /\* only for demonstration, should be removed \*/

            background: #ccc;

            padding: 20px;

        }

        /\* Style the list inside the menu \*/

        nav ul {

            list-style-type: none;

            padding: 0;

        }

        article {

            float: left;

            padding: 20px;

            width: 70%;

            background-color: #f1f1f1;

            height: 300px;

            /\* only for demonstration, should be removed \*/

        }

        /\* Clear floats after the columns \*/

        section*::after* {

            content: "";

            display: table;

            clear: both;

        }

        /\* Style the footer \*/

        footer {

            background-color: #777;

            padding: 10px;

            text-align: center;

            color: white;

        }

        /\* Responsive layout - makes the two columns/boxes stack on top of each other instead of next to each other, on small screens \*/

        @media (max-width: 600px) {

            nav,

            article {

                width: 100%;

                height: auto;

            }

        }

    </style>

</head>

<body>

    <h2>CSS Layout Float</h2>

    <p>In this example, we have created a header, two columns/boxes and a footer. On smaller screens, the columns will

        stack on top of each other.</p>

    <p>Resize the browser window to see the responsive effect (you will learn more about this in our next chapter - HTML

        Responsive.)</p>

    <header>

        <h2>Cities</h2>

    </header>

    <section>

        <nav>

            <ul>

                <li><a *href*="#">London</a></li>

                <li><a *href*="#">Paris</a></li>

                <li><a *href*="#">Tokyo</a></li>

            </ul>

        </nav>

        <article>

            <h1>London</h1>

            <p>London is the capital city of England. It is the most populous city in the United Kingdom, with a

                metropolitan area of over 13 million inhabitants.</p>

            <p>Standing on the River Thames, London has been a major settlement for two millennia, its history going

                back to its founding by the Romans, who named it Londinium.</p>

        </article>

    </section>

    <footer>

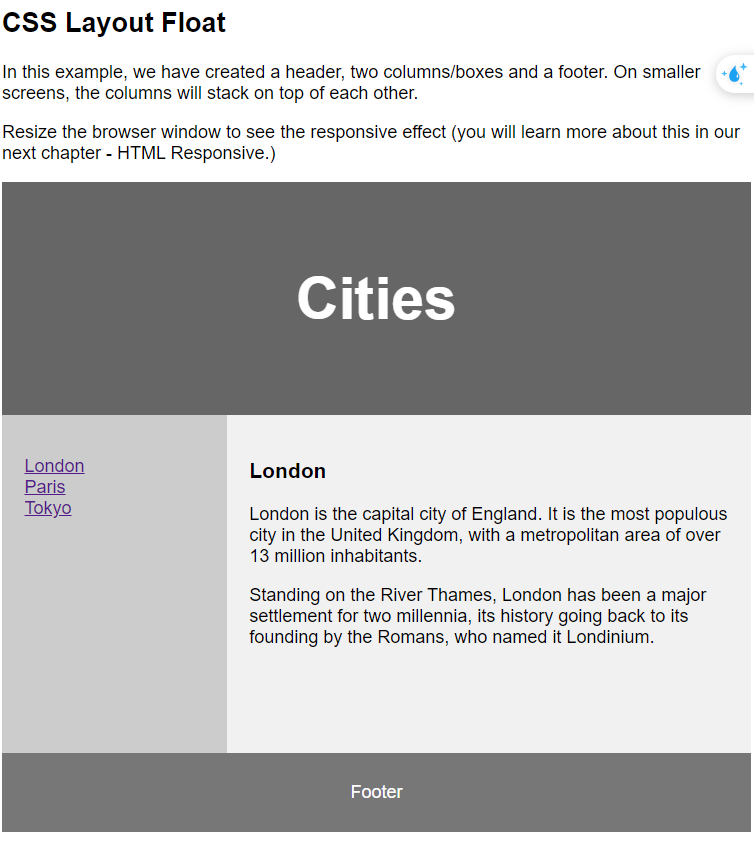
        <p>Footer</p>

    </footer>

</body>

</html>

**Output:**

****

##### Ưu điểm của kĩ thuật này là dễ dàng đọc hiểu, đặc biệt là cho các bố cục đơn giản và được hỗ trợ trên tất cả trình duyệt

##### Ngược lại thì nó là một kĩ thuật rất khó để lập trình viên kiểm soát khi bố cục trở nên phức tạp, ngoài ra không có khả năng tự động điều chỉnh chiều cao của các cột.

#### CSS Flexbox Layout

##### Là một công cụ mạnh mẽ của CSS, được thiết kế để sắp xếp và căn chỉnh các phần tử. Linh hoạt và dễ sử dụng để tạo bố cục một chiều (một hàng hoặc một cột)

**HTML Code:**

<!DOCTYPE *html*>

<html *lang*="en">

<head>

    <title>CSS Template</title>

    <meta *charset*="utf-8">

    <meta *name*="viewport" *content*="width=device-width, initial-scale=1">

    <style>

        \* {

            box-sizing: border-box;

        }

        body {

            font-family: Arial, Helvetica, sans-serif;

        }

        /\* Style the header \*/

        header {

            background-color: #666;

            padding: 30px;

            text-align: center;

            font-size: 35px;

            color: white;

        }

        /\* Container for flexboxes \*/

        section {

            display: -webkit-flex;

            display: flex;

        }

        /\* Style the navigation menu \*/

        nav {

*-webkit-flex*: 1;

*-ms-flex*: 1;

            flex: 1;

            background: #ccc;

            padding: 20px;

        }

        /\* Style the list inside the menu \*/

        nav ul {

            list-style-type: none;

            padding: 0;

        }

        /\* Style the content \*/

        article {

*-webkit-flex*: 3;

*-ms-flex*: 3;

            flex: 3;

            background-color: #f1f1f1;

            padding: 10px;

        }

        /\* Style the footer \*/

        footer {

            background-color: #777;

            padding: 10px;

            text-align: center;

            color: white;

        }

        /\* Responsive layout - makes the menu and the content (inside the section) sit on top of each other instead of next to each other \*/

        @media (max-width: 600px) {

            section {

*-webkit-flex-direction*: column;

                flex-direction: column;

            }

        }

    </style>

</head>

<body>

    <h2>CSS Layout Flexbox</h2>

    <p>In this example, we have created a header, two columns/boxes and a footer. On smaller screens, the columns will

        stack on top of each other.</p>

    <p>Resize the browser window to see the responsive effect.</p>

    <header>

        <h2>Cities</h2>

    </header>

    <section>

        <nav>

            <ul>

                <li><a *href*="#">London</a></li>

                <li><a *href*="#">Paris</a></li>

                <li><a *href*="#">Tokyo</a></li>

            </ul>

        </nav>

        <article>

            <h1>London</h1>

            <p>London is the capital city of England. It is the most populous city in the United Kingdom, with a

                metropolitan area of over 13 million inhabitants.</p>

            <p>Standing on the River Thames, London has been a major settlement for two millennia, its history going

                back to its founding by the Romans, who named it Londinium.</p>

        </article>

    </section>

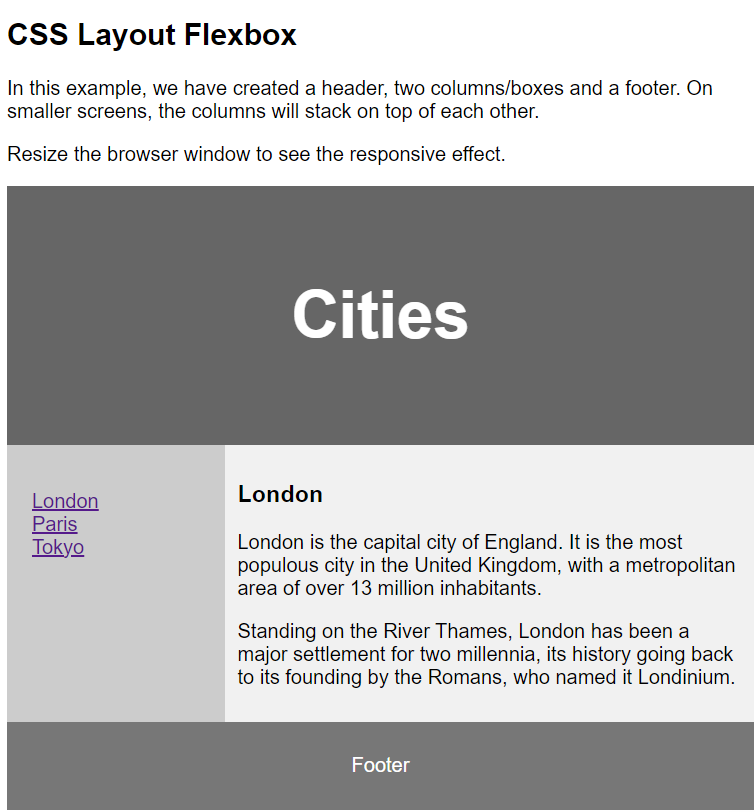
    <footer>

        <p>Footer</p>

    </footer>

</body>

</html>

**Output:  
 **

##### Ưu điểm:

###### Linh hoạt, dễ căn chỉnh các phần tử

###### Giải quyết được nhiều vấn đề của float như căn giữa, chia đều không gian

###### Dễ tạo bố cục đáp ứng (responsive)

##### Nhược điểm:

###### Tương thích tốt với bố cục một chiều, nhưng không phải là lựa chọn tốt nhất cho bố cục lưới phức tạp.

#### CSS Grid Layout:

##### là hệ thống lưới mạnh mẽ nhất của CSS, được thiết kế để tạo ra các bố cục hai chiều (cả hàng và cột). Nó cho phép bạn dễ dàng sắp xếp các phần tử vào một cấu trúc lưới.

##### Ưu điểm:

###### Dễ dàng tạo bố cục lưới phức tạp

###### Có thể kiểm soát tốt khoảng cách, vị trí và căn chỉnh các phần tử

###### Phù hợp cho các thiết kế đáp ứng

##### Nhược điểm:

###### Cần thời gian để hiểu các khái niệm như grid-template-areas, grid-template-columns, grid-template-rows.

###### Không được hỗ trợ đầy đủ trên một số trình duyệt cũ.

### HTML Symbols

#### Định nghĩa:

##### là các ký hiệu hoặc ký tự đặc biệt có thể được sử dụng trong trang HTML

##### bao gồm các ký hiệu toán học, ký tự đặc biệt, ký hiệu tiền tệ, mũi tên, ký hiệu bản quyền, …

##### Để hiển thị các ký hiệu này trong HTML, bạn có thể sử dụng các **entity** (thực thể) hoặc **mã Unicode**.

#### HTML Symbol Entities

##### là một chuỗi bắt đầu bằng dấu ***“&”*** và kết thúc bằng dấu ***“;”***

##### được sử dụng để hiển thị các ký tự đặc biệt mà bạn không thể gõ trực tiếp trên bàn phím hoặc cần tránh xung đột cú pháp trong HTML.

##### Ví dụ:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <p>I will display &euro;</p>

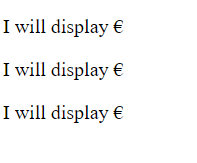
    <p>I will display &#8364;</p>

    <p>I will display &#x20AC;</p>

</body>

</html>

**Output:**

****

### HTML Emojis

#### Định nghĩa:

##### là các ký tự trong bộ ký tự UTF-8: (😄 😍 💗)

#### Ví dụ:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<head>

    <!-- Để hiển thị chính xác một trang HTML, trình duyệt web phải biết bộ ký tự được sử dụng trong trang. -->

    <!-- dùng meta tag -->

    <meta *charset*="UTF-8">

</head>

<body>

    <h1>HTML Emoji Example</h1>

    <!-- emoji -->

    <h2>

        &#128507; &#128508; &#128509; &#128510; &#128511; &#128512; &#128513; &#128514; &#128515; &#128516; &#128517;

    </h2>

    <!-- charset Attribute -->

    <!-- meta tag: A = &#65; | B = &#66; | C = &#67; -->

    <h2>

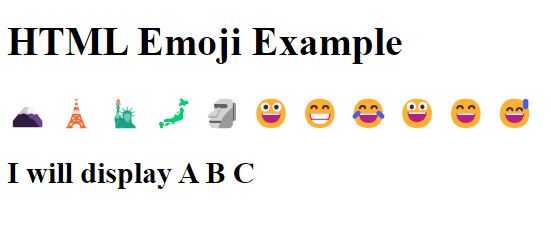
        I will display &#65; &#66; &#67;

    </h2>

</body>

</html>

**Output:**

****

### HTML URL Encode

#### Định nghĩa:

##### là quá trình chuyển đổi các ký tự đặc biệt trong URL thành một định dạng an toàn để truyền qua mạng Internet

#### Cách hoạt động:

##### Khi mã hóa URL, các ký tự không an toàn sẽ được thay thế bằng một chuỗi % theo sau là hai số thập lục phân đại diện cho mã ASCII của ký tự đó.

##### Ví dụ:

###### https://example.com/my page.html sẽ trở thành https://example.com/my%20page.html

###### https://example.com?name=John&Doe sẽ trở thành https://example.com?name=John%26Doe

### HTM Forms

Thu thập ***input*** của người dùng. ***input*** của người dùng thường được gửi đến máy chủ để xử lý.

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h2>HTML Forms</h2>

    <form *action*="https://httpbin.org/get">

        <label *for*="fname">First name:</label><br>

        <input *type*="text" *id*="fname" *name*="fname" *value*="John"><br>

        <label *for*="lname">Last name:</label><br>

        <input *type*="text" *id*="lname" *name*="lname" *value*="Doe"><br><br>

        <input *type*="submit" *value*="Submit">

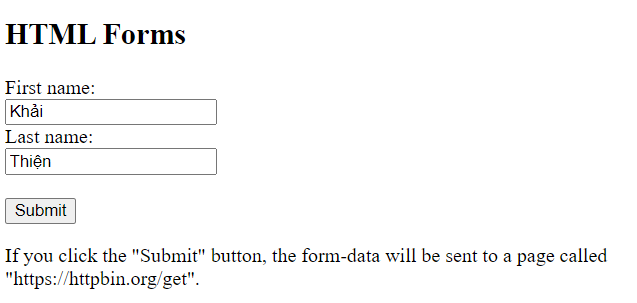
    </form>

    <p>If you click the "Submit" button, the form-data will be sent to a page called "/action\_page.php".</p>

</body>

</html>

**Output:**

****

**Khi bấm vào nút Submit thì sẽ truy cập vào đường link này:**

[**https://httpbin.org/get**](https://httpbin.org/get)

**Nhưng đường dẫn url sẽ thay đổi như sau:**

[**https://httpbin.org/get?fname=Kh%E1%BA%A3i&lname=Thi%E1%BB%87n**](https://httpbin.org/get?fname=Kh%E1%BA%A3i&lname=Thi%E1%BB%87n)

#### HTM Form Attributes (Các thuộc tính của <form> tag)

##### Thuộc tính ***action***:

###### Xác định hành động khi ***form*** được gửi

###### Thông thường, ***form data*** được gửi đến file trên server khi người dùng click vào nút **Submit.**

##### Thuộc tính ***target:***

###### Chỉ định nơi hiển thị response (phản hồi) mà được gửi đến sau khi đã **submitting *form.***

###### Thuộc tính có thể chứa các giá trị: ***\_blank, \_self (mặc định), \_parent, \_top, framename.***

##### Thuộc tính ***method:***

###### Chỉ định phương thức HTTP được sử dụng khi gửi đi **form data.**

###### **form-data** có thể được gửi như các biến URL (với ***method=”get”*** *đây là mặc định*) hoặc như **HTTP post transaction** (với ***method=”post”***)

##### Thuộc tính ***autocomplete:***

###### Chỉ định một **form** nên tắt hay bật tính năng tự động điền (mặc định là không)

##### Nếu là **on** thì trình duyệt tự động điền giá trị mà người dùng đã nhập trước đó.

#### HTML Form Elements

##### Sử dụng để tạo một biểu mẫu HTML để người dùng nhập vào

##### là nơi chứa các loại phần tử đầu vào khác nhau, chẳng hạn như: ***text fields, checkboxes, radio buttons, submit buttons***, …

##### có thể chứa một hoặc nhiều **form element như: *<input>, <label>, <select>, <button>, <textarea>, <fieldset>, <legend>, <datalist>, <output>, <option>, <optgroup>***

#### The <input> Element

##### Là phần tử được sử dụng phổ biến nhất trong ***form***

##### Một phần tử này có thể được hiển thị theo nhiều cách khác nhau, tùy thuộc vào thuộc tính ***type***.

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h1 *style*="text-align: center;">About me</h1>

    <form *action*="https://httpbin.org/get" *method*="get">

        <fieldset>

            <legend>Personal Info</legend>

            <p>

                <label *for*="firstName">First Name:</label>

                <input *type*="text" *name*="firstName" *id*="firstName" *placeholder*="Khải" *autocomplete*="on" *autofocus*

*required*>

                <!-- required bắt buộc phải nhập dữ liệu

                             autocomplete="on" tự động nhớ dữ liệu đã được điền

                             placeholder để nhắc nhở người dùng nhập dữ liệu -->

            </p>

            <p>

                <label *for*="lastName">Last Name:</label>

                <input *type*="text" *name*="lastName" *id*="lastName" *placeholder*="Thiện" *autocomplete*="on" *required*>

            </p>

            <p>

                <label *for*="password">Password:</label>

                <input *type*="password" *name*="password" *id*="password" *placeholder*="myPassword;" *required*>

            </p>

            <p>

                <label>Sex:</label>

                &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

                &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

                &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

                <input *type*="radio" *name*="sex" *id*="male">

                <label *for*="male">Male</label>&nbsp;&nbsp;&nbsp;

                <input *type*="radio" *name*="sex" *id*="female">

                <label *for*="female">Female</label>&nbsp;&nbsp;&nbsp;

                <input *type*="radio" *name*="sex" *id*="other\_sex">

                <label *for*="other\_sex">other</label>&nbsp;&nbsp;&nbsp;

            </p>

            <p>

                <label *for*="phone">Phone:</label>

                <input *type*="tel" *name*="phone" *id*="phone" *placeholder*="xxx-xxx-xxxx" *pattern*="[0-9]{10}" *required*>

            </p>

            <p>

                <label *for*="decade">Decade:</label>

                <input *type*="number" *name*="decade" *id*="decade" *min*="1950" *max*="2027" *step*="10" *value*="1980">

            </p>

            <p>

                <label *for*="coffee">Favorite Coffee:</label>

                <select *name*="coffee" *id*="coffee">

                    <optgroup *label*="Coffees">

                        <option *value*="regular coffee">Regular Coffee</option>

                        <option *value*="iced coffee" *selected*>Iced Coffee</option>

                    </optgroup>

                    <optgroup *label*="Espresso Drinks">

                        <option *value*="latte">Latte</option>

                        <option *value*="capuchino">Capuchino</option>

                        <option *value*="americano">Americano</option>

                        <option *value*="other">Other</option>

                    </optgroup>

                </select>

            </p>

            <!-- <p>

                <label for="coffee">Favorite Coffee:</label>

                <input type="text" name="coffee" id="coffee" list="coffee-list">

                <datalist id="coffee-list">

                    <option value="coffee"></option>

                    <option value="latte"></option>

                    <option value="espresso"></option>

                    <option value="cortado"></option>

                    <option value="americano"></option>

                    <option value="other"></option>

                </datalist>

            </p> -->

        </fieldset>

        <br>

        <fieldset>

            <legend>Favorite Food</legend>

            <p>

                <input *type*="checkbox" *name*="food" *id*="tacos" *value*="tacos">

                <label *for*="tacos">Tacos</label>

            </p>

            <p>

                <input *type*="checkbox" *name*="food" *id*="pizza" *value*="pizaa">

                <label *for*="pizza">Pizza</label>

            </p>

            <p>

                <input *type*="checkbox" *name*="food" *id*="chiken" *value*="chiken">

                <label *for*="chiken">Chiken</label>

            </p>

            <p>

                <input *type*="checkbox" *name*="food" *id*="other" *value*="other">

                <label *for*="other">Other</label>

            </p>

            <!-- Trùng thuộc tính name thì chỉ có 1 radio được chọn.

             Nếu không có thuộc tính name hoặc tên thuộc tính name khác nhau thì có thể chọn nhiều radio -->

        </fieldset>

        <br>

        <fieldset>

            <legend>Feedback / Note</legend>

            <label *for*="message">Message</label>

            <br>

            <textarea *name*="message" *id*="message" *cols*="30" *rows*="10"

*placeholder*="Write your message here..."></textarea>

        </fieldset>

        <br>

        <button *type*="submit">Submit</button>

        <button *type*="submit" *formaction*="https://httpbin.org/post" *formmethod*="post">Post</button>

        <button *type*="reset">Reset</button>

    </form>

</body>

</html>

**Output: là form cần điền để gửi lên server.**

****

### HTML Graphics

#### Định nghĩa:

##### đề cập đến khả năng tạo và hiển thị đồ họa trên các trang web thông qua HTML và các công nghệ liên quan

##### bao gồm việc sử dụng các thẻ và API để vẽ hình ảnh, đồ họa vector, và đồ họa bitmap trong trình duyệt. Dưới đây là một số thành phần và công nghệ chính liên quan đến đồ họa trong HTML

#### HTML Canvas:

##### Được sử dụng để vẽ đồ họa trên web page một cách nhanh chóng thông qua JavaScript

##### Chỉ chứa đồ họa, nếu muốn vẽ thì thao tác trên JavaScript hoặc trong thẻ script

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <canvas *id*="myCanvas" *width*="200" *height*="100" *style*="border:1px solid #000000;"></canvas>

    <script>

        var canvas = document.getElementById("myCanvas");

        var ctx = canvas.getContext("2d");

        ctx.fillStyle = "#FF0000"; // Màu đỏ

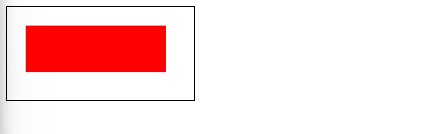
        ctx.fillRect(20, 20, 150, 50); // Vẽ hình chữ nhật

    </script>

</body>

</html>

**Output:**

****

#### HTML SVG: Thẻ <svg>

##### là một cách để tạo đồ họa vector trong HTML

##### cho phép tạo các hình dạng, văn bản, và ảnh động với độ phân giải không bị giảm khi phóng to

##### được định nghĩa bằng XML, vì vậy nó cũng có thể được chỉnh sửa bằng CSS và JavaScript.

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <!-- Vẽ hình tròn -->

    <svg *width*="100" *height*="100">

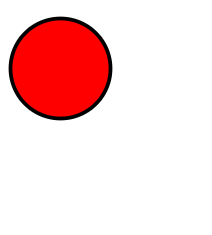
        <circle *cx*="50" *cy*="50" *r*="40" *stroke*="black" *stroke-width*="3" *fill*="red" />

    </svg>

</body>

</html>

**Output:**

****

### HTML Video

#### Định nghĩa:

##### Dùng để hiển thị video trên trang web.

##### Là cách nhúng video vào các trang web bằng thẻ <video> trong HTML.

#### Ví dụ đơn giản:

**Trước tiên tải tệp video này xuống:**



**HTML Code: Viết mã sau:**

<!DOCTYPE *html*>

<html>

<body>

    <video *width*="320" *height*="240" *controls*>

        <source *src*="movie.mp4" *type*="video/mp4">

        <source *src*="movie.ogg" *type*="video/ogg">

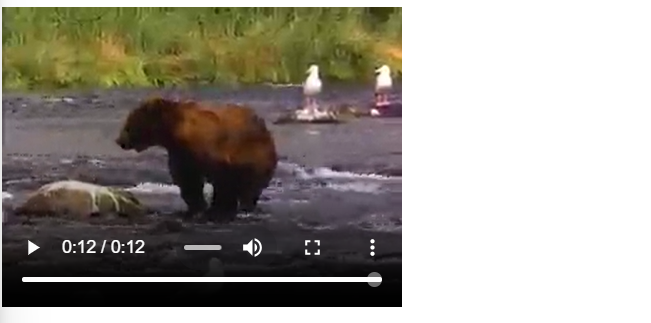
        Your browser does not support the video tag.

    </video>

</body>

</html>

**Output:**

****

#### Cách hoạt động:

##### Thuộc tính ***controls*** thêm các tính năng điều khiển video như: phát, dừng, và âm lượng.

##### Luôn bao gồm các thuộc tính ***width*** và ***height***. Nếu ***width*** và ***height*** không được đặt, trang có thể nhấp nháy trong khi tải video.

##### Phần tử <source> cho phép chỉ định các tệp video thay thế mà trình duyệt có thể chọn. Trình duyệt sẽ sử dụng định dạng được nhận dạng đầu tiên.

##### Văn bản nằm trong thẻ ***<video>*** sẽ chỉ được hiển thị trong các trình duyệt không hỗ trợ phần tử <video>.

#### HTML <video> Autoplay:

##### Sử dụng thuộc tính ***autoplay*** để khi chạy chương chình trên trình duyệt thì video sẽ tư động phát.

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <video *width*="320" *height*="240" *autoplay* *controls*>

        <source *src*="movie.mp4" *type*="video/mp4">

        <source *src*="movie.ogg" *type*="video/ogg">

        Your browser does not support the video tag.

    </video>

</body>

</html>

#### Thêm thuộc tính *muted* sau thuộc tính autoplay để video của bạn bắt đầu phát tự động nhưng không có tiếng.

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <video *width*="320" *height*="240" *autoplay* *muted*>

        <source *src*="movie.mp4" *type*="video/mp4">

        <source *src*="movie.ogg" *type*="video/ogg">

        Your browser does not support the video tag.

    </video>

</body>

</html>

### HTML ID

#### Định nghĩa:

##### Thuộc tính ***id*** được sử dụng để chỉ định một id duy nhất cho một phần tử HTML.

##### Mỗi một phần tử thì chỉ có một ***id*** (giống như mã số sinh viên hay số định danh)

##### Được sử dụng để trỏ đến một khai báo kiểu cụ thể trong một biểu định kiểu. Nó cũng được JavaScript sử dụng để truy cập và thao tác phần tử với id cụ thể.

##### Cú pháp của ***id*** là: viết ký tự băm (#), theo sau là tên ***id***. Sau đó, xác định thuộc tính CSS trong dấu ngoặc nhọn {}.

#### Ví dụ: Trong ví dụ sau, chúng ta có phần tử <h1> trỏ đến tên id "myHeader". Phần tử <h1> này sẽ được tạo kiểu theo định nghĩa kiểu #myHeader trong phần đầu:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<head>

    <style>

*#myHeader* {

            background-color: lightblue;

            color: black;

            padding: 40px;

            text-align: center;

        }

    </style>

</head>

<body>

    <h2>The id Attribute</h2>

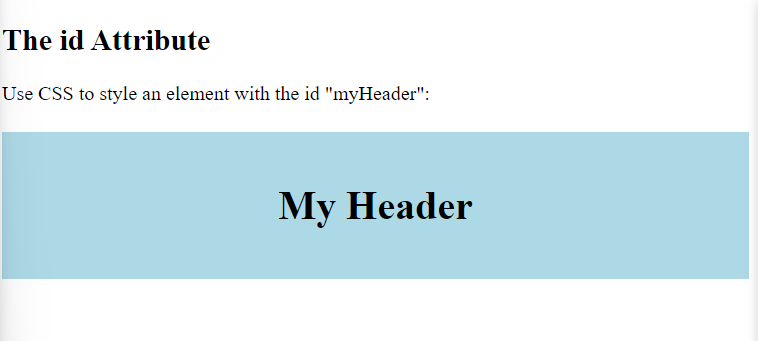
    <p>Use CSS to style an element with the id "myHeader":</p>

    <h1 *id*="myHeader">My Header</h1>

</body>

</html>

**Output:**

****

**Chú ý:** Tên id được phân biệt giữa viết hoa và thường

Tên id phải chứa ít nhất một ký tự, không được bắt đầu bằng số và không được chứa khoảng trắng (dấu cách, tab, v.v.)

#### Sự khác nhau giữa *class* và *id*:

##### Một tên ***class*** có thể được sử dụng bởi nhiều phần tử HTML, trong khi tên ***id*** chỉ được sử dụng bởi một phần tử HTML trong trang:

**HTML Code:**

<!DOCTYPE *html*>

<html>

<head>

    <style>

        /\* Style the element with the id "myHeader" \*/

*#myHeader* {

            background-color: lightblue;

            color: black;

            padding: 40px;

            text-align: center;

        }

        /\* Style all elements with the class name "city" \*/

*.city* {

            background-color: tomato;

            color: white;

            padding: 10px;

        }

    </style>

</head>

<body>

    <h2>Difference Between Class and ID</h2>

    <p>A class name can be used by multiple HTML elements, while an id name must only be used by one HTML element within

        the page:</p>

    <!-- An element with a unique id -->

    <h1 *id*="myHeader">My Cities</h1>

    <!-- Multiple elements with same class -->

    <h2 *class*="city">London</h2>

    <p>London is the capital of England.</p>

    <h2 *class*="city">Paris</h2>

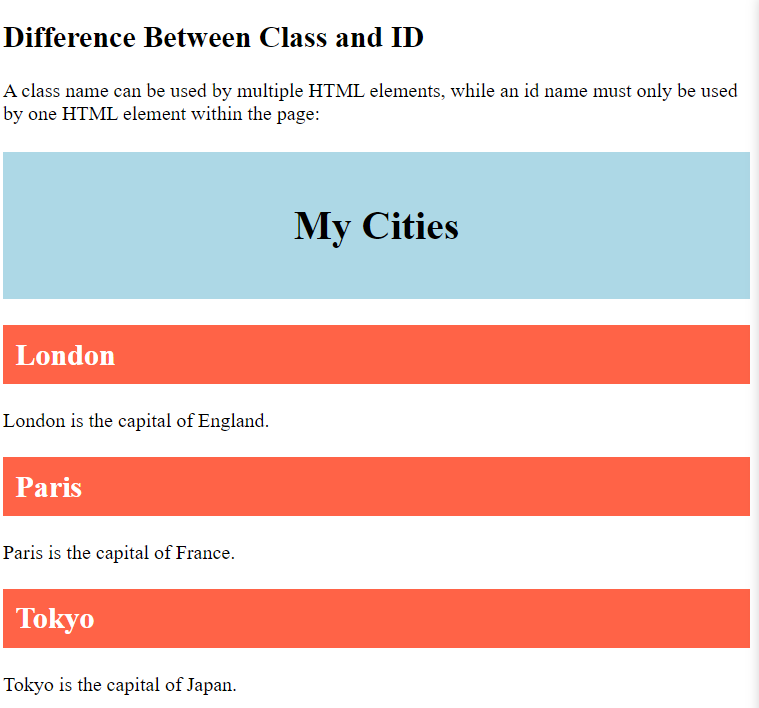
    <p>Paris is the capital of France.</p>

    <h2 *class*="city">Tokyo</h2>

    <p>Tokyo is the capital of Japan.</p>

</body>

</html>

**Output: **

#### HTML Bookmarks có ID and Links:

##### HTML Bookmarks được sử dụng để cho phép người đọc chuyển đến các phần cụ thể của trang web.

##### Để sử dụng ***bookmarks***, trước tiên bạn phải tạo dấu trang rồi thêm liên kết vào dấu trang đó.

##### Sau đó, khi nhấp vào liên kết, trang sẽ cuộn đến vị trí có dấu trang.

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <!-- TODO: Bước 2: thêm liên kết vào dấu trang ("Jump to Chapter 4"), từ trong cùng một trang:

 -->

    <p><a *href*="#C4">Jump to Chapter 4</a></p>

    <p><a *href*="#C10">Jump to Chapter 10</a></p> <!-- Tương tự -->

    <!-- Xong Bước 2 -->

    <h2>Chapter 1</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 2</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 3</h2>

    <p>This chapter explains ba bla bla</p>

    <!-- TODO: Bước 1: Tạo bookmarks (dấu trang) với thuộc tính id -->

    <h2 *id*="C4">Chapter 4</h2>

    <!-- Xong Bước 1 -->

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 5</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 6</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 7</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 8</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 9</h2>

    <p>This chapter explains ba bla bla</p>

    <!-- Tương tư bước 1 -->

    <h2 *id*="C10">Chapter 10</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 11</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 12</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 13</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 14</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 15</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 16</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 17</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 18</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 19</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 20</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 21</h2>

    <p>This chapter explains ba bla bla</p>

    <h2>Chapter 22</h2>

    <p>This chapter explains ba bla bla</p>

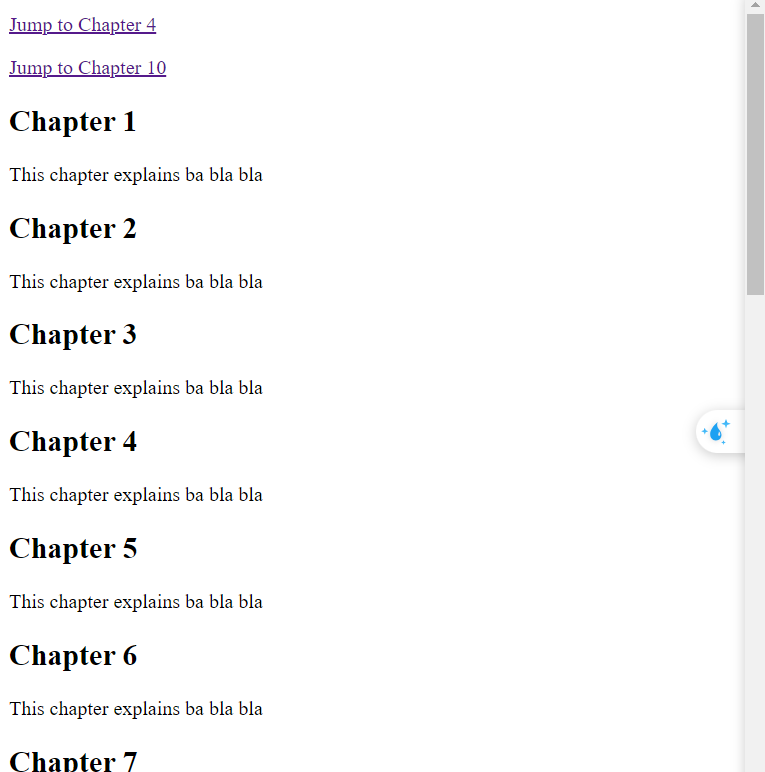
    <h2>Chapter 23</h2>

    <p>This chapter explains ba bla bla</p>

</body>

</html>

**Copy mã code trên và chạy thử:**

****

Khi nhấp vào “Jump to Chapter 4” (hoặc “Jump to Chapter 10”) thì sẽ cuộn xuống phần Chapter 4 (hoặc Chapter 10).

#### Sử dụng thuộc tính id trong JavaScript:

##### Thuộc tính ***id*** cũng có thể được JavaScript sử dụng để thực hiện một số tác vụ cho thành phần cụ thể đó.

##### JavaScript có thể truy cập một phần tử có ***id*** cụ thể bằng phương thức getElementById().

**HTML Code:**

<!DOCTYPE *html*>

<html>

<body>

    <h2>Using The id Attribute in JavaScript</h2>

    <p>JavaScript can access an element with a specified id by using the getElementById() method:</p>

    <h1 *id*="myHeader">Hello World!</h1>

    <button *onclick*="displayResult()">Change text</button>

    <script>

        /\* Viết hàm thay đổi nội dung của thẻ h1 \*/

        function displayResult() {

            /\* Thiết lập biến displayText để lưu nội dung của thẻ h1 \*/

            let displayText = document.getElementById("myHeader").innerHTML;

            /\* Kiểm tra và đặt điều kiện để thay đổi nội dung của thẻ h1 \*/

            if (displayText == "Hello World!") {

                document.getElementById("myHeader").innerHTML = "Have a nice day!";

            } else {

                document.getElementById("myHeader").innerHTML = "Hello World!";

            }

        }

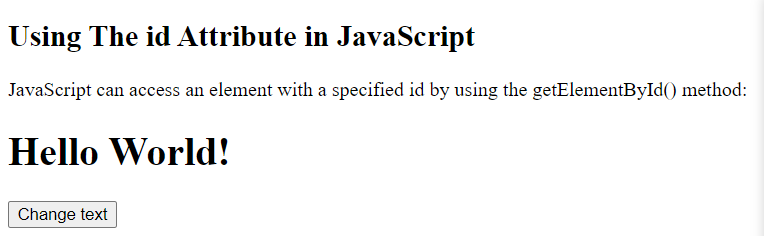
    </script>

</body>

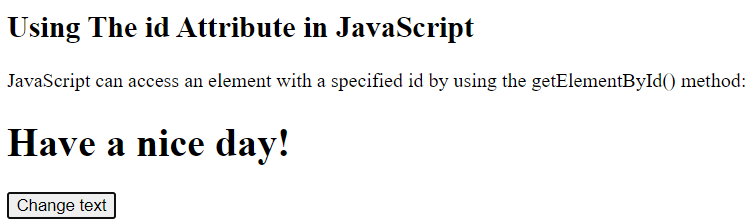
</html>

**Ouptut:**

**Ban đầu: (O1)**

****

**Khi bấm nút “Change text”: (O2)**

****

**Khi bấm nút thì O1 và O2 sẽ hiển thị lên thay phiên nhau cho mỗi lần bấm nút.**

### HTML Plug-ins

#### Định nghĩa:

##### Plug-in là các chương trình máy tính mở rộng chức năng tiêu chuẩn của trình duyệt.

##### được thiết kế để sử dụng cho nhiều mục đích khác nhau:

###### Để chạy các ứng dụng Java

###### Để chạy các điều khiển Microsoft ActiveX

###### Để hiển thị phim Flash

###### Để hiển thị bản đồ

###### Để quét virus

###### Để xác minh id ngân hàng

Warning!

Hầu hết các trình duyệt không còn hỗ trợ Java Applet và Plug-in nữa.

Điều khiển ActiveX không còn được hỗ trợ trong bất kỳ trình duyệt nào.

Hỗ trợ cho Shockwave Flash cũng đã bị tắt trong các trình duyệt hiện đại.

#### The <object> Element

##### Được hỗ trợ bới tất cả các trình duyệt

##### Phần tử ***<object>*** xác định một đối tượng được nhúng trong tài liệu HTML.

##### Nó được thiết kế để nhúng các plug-in (như ứng dụng Java, trình đọc PDF và Trình phát Flash) vào các trang web nhưng cũng có thể được sử dụng để đưa HTML vào HTML:

**HTML Code: đưa HTML vào HTML**

<!-- Source code -->

<!DOCTYPE *html*>

<html>

<body>

    <object *width*="100%" *height*="500px" *data*="snippet.html"></object>

</body>

</html>

Tải snippet.html tại đây: ([snippet.html](https://github.com/HTKFOOLISH/TT_IoT_Architecture_and_Protocol/blob/master/week2/snippet.html))

##### Đưa hình ảnh vào:

**HTML Code:**

<!DOCTYPE *html*>

<html>

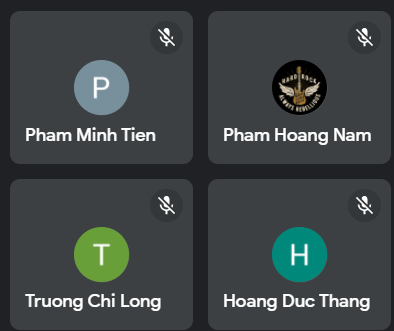
<body>

    <object *data*="google.png"></object>

</body>

</html>

Website (interface, functions) [1] [2]



Hình 1. 1 Danh sách nhóm 1

…………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………

**TÀI LIỆU THAM KHẢO**

(lưu ý: trích dẫn theo chuẩn IEEE và trích dẫn tự động, xem video clip hướng dẫn trên trang dạy học số)

Clip demo: link (youtube)

# References

|  |  |
| --- | --- |
| [1] | P. M. Tien, Ky nang dinh huong noi dung nghien cuu khoa hoc, Tp. HCM: ĐH SPKT Tp. HCM, 2021. |
| [2] | N. C. Đức, Kỹ năng lập trình C lần 2, Tp.HCM: NXB ĐH SPKT Tp.HCM, 2021. |

[Hình 1. 1 Danh sách nhóm 1 1](#_Toc83707706)