



VIETNAM
AUSTRALIA
Vocational College

Slides 1.3: Thao tác với các thẻ Image, List, Table, Form

Mentor: Nguyễn Bá Minh Đạo



Nội dung

1. Thêm các hình ảnh vào trang Web
2. Gom nhóm các phần tử HTML với div
3. Cấp độ Block, Inline và Inline Block
4. Các loại danh sách Lists trong HTML
5. Thuộc tính đặc biệt style của HTML
6. Thuộc tính id, class của HTML
7. Tạo bảng biểu Table, biểu mẫu Form



Thêm các hình ảnh vào trang Web

❑ Các loại hình ảnh trong trang Web:

- ♦ Một số loại **images phổ biến**: ảnh chụp (photos), ảnh vẽ tay (drawings), sơ đồ (diagrams), đồ thị (charts) và nhiều loại đồ họa (graphics) khác,...
- ♦ Trên một trang Web, **images giúp truyền tải thông tin trực quan, chia nhỏ văn bản và đóng góp vào thiết kế và tính thẩm mỹ** của một **trang Web**.
- ♦ Khi kết hợp **images** vào **website**, web designers cần chú ý **định dạng file** và **kích thước images** vì điều đó **quyết định trình duyệt mất bao lâu để hiển thị hình ảnh**





Thêm các hình ảnh vào trang Web

❑ Các loại hình ảnh trong trang Web:

- ◆ Hình ảnh **phải hỗ trợ mục đích** của **trang** hoặc **minh họa** cho **nội dung**:
 - Hình ảnh cũng **có thể trình bày trực quan các sản phẩm** và **dịch vụ** của **công ty**.
 - Khi xác định hình ảnh sẽ sử dụng trong trang web của bạn, **hãy chọn những hình ảnh liên quan trực tiếp đến nội dung**.
 - Hình ảnh không hỗ trợ nội dung có thể **gây mất tập trung** hoặc **khó hiểu**.
- ◆ Trong Web, **chỉ nên dùng** file **images** ở **một số định dạng**: **GIF, PNG, JPG** hoặc **SVG,...**





Thêm các hình ảnh vào trang Web

❑ Các loại hình ảnh trong trang Web:

♦ Ví dụ về website **Let's Move** - website về giáo dục sử dụng để **quảng bá việc ăn uống lành mạnh**. Chú ý website sẽ có logo, photo để **thể hiện một lối sống lành mạnh**.





Thêm các hình ảnh vào trang Web

❑ Các loại định dạng của hình ảnh:

- ◆ Cần nắm rõ đặc tính của từng loại hình ảnh để chọn lựa phù hợp cho trang Web.

Định dạng	Ý nghĩa	Ưu điểm	Nhược điểm	Nên dùng cho
GIF (Graphics Interchange Format)	các định dạng trao đổi đồ họa (phát âm là "jiff")	Kích thước tệp nhỏ; hỗ trợ tính minh bạch và hoạt ảnh	Giới hạn ở 256 màu	Vẽ nét; được thay thế bằng định dạng tệp PNG
PNG (Portable Network Graphics)	các hình ảnh đồ họa mạng di động	Kích thước tệp nhỏ; hỗ trợ tính minh bạch và hơn một triệu màu	Không hỗ trợ hoạt ảnh	Hình ảnh không phải là ảnh kỹ thuật số



Thêm các hình ảnh vào trang Web

❑ Các loại định dạng của hình ảnh:

- ◆ Cần nắm rõ đặc tính của từng loại hình ảnh để chọn lựa phù hợp cho trang Web.

Định dạng	Ý nghĩa	Ưu điểm	Nhược điểm	Nên dùng cho
JPG/JPEG (Joint Photographic Experts Group)	các định dạng nhóm chung chuyên gia nhiếp ảnh (phát âm là “jay-peg”)	Hỗ trợ hơn một triệu màu	Kích thước tệp lớn	Ảnh kỹ thuật số
SVG (Scalable Vector Graphics)	các hình ảnh đồ họa vector có thể mở rộng	Linh hoạt; có thể mở rộng; không cần tệp vì đồ họa được tạo bằng mã	Không được hỗ trợ bởi các trình duyệt cũ hơn, không phải tất cả các trình duyệt hiện đại hỗ trợ nó 100%	Hình dạng, đường kẻ, văn bản và độ dốc



Thêm các hình ảnh vào trang Web


❑ Thẻ hình ảnh và các thuộc tính thẻ hình ảnh:

♦ Thẻ hình ảnh, ****, là **một thẻ HTML trống** được sử dụng **để thêm hình ảnh vào trang web**. Là một thẻ trống, **không có thẻ kết thúc**. Các thuộc tính phổ biến của ****:

Thuộc tính	Chức năng
src	Xác định tên tệp của hình ảnh để hiển thị
alt	Chỉ định văn bản thay thế để hiển thị khi một hình ảnh đang được tải Đặc biệt hữu ích cho trình đọc màn hình, dịch thông tin trên màn hình máy tính vào đầu ra âm thanh. Nên mô tả ngắn gọn mục đích của hình ảnh trong 50 ký tự trở xuống
height/width	Xác định chiều cao/rộng của hình ảnh bằng pixel, giúp cải thiện thời gian tải

=> Nên **luôn dùng thuộc tính alt trong thẻ hình ảnh**, văn bản thay thế cần ngắn gọn vì **trình đọc màn hình sẽ đọc lại văn bản thay thế** và **tăng cường khả năng tiếp cận**.

♦ Ví dụ thêm thẻ img để hiển thị logo website VUS Fitness Club (chú ý đường dẫn):

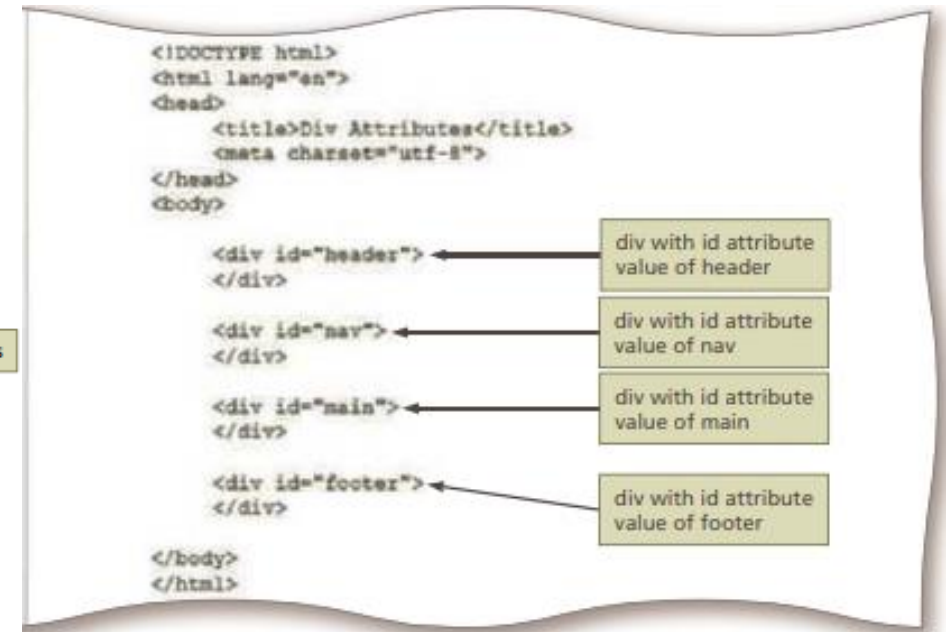
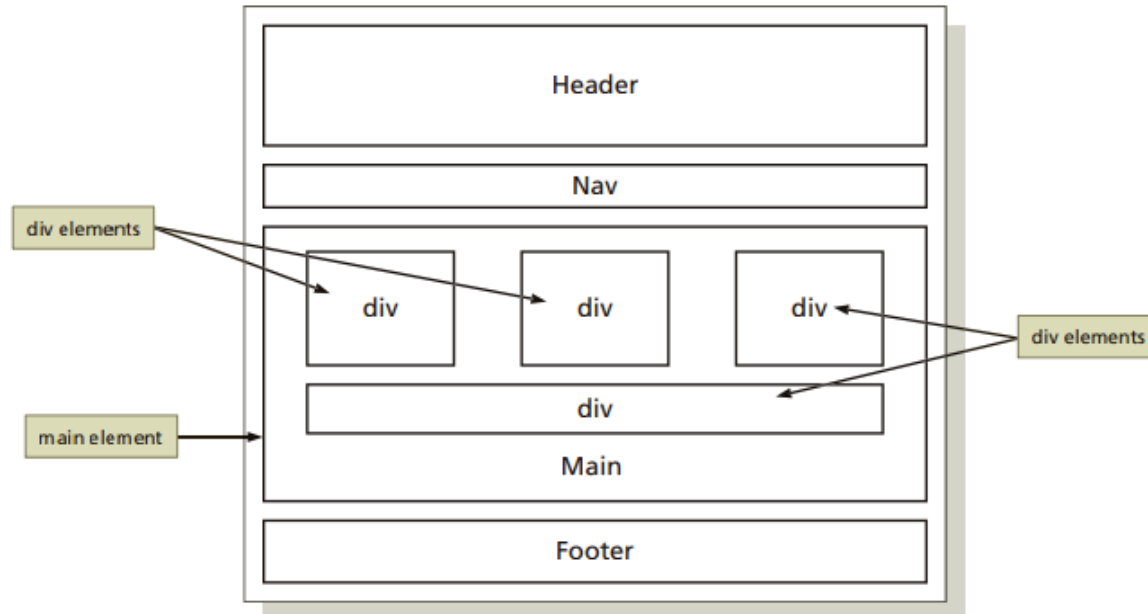
```
> img.html >  img  
1 
```




Gom nhóm các phần tử HTML với div

❑ Thẻ <div> trong HTML:

- ♦ Thẻ **div** được **sử dụng để xác định một khu vực** hoặc **một bộ phận** trong **trang web**.
- ♦ Bạn **chèn các phần tử div bằng các thẻ <div>** nội dung thẻ div **</div>**.
- ♦ Các **semantic HTML5 thường thay thế** các phần tử **div** cho **các khu vực: header, navigation, main, footer** vì các phần tử HTML5 mới, phản ánh tốt hơn mục đích của thẻ.

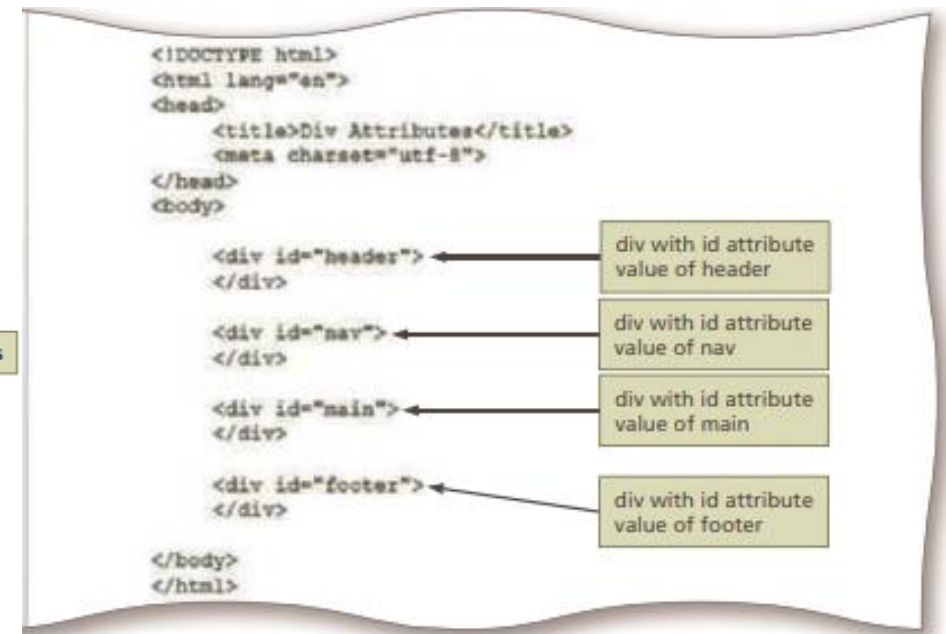
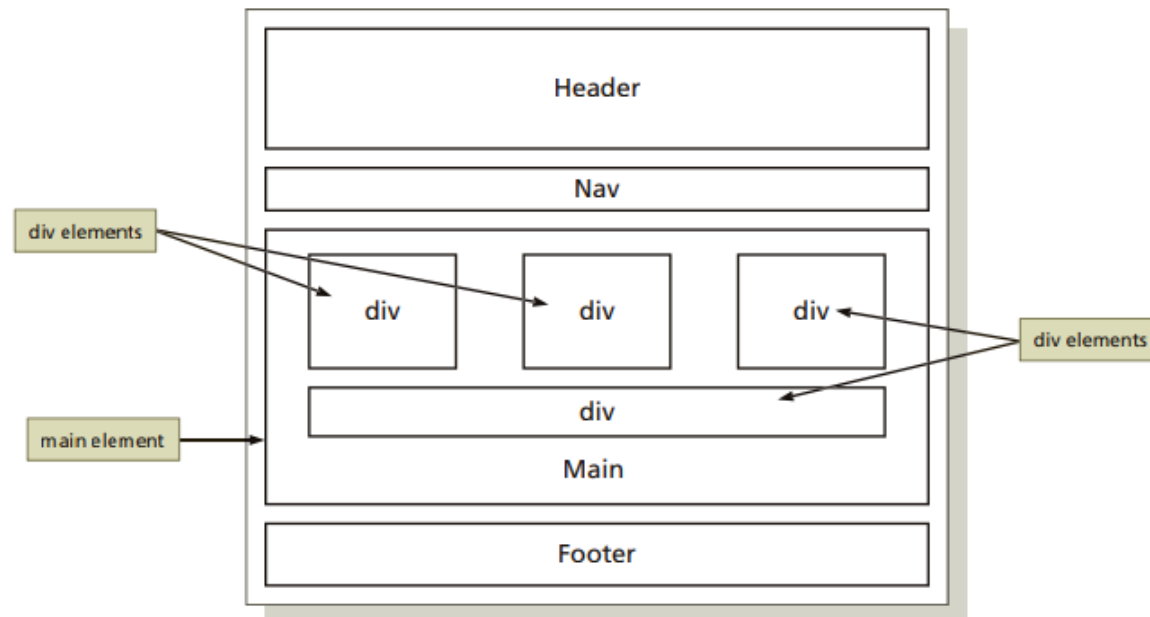




Gom nhóm các phần tử HTML với div

❑ Thẻ <div> trong HTML:

- ◆ Tuy nhiên, các nhà thiết kế web vẫn sử dụng phần tử div trên trang web của họ; vì div hỗ trợ tất cả các phiên bản trình duyệt từ cũ tới mới nhất.
- ◆ Thông thường, các nhà thiết kế web sử dụng các phần tử div để cấu trúc các phần của trang web mà phần tử HTML5 không hỗ trợ.





Gom nhóm các phần tử HTML với div

❑ Thẻ <div> trong HTML

◆ Ví dụ sử dụng **div**:

The image shows a Visual Studio Code editor on the left and a web browser on the right. The editor displays the HTML code for 'DivTag.html', which uses nested `<div>` tags with inline styles to create a list of frontend and backend topics. The browser on the right shows the rendered output, which consists of two main sections: a blue section for 'Frontend' topics and a red section for 'Backend' topics, each containing a list of specific technologies.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Div Tag</title>
5   </head>
6   <body>
7     <div style="background-color: blue; color: white;">
8       <h1>Đây là danh sách các khóa học Frontend</h1>
9       <h2>Frontend ReactJS</h2>
10      <h2>Frontend Angular</h2>
11    </div>
12    <div style="background-color: red; color: white;">
13      <h1>Đây là danh sách các khóa học Backend</h1>
14      <div style="background-color: purple; color: white;">
15        <h2>Backend Java</h2>
16      </div>
17      <div style="background-color: green; color: white;">
18        <h2>Backend PHP</h2>
19      </div>
20    </div>
21  </body>
22 </html>
```

Div Tag

127.0.0.1:5500/DivTag.html

Đây là danh sách các khóa học Frontend

Frontend ReactJS

Frontend Angular

Đây là danh sách các khóa học Backend

Backend Java

Backend PHP



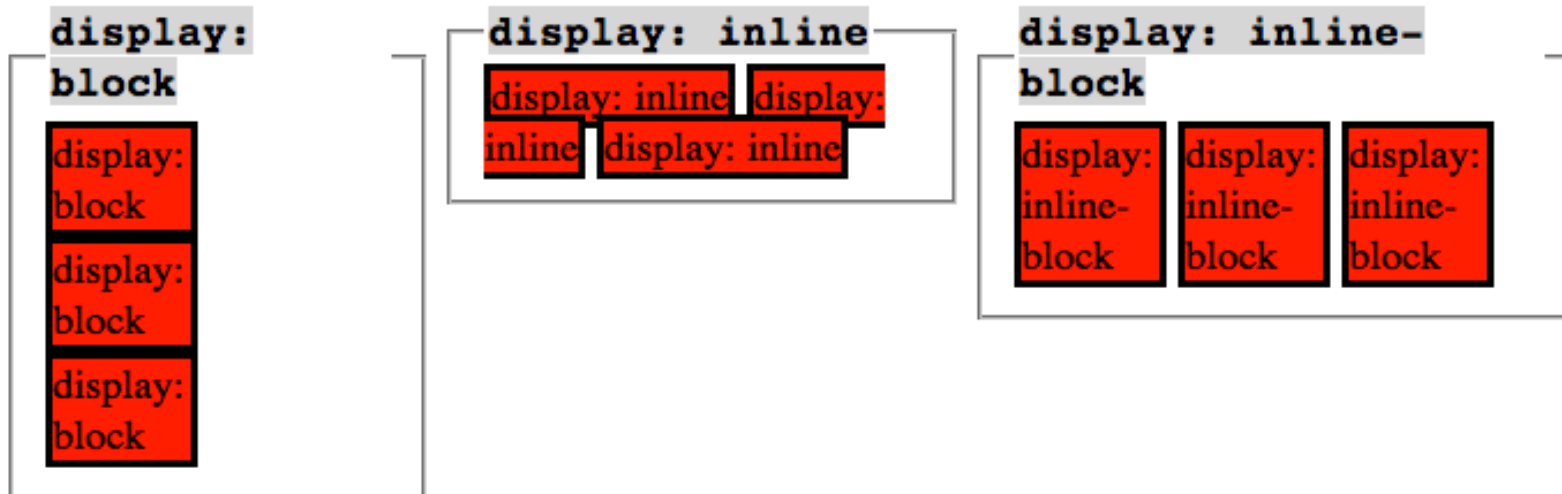
Cấp độ Block, Inline, Inline Block

❑ Các loại hiển thị trong HTML:

- ♦ Mỗi một phần tử HTML đều có một giá trị hiển thị mặc định.
- ♦ Điều đó tùy thuộc vào phần tử đó là loại gì.
- ♦ Trong HTML, chúng ta có 3 loại giá trị hiển thị: **block**, **inline** và **inline-block**

`block` vs `inline` vs `inline-block`

Below are a bunch of `<div style="width: 50px"...>` with different `display:` settings.





Cấp độ Block, Inline, Inline Block

❑ Cấp độ Block, Inline, Inline Block của các phần tử HTML:

Cấp độ Block	Cấp độ Inline
Một phần tử ở cấp độ Block thường bắt đầu trên một dòng	Một phần tử ở cấp độ Inline thường không bắt đầu trên một dòng mới
Một phần tử ở cấp độ Block thường chiếm toàn bộ chiều rộng có sẵn	Một phần tử ở cấp độ Inline thường chỉ chiếm nhiều chiều rộng khi cần thiết
<div> là một phần tử ở cấp độ Block	 là một phần tử ở cấp độ Inline
* Lưu ý: Kết hợp 2 tính chất của cấp độ Block và Inline ta được cấp độ Inline Block	

```
<div style="border: 1px solid black;">Hello <br>Block Attribute</div>
```

Hello
Block Attribute

Ví dụ về
cấp độ
Block

<p>Đây là thẻ span ở cấp độ Inline Hello World là phần tử bên trong một đoạn văn.</p>

<p>Phần tử thẻ SPAN là một phần tử ở cấp độ Inline, và không phải bắt đầu trên một dòng mới và chỉ chiếm một độ rộng cần thiết.</p>

Đây là thẻ span ở cấp độ Inline Hello World là phần tử bên trong một đoạn văn.

Phần tử thẻ SPAN là một phần tử ở cấp độ Inline, và không phải bắt đầu trên một dòng mới và chỉ chiếm một độ rộng cần thiết.

Ví dụ về
cấp độ
Inline



Cấp độ Block, Inline, Inline Block

❑ Một số lưu ý khi dùng Block và Inline

- ♦ Một phần tử ở cấp độ Inline **không thể chứa** một phần tử ở cấp độ Block.
- ♦ Phần tử `<div>`, `` không có thuộc tính bắt buộc nào cả, nhưng vẫn thường sử dụng các thuộc tính `style`, `class` và `id`.

- ♦ Phần tử `<div>`, `` thường được dùng chung với thuộc tính `<style>`.

- ♦ Ví dụ với thẻ `<div>`:

```
<div style="background-color:black;color:white;padding:20px;">
  <h2>London</h2>
  <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>
</div>
```

- ♦ Ví dụ với thẻ ``:

```
<p>My mother has <span style="color:blue;font-weight:bold">blue</span> eyes and
my father has <span style="color:darkolivegreen;font-weight:bold">dark
green</span> eyes.</p>
```




Các loại danh sách Lists trong HTML

❑ Các loại thẻ danh sách:

- ◆ Có 3 loại thẻ danh sách sau:

Kiểu danh sách	Thẻ danh sách	Thẻ phần tử
Danh sách có thứ tự	<code></code>	<code></code>
Danh sách không có thứ tự	<code></code>	<code></code>
Danh sách mô tả	<code><dl></dl></code>	<code><dt></dt></code> , <code><dd></dd></code>

- ◆ Mỗi loại thẻ danh sách sẽ **đặc trưng** cho một kiểu danh sách khác nhau.
- ◆ Mỗi loại thẻ danh sách cũng **sẽ có các loại thẻ phần tử** khác nhau.
- ◆ Ta chỉ **thường gặp 2 loại**: **danh sách có thứ tự**, **danh sách không có thứ tự**.



Các loại danh sách Lists trong HTML

❑ Các loại thẻ danh sách:

◆ Danh sách **có thứ tự**:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Danh sách có thứ tự (ol)</title>
5   </head>
6   <body>
7     <h4>Danh sách kiểu số (mặc định):</h4>
8     <ol>
9       <li>Frontend</li>
10      <li>Backend</li>
11      <li>Database</li>
12      <li>FullStack</li>
13    </ol>
14
15    <h4>Danh sách kiểu ký tự:</h4>
16    <ol type="a">
17      <li>Frontend</li>
18      <li>Backend</li>
19      <li>Database</li>
20      <li>FullStack</li>
21    </ol>
22
23    <h4>Danh sách kiểu ký tự la mã:</h4>
24    <ol type="i">
25      <li>Frontend</li>
26      <li>Backend</li>
27      <li>Database</li>
28      <li>FullStack</li>
29    </ol>
30  </body>
31 </html>
```

Danh sách có thứ tự (ol)

Danh sách kiểu số (mặc định):

1. Frontend
2. Backend
3. Database
4. FullStack

Danh sách kiểu ký tự:

- a. Frontend
- b. Backend
- c. Database
- d. FullStack

Danh sách kiểu ký tự la mã:

- i. Frontend
- ii. Backend
- iii. Database
- iv. FullStack

◆ Danh sách **không có thứ tự**:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Danh sách không có thứ tự (ul)</title>
5   </head>
6   <body>
7     <h4>Danh sách kiểu viên đạn Disc:</h4>
8     <ul>
9       <li>Frontend</li>
10      <li>Backend</li>
11      <li>Database</li>
12      <li>FullStack</li>
13    </ul>
14
15    <h4>Danh sách kiểu viên đạn Circle:</h4>
16    <ul type="circle">
17      <li>Frontend</li>
18      <li>Backend</li>
19      <li>Database</li>
20      <li>FullStack</li>
21    </ul>
22
23    <h4>Danh sách kiểu viên đạn Square:</h4>
24    <ul type="square">
25      <li>Frontend</li>
26      <li>Backend</li>
27      <li>Database</li>
28      <li>FullStack</li>
29    </ul>
30  </body>
31 </html>
```

Danh sách không có thứ tự (ul)

Danh sách kiểu viên đạn Disc:

- Frontend
- Backend
- Database
- FullStack

Danh sách kiểu viên đạn Circle:

- Frontend
- Backend
- Database
- FullStack

Danh sách kiểu viên đạn Square:

- Frontend
- Backend
- Database
- FullStack



Các loại danh sách Lists trong HTML

❑ Các loại thẻ danh sách:

◆ Danh sách **có thứ tự** **cải biên**:

The screenshot shows a Visual Studio Code editor with a file named `ListTagOL2.html`. The code defines an HTML document with two ordered lists. The first list is a simple ordered list with five items: Frontend, Backend, Database, FullStack, and DevOps. The second list is a more complex ordered list with three items: 'Đây là vị trí thứ 3 (ký tự c)', 'Đây là vị trí thứ 5 (ký tự e)', and 'Danh sách cũng có thể lồng nhau:'. The third item is further nested with two sub-items: 'Ta có thể lồng sâu không giới hạn.' and '...'. The browser preview on the right shows the rendered HTML, displaying the first list as a standard ordered list and the second list as a nested ordered list.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Danh sách có thứ tự (ol)</title>
5   </head>
6   <body>
7     <h4>Danh sách có thứ tự cũng có thể có nhiều loại:</h4>
8     <ol>
9       <li type="a">Frontend</li>
10      <li type="A">Backend</li>
11      <li type="i">Database</li>
12      <li type="I">FullStack</li>
13      <li type="1">DevOps</li>
14    </ol>
15
16    <h4>Danh sách có thứ tự cũng có thể bắt đầu bằng nhiều giá trị,
17    nhiều kiểu khác nhau:</h4>
18    <ol start="3" type="a">
19      <li>Đây là vị trí thứ 3 (ký tự c)</li>
20      <li value="5">Đây là vị trí thứ 5 (ký tự e)</li>
21      <ol start="7" type="1">
22        <li>Danh sách cũng có thể lồng nhau:</li>
23        <ol type="I">
24          <li value="2">Ta có thể lồng sâu không giới hạn.</li>
25          <li>...</li>
26        </ol>
27      </ol>
28    </ol>
29  </body>
30 </html>
```

Danh sách có thứ tự (ol)

Danh sách có thứ tự cũng có thể có nhiều loại:

- a. Frontend
- B. Backend
- iii. Database
- IV. FullStack
- 5. DevOps

Danh sách có thứ tự cũng có thể bắt đầu bằng nhiều giá trị, nhiều kiểu khác nhau:

- c. Đây là vị trí thứ 3 (ký tự c)
- e. Đây là vị trí thứ 5 (ký tự e)
 - 7. Danh sách cũng có thể lồng nhau:
 - II. Ta có thể lồng sâu không giới hạn.
 - III. ...



Thuộc tính đặc biệt style của HTML

❑ Giới thiệu thuộc tính style:

- ◆ Là một thuộc tính đặc biệt, giá trị của nó gồm một hoặc nhiều cặp **thuộc tính**: **giá trị**.
- ◆ Các thuộc tính được đặt trong thẻ mở dưới dạng sau:
`<tagname style="property: value; property: value;"> Nội dung... </tagname>`
- ◆ Với **property** là tên thuộc tính, **value** là giá trị thuộc tính.
- ◆ Một số thuộc tính style quan trọng:
 - **color**: thiết lập màu cho đoạn văn bản
 - **background-color**: thiết lập màu nền cho đoạn văn bản
 - **font-size**: thiết lập kích thước cho đoạn văn bản
 - **text-align**: căn lề cho đoạn văn bản



Thuộc tính đặc biệt style của HTML

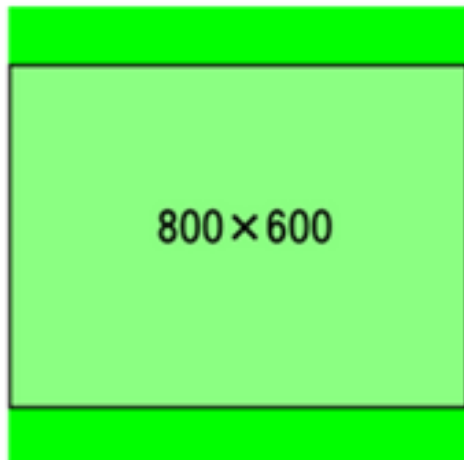
❑ Thuộc tính style width:

♦ **width**: thiết lập **chiều rộng** cho phần tử HTML (Đơn vị: **px, rem, em, %**.)

▪ Mặc định những phần tử block sẽ có **width = 100%** so với thành phần chứa nó.

▪ Cấu trúc như sau:

```
<div style="width: 100px;">Hello</div>
```



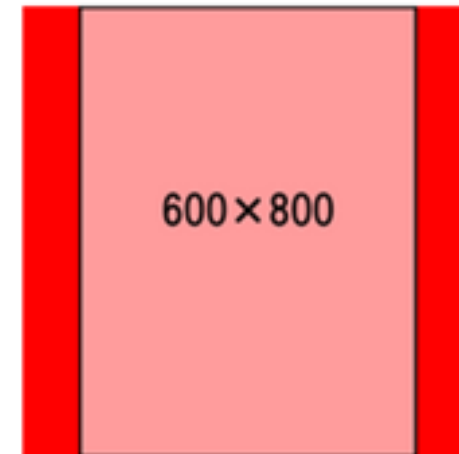
❑ Thuộc tính style height:

♦ **height**: thiết lập **chiều cao** cho phần tử HTML (Đơn vị: **px, rem, em, %**.)

▪ Mặc định những phần tử block sẽ có **height = với chiều cao các thành phần chứa nó**.

▪ Cấu trúc như sau:

```
<div style="height: 100px;">Hello</div>
```





Thuộc tính đặc biệt style của HTML

❑ Thuộc tính style color:

- ◆ Thiết lập **màu** cho **đoạn văn bản**.
- ◆ Có các loại giá trị:
 - Tên màu
 - Mã màu
 - ...

The screenshot shows a code editor with a file named 'StyleColorProperty.html'. The code is as follows:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Style Color Property</title>
5   </head>
6   <body>
7     <p style="color: blue;">Fast Track Coding Program</p>
8   </body>
9 </html>
```

Below the code editor, a browser window is shown with the title 'Style Color Property'. The address bar displays '127.0.0.1:5500/StyleColorProperty.html'. The rendered page shows the text 'Fast Track Coding Program' in blue.

❑ Thuộc tính style background-color:

- ◆ Thiết lập **màu nền** cho **đoạn văn bản**.
- ◆ Có các loại giá trị:
 - Tên màu
 - Mã màu
 - ...

The screenshot shows a code editor with a file named 'StyleBackgroundColorProperty.html'. The code is as follows:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Style Background Color Property</title>
5   </head>
6   <body>
7     <p style="background-color: yellow;">Fast Track Coding Program</p>
8   </body>
9 </html>
```

Below the code editor, a browser window is shown with the title 'Style Background Color Property'. The address bar displays '127.0.0.1:5500/StyleBackgroundColorProperty.html'. The rendered page shows the text 'Fast Track Coding Program' on a yellow background.



Thuộc tính đặc biệt style của HTML

❑ Thuộc tính style font-size:

- ◆ Thiết lập **kích thước** cho **đoạn văn bản**.
- ◆ Có các loại giá trị:
 - px, rem, em
 - ...

The screenshot shows a code editor with the following HTML code:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Style Font Size Property</title>
  </head>
  <body>
    <p style="font-size: 25px;">Fast Track Coding Program</p>
  </body>
</html>
```

Below the code editor, a browser window displays the rendered page. The title is "Style Font Size Property" and the text "Fast Track Coding Program" is shown in a larger font size.

❑ Thuộc tính style text-align:

- ◆ **Căn lề** cho **đoạn văn bản**.
- ◆ Có các loại giá trị:
 - left, right
 - center, justify

The screenshot shows a code editor with the following HTML code:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Style Text Align Property</title>
  </head>
  <body>
    <p style="text-align: left;">Chữ VUS nằm bên trái màn hình</p>
    <p style="text-align: right;">Chữ VUS nằm bên phải màn hình</p>
    <p style="text-align: center;">Chữ VUS nằm ở giữa màn hình</p>
    <p style="text-align: justify;">Chữ VUS được canh lề 2 bên lề nhé</p>
  </body>
</html>
```

Below the code editor, a browser window displays the rendered page. The text is aligned as follows:

- Chữ VUS nằm bên trái màn hình (Left-aligned)
- Chữ VUS nằm bên phải màn hình (Right-aligned)
- Chữ VUS nằm ở giữa màn hình (Center-aligned)
- Chữ VUS được canh lề 2 bên lề nhé (Justified)



Thuộc tính id, class của HTML

❑ Giới thiệu thuộc tính id:

- ◆ Thuộc tính **id** dùng để **chỉ định định danh duy nhất** cho một phần tử **HTML**.
- ◆ Giá trị của thuộc tính **id** phải là duy nhất trong HTML.
- ◆ Thuộc tính **id** được sử dụng để **trở đến một khai báo cụ thể trong một thẻ HTML**.
- ◆ Được **JavaScript** sử dụng để **truy cập và thao tác phần tử HTML với id cụ thể**.
- ◆ Cú pháp sử dụng **id**: viết một ký tự **#** (dấu thăng), theo sau là tên **id**. Sau đó, xác định các thuộc tính CSS trong dấu ngoặc nhọn **{}**.

◆ Chú ý: Tên **id** phân biệt chữ hoa, chữ thường. **Phải chứa** ít nhất một ký tự và không chứa khoảng trắng (dấu cách, tab,...)

```
<body>
  <h2>Thuộc tính id</h2>
  <p>Dùng CSS để định dạng kiểu dáng phần tử với id là myHeader:</p>

  <h1 id="myHeader">Header của tôi</h1>
</body>
```

```
#myHeader {
  background-color: lightblue;
  color: black;
  padding: 40px;
  text-align: center;
}
```



Thuộc tính id, class của HTML

❑ Giới thiệu thuộc tính class:

- ◆ Thuộc tính **class** thường **được dùng để đặt tên** cho **các thẻ dùng chung css**.
- ◆ Chúng ta **có thể** chuyển các css từ thuộc tính style thành mã css và **truy cập mã css** này **thông qua thuộc tính class**.
- ◆ Chúng ta sẽ sử dụng thuộc tính này nhiều hơn ở các bài học về css.

```
<h1>Weights</h1>
Our facility includes a weight training area with several weight op
<ul> ...
</ul><br>

<h1>Cardio</h1>
Burn fat through cardio workouts. If you need to lose 20 lbs or mor
<ul> ...
</ul><br>
```



Tạo bảng biểu Table, biểu mẫu Form

❑ Thẻ <table> trong HTML:

- ◆ Để khởi tạo một **bảng** ta dùng thẻ: `<table>`
- ◆ Để khởi tạo một **dòng** ta dùng thẻ: `<tr>` (Thẻ con của thẻ `<table>`)
- ◆ Để khởi tạo một **ô tiêu đề** ta dùng thẻ: `<th>` (Thẻ con của thẻ `<tr>`)
- ◆ Để khởi tạo một **ô** ta dùng thẻ: `<td>` (Thẻ con của thẻ `<tr>`)
- ◆ Ví dụ:

Bảng HTML (cơ bản)

Tên	Họ	Tuổi
Tý	Đinh	20
Tèo	Lý	25
Tô	Tiền	30

- ◆ Lưu ý: Các phần tử `<td>` là vùng chứa dữ liệu của bảng. Chúng có thể chứa tất cả các loại phần tử HTML: **văn bản, hình ảnh, danh sách, các bảng khác, v.v.**



Tạo bảng biểu Table, biểu mẫu Forms

❑ Thẻ <table> trong HTML

♦ Ví dụ 1: Bảng cơ bản

```
<h2>Bảng HTML (cơ bản)</h2>

<table style="width:100%">
  <tr>
    <th>Tên</th>
    <th>Họ</th>
    <th>Tuổi</th>
  </tr>
  <tr>
    <td>Tý</td>
    <td>Đình</td>
    <td>20</td>
  </tr>
  <tr>
    <td>Tèo</td>
    <td>Lý</td>
    <td>25</td>
  </tr>
  <tr>
    <td>Tồ</td>
    <td>Tiền</td>
    <td>30</td>
  </tr>
</table>
```

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5500/table_tag_01.html'. The page content is a table titled 'Bảng HTML (cơ bản)'. The table has three columns: 'Tên', 'Họ', and 'Tuổi'. The data rows are as follows:

Tên	Họ	Tuổi
Tý	Đình	20
Tèo	Lý	25
Tồ	Tiền	30



Tạo bảng biểu Table, biểu mẫu Forms

❑ Thẻ <table> trong HTML

♦ Ví dụ 2: Bảng với border

```
<h2>Bảng HTML (với border)</h2>

<table style="width:100%">
  <tr>
    <th>Tên</th>
    <th>Họ</th>
    <th>Tuổi</th>
  </tr>
  <tr>
    <td>Tý</td>
    <td>Đinh</td>
    <td>20</td>
  </tr>
  <tr>
    <td>Tèo</td>
    <td>Lý</td>
    <td>25</td>
  </tr>
  <tr>
    <td>Tồ</td>
    <td>Tiền</td>
    <td>30</td>
  </tr>
</table>
```

```
<style>
  table, th, td {
    border: 1px solid black;
  }
</style>
```

Bảng HTML (với border)

Tên	Họ	Tuổi
Tý	Đinh	20
Tèo	Lý	25
Tồ	Tiền	30

```
<style>
  table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
  }
</style>
```

Bảng HTML (với border đã thu gọn)

Tên	Họ	Tuổi
Tý	Đinh	20
Tèo	Lý	25
Tồ	Tiền	30



Tạo bảng biểu Table, biểu mẫu Form

❑ Thẻ <table> trong HTML

♦ Ví dụ 4: Bảng với tiêu đề canh trái

```
<h2>Bảng HTML (với tiêu đề canh trái)</h2>

<table style="width:100%">
  <tr>
    <th>Tên</th>
    <th>Họ</th>
    <th>Tuổi</th>
  </tr>
  <tr>
    <td>Tý</td>
    <td>Đinh</td>
    <td>20</td>
  </tr>
  <tr>
    <td>Tèo</td>
    <td>Lý</td>
    <td>25</td>
  </tr>
  <tr>
    <td>Tồ</td>
    <td>Tiền</td>
    <td>30</td>
  </tr>
</table>
```

```
table, th, td {
  border: 1px solid black;
  border-collapse: collapse;
}
```

```
th, td {
  padding: 5px;
}
th {
  text-align: left;
}
```

Tên	Họ	Tuổi
Tý	Đinh	20
Tèo	Lý	25
Tồ	Tiền	30



Tạo bảng biểu Table, biểu mẫu Form

❑ Một số thuộc tính thường dùng của thẻ <table>:

Thuộc tính	Ý nghĩa
colspan/ rowspan	Gộp cột/ Gộp dòng
width/ height	Thiết lập chiều rộng/dài cho bảng
background	Thiết lập ảnh nền cho bảng, ô
bgcolor	Thiết lập màu nền cho bảng, ô
align	Giống hàng ngang chữ trong ô (left, right, center, justify)
valign	Giống hàng dọc chữ trong ô (top, middle, bottom, baseline)
cellpadding	Khoảng cách từ đường biên của ô đến nội dung
cellspacing	Khoảng cách giữa các ô
boder	Thiết lập đường viền cho bảng, ô
border-collapse	Thu gọn đường viền giữa các ô trong bảng (css)



Tạo bảng biểu Table, biểu mẫu Form

❑ Giới thiệu thẻ biểu mẫu Form

♦ Một biểu mẫu HTML được sử dụng để **thu thập thông tin đầu vào của người dùng**. Đầu vào của người dùng **thường được gửi đến máy chủ để xử lý**.

♦ Phần tử HTML **<form>** được sử dụng để tạo một biểu mẫu HTML cho đầu vào của người dùng.

♦ Cấu trúc:

<form>

...

form elements

...

</form>

♦ Phần tử **<form>** là **vùng chứa cho các loại phần tử đầu vào khác nhau**, chẳng hạn như: **trường văn bản** (text fields), **hộp kiểm** (checkboxes), **nút radio** (radio button), **nút gửi** (submit button), v.v.



Tạo bảng biểu Table, biểu mẫu Form

❑ Thẻ `<input>` trong thẻ biểu mẫu Form

- ♦ Phần tử `<input>` là phần tử biểu mẫu được sử dụng nhiều nhất.
- ♦ Phần tử `<input>` có thể được hiển thị theo nhiều cách, tùy thuộc thuộc tính **type**.
- ♦ Các loại thuộc tính **type** của thẻ `<input>`:

Type	Mô tả
<code><input type="text"></code>	Hiển thị trường nhập văn bản một dòng
<code><input type="radio"></code>	Hiển thị một nút radio (để chọn một trong nhiều lựa chọn)
<code><input type="checkbox"></code>	Hiển thị hộp kiểm (để chọn không hoặc nhiều lựa chọn)
<code><input type="submit"></code>	Hiển thị nút gửi (để gửi biểu mẫu)
<code><input type="button"></code>	Hiển thị một nút có thể nhấp



Tạo bảng biểu Table, biểu mẫu Form

❑ Thẻ <input> trong thẻ biểu mẫu Form:

form_tag.html X

form_tag.html > html > body > p

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Form Tag</title>
5   </head>
6   <body>
7     <h2>HTML Forms</h2>
8
9     <form action="/action_page.php">
10       <label for="fname">Tên:</label><br>
11       <input type="text" id="fname" name="fname" value="Tèo"><br>
12       <label for="lname">Họ:</label><br>
13       <input type="text" id="lname" name="lname" value="Nguyễn"><br>
14       <input type="submit" value="Đăng ký"><br>
15     </form>
16     <p>Nếu bạn nhấn nút "Đăng ký", dữ liệu form sẽ được gửi tới một trang
17       gọi là /action_page.php</p>
18   </body>
19 </html>
```

Form Tag

127.0.0.1:5500/form_tag.html

HTML Forms

Tên:

Họ:

Nếu bạn nhấn nút "Đăng ký", dữ liệu form sẽ được gửi tới một trang gọi là /action_page.php



Tổng kết nội dung bài học

- ☐ Thêm các thẻ hình ảnh vào trang Web
- ☐ Gom nhóm các phần tử HTML với div
- ☐ Cấp độ Block, Inline và Inline Block
- ☐ Các loại danh sách Lists trong HTML
- ☐ Thuộc tính đặc biệt style của HTML
- ☐ Thuộc tính id, class của HTML
- ☐ Tạo bảng biểu Table, biểu mẫu Form

Let's
Recap

