

ỨNG DỤNG WEB

Nguyễn Thị Mai Trang

1

Chương 2

Ngôn ngữ HTML

2

Mục tiêu

- Xác định được đặc điểm và cấu trúc của trang HTML.
- Phân biệt được các thẻ HTML dùng để tạo ra các phần tử trên trang web, xác định được vai trò của từng phần tử và cách sử dụng chúng
- Vận dụng được cú pháp HTML và kết hợp các thẻ HTML để thiết kế trang web.
- Trình bày được một số đặc trưng mới của HTML5 và sử dụng được một số phần tử mới trong HTML5.

3

3

NỘI DUNG CHƯƠNG 2

1. Giới thiệu HTML
2. Các thẻ HTML cơ bản
3. Siêu liên kết
4. Danh sách
5. Bảng
6. Multimedia
7. Form và Control
8. Giới thiệu HTML5

Ứng dụng Web

4

4

2.1. Giới thiệu HTML

- HTML (**H**yper**T**ext **M**arkup **L**anguage) là một ngôn ngữ đánh dấu siêu văn bản để tạo ra các trang web.
- Tập tin có phần mở rộng là htm hoặc html.
- Nội dung tập tin HTML chứa các thẻ (tag) + các thuộc tính:
 - Trình bày hình thức và nội dung của trang web
 - Truy xuất thông tin trực tuyến bằng các liên kết
 - Thu thập thông tin người dùng bằng các biểu mẫu
 - ...
- Là một ngôn ngữ đơn giản và linh hoạt, tiêu chuẩn mở, miễn phí, không có chủ sở hữu → được phát triển mạnh mẽ.
- Các quy tắc tạo trang HTML được duy trì bởi W3C → chuẩn hóa ngôn ngữ web trên toàn thế giới.

5

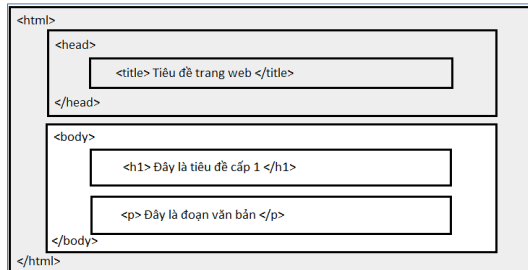
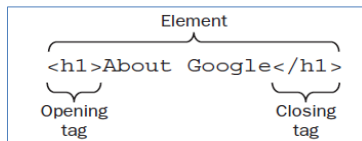
Giới thiệu ngôn ngữ HTML (tt)

- Các phiên bản HTML:
 - HTML 1.0 năm 1993
 - HTML 2.0 năm 1995
 - HTML 3.0 năm 1996
 - HTML 4.0 năm 1997
 - HTML 4.01 năm 1999
 - XHTML năm 2000
 - HTML5 năm 2014

6

Giới thiệu ngôn ngữ HTML (tt)

- Cấu trúc trang HTML
 - Tập hợp các phần tử (elements) và các thẻ (tags)



Ứng dụng Web

7

7

Giới thiệu ngôn ngữ HTML (tt)

- Cấu trúc trang HTML (tt)
 - Nên bắt đầu với thẻ `<!doctype>`: cho biết phiên bản HTML
 - Cặp thẻ `<html> </html>`: bắt đầu và kết thúc tài liệu HTML (bắt buộc)
 - Cặp thẻ `<head> </head>`: không bắt buộc
 - Chứa các thông tin như tiêu đề và mô tả trang, cách mã hóa tài liệu, nguồn tham chiếu đến dữ liệu JavaScript, CSS,... và các thông tin khác.
 - Nội dung nằm trong thẻ này không hiển thị trên trang web.
 - Cặp thẻ `<body> </body>`: bắt buộc
 - Chứa toàn bộ nội dung của trang web mà người dùng nhìn thấy trong cửa sổ trình duyệt.

Ứng dụng Web

8

8


Giới thiệu ngôn ngữ HTML (tt)

- Cấu trúc trang HTML, ví dụ:

```

<!DOCTYPE html>
<html>
<head>
  <title>Tieu de trang web!</title>
</head>
<body>
  <h1>Day la tieu de cap 1</h1>
  <p>Đây là nội dung đoạn văn bản thu nhất</p>
  <p>Đây là nội dung đoạn văn bản thu hai</p>
</body>
</html>

```



Tieu de trang web!

Day la tieu de cap 1

Day la noi dung doan van ban thu nhât

Day la noi dung doan van ban thu hai

Ứng dụng Web 9

9

Giới thiệu ngôn ngữ HTML (tt)

- Một số ký tự đặc biệt trong HTML (tham khảo thêm trong https://www.w3schools.com/html/html_symbols.asp):

©	©
™	™
®	®
“	"e;
Khoảng trắng:	
&	&
<	<
>	>

10

NỘI DUNG CHƯƠNG 2

1. Giới thiệu HTML
2. Các thẻ HTML cơ bản
3. Siêu liên kết
4. Danh sách
5. Bảng
6. Multimedia
7. Form và Control
8. Giới thiệu HTML5

11

2.2 Các thẻ HTML cơ bản

- Cú pháp, thuộc tính của thẻ
- Nhóm thẻ mô tả thông tin trang web
- Các thẻ về tiêu đề, đoạn
- Các thẻ định dạng và đánh dấu văn bản

12

2.2.1 Cú pháp, thuộc tính của thẻ

- Phần văn bản nằm giữa cặp ký tự **<** và **>** được gọi là một thẻ (tag).
- Mỗi thẻ được xác định bởi tên thẻ (tag name), có thể có thêm các thuộc tính để mở rộng chức năng của thẻ
- Tên thẻ thông thường là tên viết tắt chức năng của thẻ để dễ hiểu.
- Các thẻ có thể lồng nhau.

13

13

Cú pháp, thuộc tính của thẻ (tt)

- Mỗi thẻ gồm một thẻ bắt đầu và một thẻ kết thúc, có dạng:
 - Thẻ bắt đầu: **< tag_name >**
 - Thẻ kết thúc: **</ tag_name >**
- Phần văn bản nằm giữa thẻ bắt đầu và thẻ kết thúc là nội dung của phần tử.
 Ví dụ: **<i>** Đây là văn bản được định dạng in nghiêng **</i>**
- HTML tags (ngoại trừ XHTML) không phân biệt chữ hoa hay chữ thường (**** giống như ****)
- Có một số thẻ không có thẻ kết thúc, ví dụ **
**

14

14

Cú pháp, thuộc tính của thẻ (tt)

- Thẻ có thể có thuộc tính
 - Thuộc tính được đặt trong thẻ bắt đầu, có dạng:
`att_name = "value"` hoặc `att_name = 'value'`
 - Các giá trị của thuộc tính phải được để trong dấu ngoặc kép (" "), hoặc dấu nháy đơn (' ').
 - Nếu có nhiều thuộc tính, các thuộc tính phải cách nhau một khoảng trắng
 - Ví dụ:
 - `<body bgcolor="red">`
 - `<table bgcolor='blue' border='1'>`

15

15

2.2.2 Nhóm thẻ mô tả thông tin trang web

- Không hiển thị trong nội dung trang web
- Thường được sử dụng để mô tả thông tin trang web giúp trình duyệt nhận biết và hiển thị trang web một cách phù hợp
- Bao gồm:
 - `<!doctype>`
 - `<html>`
 - `<head>`
 - `<meta>`
 - `<title>`
 - `<body>`
 - `<!-- →`

16

16

Thẻ <!DOCTYPE>

- Được dùng để thông báo cho trình duyệt biết loại tài liệu và phiên bản của tài liệu để hiển thị nội dung trang web một cách phù hợp
- Tất cả tài liệu HTML nên bắt đầu bằng thẻ này
- Cách viết: không phân biệt chữ hoa, chữ thường :
 - <!DOCTYPE html>
 - <!DOCTYPE HTML>
 - <!doctype html>
 - <!Doctype Html>

17

17

Thẻ <!DOCTYPE> (tt)

- Khai báo phụ thuộc loại tài liệu và các phiên bản:
 - **HTML5:**
 <!DOCTYPE html>
 - **HTML 4.01:**
 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
 - **XHTML 1.0:**
 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

18

18

Phần tử HTML

- Cú pháp: `<html>...</html>`
- Bắt đầu và kết thúc một tài liệu HTML
- Cho biết là tất cả những nội dung nằm giữa cặp thẻ này được định nghĩa là mã HTML.

19

19

Phần tử head

- Cú pháp: `<head>...</head>`
- Xác định phần đầu của trang web, chứa các thông tin như tiêu đề, cách mã hóa tài liệu, nguồn tham chiếu đến dữ liệu JavaScript, CSS,...
- Nội dung nằm trong thẻ này không hiển thị trên giao diện của trình duyệt.
- Thẻ này có thể chứa các thẻ khác như:
 - `<title></title>`
 - `<style></style>`
 - `<script></script>`
 - `<link>`
 - `<meta>`
 - ...

20

20

Thẻ <META>

- Được dùng để cung cấp một số thông tin cho trình duyệt:
 - Định nghĩa mô tả trang web (description)
 - Định nghĩa từ khóa dùng trong các công cụ tìm kiếm (keywords)
 - Tên tác giả của tài liệu (author)
 - Lần sửa đổi lần cuối (last modified)
 - ...
- Dữ liệu được mô tả trong thẻ <meta> không hiển thị trên trang web, mà dùng cho các máy và các công cụ tìm kiếm khai thác.
- Thẻ <meta> luôn được đặt trong <head></head>

21

21

Thẻ <META>

- Có hai cách khai báo thẻ <meta>:
 - <meta name="Ten" content="Gia tri">

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

- <meta http-equiv="ten" content="Gia tri">

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
```

22

22

Thẻ <META>

• Các thuộc tính của thẻ <meta>:

Tên thuộc tính	Mô tả	Giá trị - ví dụ
charset	Xác định loại mã hoá ký tự cho tài liệu HTML	charset = "UTF-8"
http-equiv	Cung cấp thông tin HTTP header cho máy chủ web để tạo một hồi đáp tương ứng.	content-type default-style refresh
name	Thuộc tính tên	author description generator keywords
content	Định nghĩa giá trị của thuộc tính (kết hợp với thuộc tính name hoặc http-equiv)	http-equiv="content-type" content="text/html; charset=UTF-8"

23

23

Thẻ <META> (tt)

• Ví dụ:

<meta charset="UTF-8"> - chỉ định bộ mã cho ký tự là UTF-8

<meta name="description" content="Ứng dụng Web"> - định nghĩa mô tả trang web với nội dung là "Ứng dụng web"

<meta name="keywords" content="HTML,CSS,XML,JavaScript"> - định nghĩa các từ khóa cho công cụ tìm kiếm trang web là HTML, CSS, XML, JavaScript

<meta name="author" content="Khoa CNTT"> - Định nghĩa tên tác giả là Khoa CNTT

<meta http-equiv="refresh" content="30"> - tải lại trang web sau 30 giây

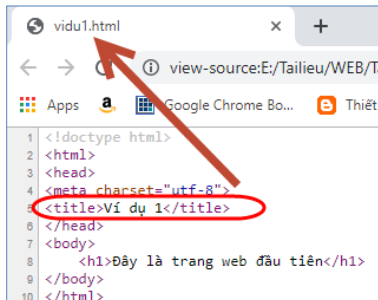
<meta http-equiv="refresh" content="3; url=file.htm"> - chuyển hướng trang web, sau 3 giây chuyển đến trang file.htm

24

24

Thẻ title

- Cú pháp: <title>...</title>
- Nằm trong phần tử <head>
- Hiện thị chuỗi tiêu đề của trang web trên trình duyệt.



```

1 <!doctype html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>Ví dụ 1</title>
6 </head>
7 <body>
8 <h1>Đây là trang web đầu tiên</h1>
9 </body>
10 </html>
  
```

25

25

Phần tử body

- Cú pháp: <body> ... </body>
- Chứa nội dung chính của trang web.
- Hầu hết các mã HTML và nội dung mà người dùng nhìn thấy trên trình duyệt đều nằm trong cặp thẻ này
- Một số thuộc tính thông dụng của thẻ <body>:
 - bgcolor: màu nền của trang web.
 - background: ảnh nền của trang web.
 - text: màu chữ trên trang web.

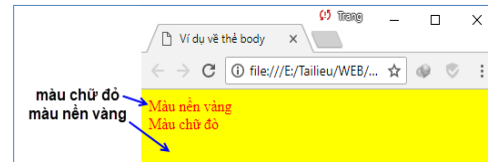
26

26

Thẻ body (tt)

- Ví dụ:

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Ví dụ về thẻ body</title>
</head>
<body bgcolor="yellow" text="red" >
    Màu nền vàng <br>
    Màu chữ đỏ
</body>
</html>
```



27

27

Chú thích

- Cú pháp **<!--** nội dung chú thích **-->**
- Phần nội dung nằm giữa thẻ này sẽ không hiển thị trên trình duyệt. Ví dụ:

```
<!-- Đây là chú thích, không hiển thị trên trình duyệt -->
<p>Đây là phần nội dung hiển thị trên trình duyệt.</p>
```

28

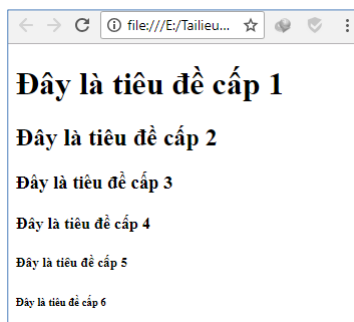
28

2.2.3 Các thẻ về tiêu đề, đoạn

- **Các thẻ tiêu đề:**

- Gồm các thẻ từ <h1> đến <h6>.
- Ví dụ:

```
<h1>Đây là tiêu đề cấp 1</h1>
<h2>Đây là tiêu đề cấp 2</h2>
<h3>Đây là tiêu đề cấp 3</h3>
<h4>Đây là tiêu đề cấp 4</h4>
<h5>Đây là tiêu đề cấp 5</h5>
<h6>Đây là tiêu đề cấp 6</h6>
```



29

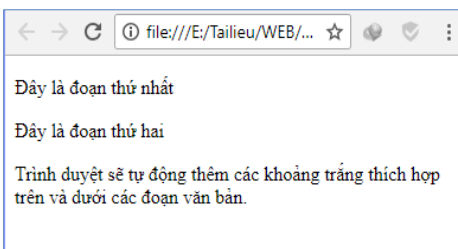
29

Các thẻ về tiêu đề, đoạn (tt)

- **Thẻ <p></p>:**

- Tạo một đoạn văn bản. Trình duyệt sẽ tự động thêm các khoảng trắng thích hợp trên và dưới các đoạn văn bản nằm trong thẻ này.
- Ví dụ:

```
<p>Đây là đoạn thứ nhất</p>
<p>Đây là đoạn thứ hai</p>
<p>Trình duyệt sẽ tự động thêm
các khoảng trắng thích hợp trên
và dưới các đoạn văn bản.</p>
```



30

30

Các thẻ về tiêu đề, đoạn (tt)

• Thẻ <div></div>:

- Được sử dụng để chứa các phần tử HTML khác khi cần kết hợp với CSS, Javascript trong định dạng và xử lý chung một nhóm các phần tử. Ví dụ:

```
<p>Đoạn văn bản nằm ngoài thẻ <div></p>
```

```
<div style="background-color:lightblue">
```

```
<h3>Tiêu đề <H3> nằm trong phần tử  
<div></h3>
```

```
<p>Đoạn văn bản nằm trong phần tử <div> có  
nền xanh nhạt</p>
```

```
</div>
```

```
<p>Đoạn văn bản nằm ngoài phần tử <div></p>
```

Đoạn văn bản nằm ngoài thẻ <div>

Tiêu đề <H3> nằm trong phần tử <div>

Đoạn văn bản nằm trong phần tử <div> có nền xanh nhạt

Đoạn văn bản nằm ngoài phần tử <div>

31

31

Các thẻ về tiêu đề, đoạn (tt)

• Thẻ <iframe></iframe>:

- Dùng để nhúng một tài liệu khác vào trang web hiện tại.
- Ví dụ:

```
<iframe width="640px" height="480px" src="http://ou.edu.vn">  
</iframe>
```

32

32

Các thẻ về tiêu đề, đoạn (tt)

• Thẻ
:

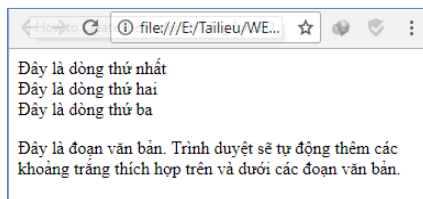
- Chèn một ngắt dòng, phần văn bản nằm sau thẻ này sẽ chuyển sang một dòng mới (tương tự Shift + Enter trong Word)
- Thẻ
 không có thẻ kết thúc
- Ví dụ:

Đây là dòng thứ nhất

Đây là dòng thứ hai

Đây là dòng thứ ba

<p> Đây là đoạn văn bản. Trình duyệt sẽ tự động thêm các khoảng trắng thích hợp trên và dưới các đoạn văn bản. </p>



33

33

Các thẻ về tiêu đề, đoạn (tt)

• Thẻ <hr>:

- Tạo một đường kẻ ngang trong tài liệu
- <hr> không có thẻ kết thúc.
- Một số thuộc tính của thẻ <hr>:
 - align: định lề cho đường kẻ, gồm các giá trị: left, right, center.
 - noshade: giá trị là noshade chỉ định đường kẻ là một màu (không có bóng)
 - size: độ dày của đường kẻ với đơn vị là px (pixel).
 - width: chiều rộng của đường kẻ theo đơn vị là pixel hoặc %.

34

34

Các thẻ về tiêu đề, đoạn (tt)

- Thẻ **<hr>**:

– Ví dụ:

`
`Đường kẻ mặc định:

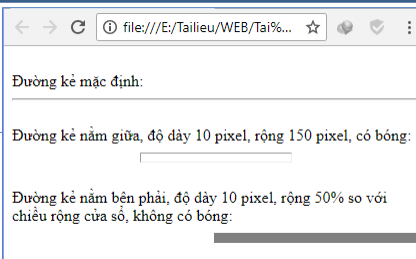
`<hr>`

`
`Đường kẻ nằm giữa, dày 10 pixel, rộng 150 pixel, có bóng:

`<hr align="center" size="10px" width="150px">`

`
`Đường kẻ nằm bên phải, dày 10 pixel, rộng 50% so với chiều rộng cửa sổ, không có bóng:

`<hr align="right" size="10px" width="50%" noshade>`



35

35

2.2.4 Các thẻ định dạng và đánh dấu văn bản

- ****: văn bản nằm bên trong thẻ được in đậm.
- **<i></i>**: văn bản nằm bên trong thẻ được in nghiêng.
- ****: văn bản nằm bên trong thẻ được đánh dấu quan trọng.
- ****: văn bản nằm bên trong thẻ được nhấn mạnh.
- **<mark></mark>**: văn bản nằm bên trong thẻ được tô sáng (highlight).
- **<small></small>**: văn bản nằm bên trong thẻ được định dạng với kích cỡ nhỏ.
- ****: văn bản nằm bên trong thẻ được gạch ngang (xóa).
- **<ins></ins>**: văn bản nằm bên trong thẻ hiển thị dạng như được chèn vào (có gạch chân).

36

36

Các thẻ định dạng và đánh dấu văn bản (tt)

- ****: văn bản nằm bên trong thẻ được định dạng thành chỉ số dưới.
- ****: văn bản nằm bên trong thẻ được định dạng thành chỉ số trên.
- **<q></q>**: văn bản nằm bên trong thẻ được đặt trong cặp nháy kép ("").
- **<blockquote></blockquote>**: chứa văn bản được trích dẫn từ một nguồn khác, thường được thụt lề khi hiển thị.
- **<address></address>**: chứa thông tin về địa chỉ tác giả, công ty,... được hiển thị dạng in nghiêng, có ngắt dòng.
- **<pre></pre>**: văn bản trong thẻ này được hiển thị với phông chữ có chiều rộng cố định (thường là Courier), giữ nguyên khoảng trắng, dấu ngắt dòng như khi soạn thảo

37

37

Các thẻ định dạng và đánh dấu văn bản (tt)


```

<i> Đây là văn bản được in nghiêng </i>
<br>Chú ý, phần này <strong>quan trọng</strong>
<br>Đây là phần được <em>nhấn mạnh</em>
<br>Đây là văn bản được <mark>đánh dấu</mark>
<br>VĂN BẢN BÌNH THƯỜNG <small>VĂN BẢN CỖ NHỎ</small>
<br>Văn bản này <del>phải được xóa</del>
<br>Văn bản <ins>chèn thêm</ins>
<br>Công thức nước là H<sub>2</sub></sub>O
<br>Bình phương của x được viết là x<sup>2</sup></sup>
<br>HTML là viết tắt của <q>HyperText Markup Language</q>
<br><blockquote>HTML là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web
<address>Wikipedia</address> </blockquote>
<br><b> Chương trình C++ đơn giản:</b>
<pre>
    int main()
    {
        cout << "Chương trình C++ đầu tiên" << endl;
        return 0;
    }
</pre>

```

38

38



TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Các thẻ định dạng và đánh dấu văn bản (tt)

Đây là văn bản được in nghiêng → `<i>`

Chú ý, phần này **quan trọng** → ``

Đây là phần được *nhấn mạnh* → ``

Đây là văn bản được **đánh dấu** → `<mark>`

VĂN BẢN BÌNH THƯỜNG VĂN BẢN CỎ NHỎ → `<small>`

Văn bản này ~~phải~~ được xóa → ``

Văn bản chèn thêm → `<ins>`

Công thức nước là H₂O → `<sub>`

Bình phương của x được viết là x² → `<sup>`

HTML là viết tắt của "HyperText Markup Language" → `<q>`

`<blockquote>` HTML là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web Wikipedia → `<address>`


Chương trình C++ đơn giản: → ``

`<pre>`

```
int main()
{
    cout << "Chương trình C++ đầu tiên" << endl;
    return 0;
}
```

39

39



TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

NỘI DUNG CHƯƠNG 2

1. Giới thiệu HTML
2. Các thẻ HTML cơ bản
3. Siêu liên kết
4. Danh sách
5. Bảng
6. Multimedia
7. Form và Control
8. Giới thiệu HTML5

40

40

2.3 Siêu liên kết (Hyperlink)

- Siêu liên kết là một phần tử bên trong tài liệu liên kết đến một vị trí khác trong cùng tài liệu đó hoặc chuyển đến một tập tin, một trang web khác.
- Khi click vào siêu liên kết, người dùng được đưa đến địa chỉ đã chỉ định trong liên kết (URL).
- Có hai loại liên kết:
 - Liên kết trong (liên kết nội): chỉ đến các vị trí khác trong cùng tài liệu.
 - Liên kết ngoài: chỉ đến các tập tin khác trên website hoặc máy chủ khác.

41

41

Siêu liên kết (tt)

- Một siêu liên kết có hai thành phần:
 - Địa chỉ URL của vị trí/file được liên kết đến.
 - Nội dung hiển thị trên liên kết, có thể là văn bản, hình ảnh,...
- Tạo siêu liên kết theo cú pháp:

** nội dung hiển thị trên liên kết **

42

42

Siêu liên kết (tt)

- Một số thuộc tính của thẻ <a>:
 - **href**: địa chỉ URL đến tập tin hoặc một vị trí liên kết đến, có thể là địa chỉ tuyệt đối hoặc địa chỉ tương đối.
 - **target**: định nghĩa cách mở tập tin liên kết, gồm các giá trị:
 - **_blank**: mở trang liên kết trong tab mới
 - **_self**: mặc định, mở trang liên kết tại cửa sổ hiện hành
 - **_parent**: mở trang liên kết trên cửa sổ cha
 - **_top**: chuyển tới tab hiện tại, thường dùng khi muốn thoát khỏi iframe.
 - **title**: tooltip (chú thích) khi con trỏ chuột nằm trên liên kết.

43

43

Siêu liên kết (tt)

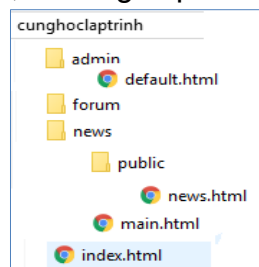
- Địa chỉ tuyệt đối:
 - Có dạng: protocol://host.domain[:port]/file.
 - Trong đó:
 - protocol là tên giao thức, ví dụ: http, https,...
 - host.domain: tên miền của máy chủ web.
 - port: tùy chọn, mặc định giao thức http sử dụng port 80.
 - /: vị trí thư mục gốc của website
 - file: đường dẫn đến tập tin cần truy xuất.
 - Ví dụ: <http://cunghoclaphinh.com/admin/index.html> là một địa chỉ tuyệt đối.

44

44

Siêu liên kết (tt)

- Địa chỉ tương đối:
 - Là vị trí tương đối so với trang web mà liên kết hướng tới.
 - Là một phần của đường dẫn tuyệt đối, thường là phần đường dẫn đến tập tin.
 - Giả sử ta có thư mục web như sau:



45

45

Siêu liên kết (tt)

- Địa chỉ tương đối (tt):
 - Từ tập tin index.html, tạo liên kết đến tập tin admin/default.html:
 - ` Quản trị `
 - Từ tập tin news/main.html, tạo liên kết đến tập tin admin/default.html:
 - ` Quản trị `
 - Từ tập tin news/public/news.html, tạo liên kết đến tập tin admin/default.html:
 - ` Quản trị `
 - Trong đó ký tự .. được dùng để chỉ đến thư mục cha của thư mục hiện hành.

46

46

Siêu liên kết (tt)

- Liên kết nội (Internal):

- Liên kết chuyển đến một vị trí khác trên cùng một trang web

- Tạo liên kết nội:

- Đánh dấu (bookmark) các vị trí muốn liên kết đến, bằng cách thiết lập thuộc tính **id** cho các phần tử.

- Ví dụ:
một phần nội dung trong tập tin
có tên là baigiang.html

```
<h2 id="ch01">Chương 1 </h2>
<!-- nội dung chương 1 đặt ở đây -->
<h2 id="ch02">Chương 2 </h2>
<!-- nội dung chương 2 đặt ở đây -->
<h2 id="ch03">Chương 3 </h2>
<!-- nội dung chương 3 đặt ở đây -->
```

47

47

Siêu liên kết (tt)

- Liên kết nội (tt):

- Đầu tập tin, tạo mục lục liên kết đến các phần tử đã bookmark:

```
<h1>Mục lục </h1>
<a href="#ch01">Xem nội dung chương 1 </a> <br>
<a href="#ch02">Xem nội dung chương 2 </a> <br>
<a href="#ch03">Xem nội dung chương 3 </a>
```

- Kết quả hiển thị trên trình duyệt:



48

48

Siêu liên kết (tt)

- Liên kết ngoài (External):
 - Liên kết chuyển đến trang web khác
 - Ví dụ: <a href=<http://cunghoclaptrinh.com>> Web học lập trình
 - Liên kết đến một vị trí được chỉ định trong trang web khác:
 - Tạo liên kết đến một bookmark trên trang web khác (tương tự liên kết nội).
 - Ví dụ: hai tập tin ontap.html, baigiang.html nằm trong cùng một thư mục. Trên tập tin ontap.html, ta có thể tạo liên kết để xem nội dung chương 3 trong tập tin baigiang.html theo cú pháp như sau:
 Chương 3

49

49

Siêu liên kết (tt)

- Liên kết đến một địa chỉ email
 - Cú pháp: <a href = "<mailto:diachimail>" > Chuỗi hiển thị
 - Khi người dùng click vào liên kết, chương trình mail client trên máy tính của người dùng sẽ được kích hoạt và địa chỉ email trên liên kết sẽ được điền vào vị trí người nhận.
 - Ví dụ

Nếu có thắc mắc về diễn đàn, xin liên hệ đến địa chỉ email của
 Quản trị viên
 - Kết quả hiển thị trên trình duyệt



50

50

Siêu liên kết (tt)

- Liên kết tải về một tập tin trên một FTP server
 - Cú pháp:
 - <a href = "<ftp://server/path/filename>" > Chuỗi hiển thị
 - Ví dụ:

```
<a href = "ftp://ftp.cunghoclaptrinh.com/slides.rar" >
  Tải về bài giảng Ứng dụng web
</a>
```

51

51

NỘI DUNG CHƯƠNG 2

1. Giới thiệu HTML
2. Các thẻ HTML cơ bản
3. Siêu liên kết
4. Danh sách
5. Bảng
6. Multimedia
7. Form và Control
8. Giới thiệu HTML5

Ứng dụng Web

52

52

2.4 Danh sách

- Danh sách không có thứ tự (Unordered list): sử dụng các ký tự đầu dòng đặt trước mỗi mục trong danh sách, được sử dụng khi thứ tự các phần tử trong danh sách không quan trọng.
- Danh sách có thứ tự (Ordered list): sử dụng các chữ số hoặc chữ cái đặt trước mỗi mục trong danh sách theo thứ tự tăng dần.
- Danh sách định nghĩa (Definition lists): mỗi mục là một thuật ngữ và phần định nghĩa (mô tả/giải thích) cho thuật ngữ đó.

Ứng dụng Web

53

53

Danh sách

- Danh sách không có thứ tự:
 - Sử dụng thẻ ``.
 - Mỗi mục trong danh sách sử dụng thẻ ``.
 - Ví dụ:

```
<h2> Các trình duyệt phổ biến </h2>
<ul>
<li>Firefox</li>
<li>Chrome</li>
<li>Internet Explorer</li>
<li>Opera</li>
</ul>
```

Các trình duyệt phổ biến

- Firefox
- Chrome
- Internet Explorer
- Opera

Ứng dụng Web

54

54

Danh sách

- Danh sách không có thứ tự (tt):
 - Thuộc tính **compact**: thuộc tính luận lý, nếu có, danh sách sẽ hiển thị nhỏ hơn bình thường, bằng cách giảm khoảng cách giữa các dòng và thụt lề của danh sách.

Ví dụ:

```
<h2> Các trình duyệt phổ biến </h2>
<ul compact>
  <li>Firefox</li>
  <li>Chrome</li>
  <li>Internet Explorer</li>
  <li>Opera</li>
</ul>
```

Ứng dụng Web

55

55

Danh sách

- Danh sách không có thứ tự (tt):
 - Thuộc tính **type**: loại ký tự
 - disc: mặc định, dạng hình tròn tô nền
 - circle: hình tròn không tô nền
 - square: hình vuông có tô nền
 - none: không có ký hiệu đầu dòng

```
<h2>Các trình duyệt phổ biến</h2>
<ul type="circle">
  <li>Firefox</li>
  <li>Chrome</li>
  <li>Internet Explorer</li>
  <li>Opera</li>
</ul>
```

Các trình duyệt phổ biến

- Firefox
- Chrome
- Internet Explorer
- Opera

Ứng dụng Web

56

56

Danh sách

- Danh sách có thứ tự:

- Sử dụng thẻ `` ``.
- Mỗi mục trong danh sách sử dụng thẻ `` ``.
- Ví dụ:

```
<h2>Top 5 ngôn ngữ lập trình web phổ biến nhất</h2>
<ol>
  <li>JavaScript</li>
  <li>Java</li>
  <li>Python</li>
  <li>PHP</li>
  <li>C#</li>
</ol>
```

Top 5 ngôn ngữ lập trình web phổ biến nhất

1. JavaScript
2. Java
3. Python
4. PHP
5. C#

Danh sách

- Danh sách có thứ tự (tt):

- Thuộc tính **type**, gồm các giá trị:
 - 1 (chữ số 1, 2, 3, 4,... phía trước danh sách, mặc định)
 - a (chữ cái thường a, b, c, d,... phía trước danh sách)
 - A (chữ cái in hoa A, B, C, D,... phía trước danh sách)
 - i (chữ số la mã thường i, ii, iii, iv,... phía trước danh sách)
 - I (chữ số la mã in hoa I, II, III, IV,... phía trước danh sách)

Danh sách

- Danh sách có thứ tự (tt):
 - Thuộc tính **start**: quy định thứ tự bắt đầu của danh sách
 - Ví dụ:

```
<h2>Top 5 ngôn ngữ lập trình web phổ biến tiếp theo</h2>
```

```
<ol start="6">
```

```
<li>Ruby</li>
```

```
<li>Go</li>
```

```
<li>HTML</li>
```

```
<li>SQL</li>
```

```
<li>CSS</li>
```

```
</ol>
```

Top 5 ngôn ngữ lập trình web phổ biến tiếp theo

6. Ruby
7. Go
8. HTML
9. SQL
10. CSS

Theo [Stack Overflow 2020](#)

Ứng dụng Web

59

59

Danh sách

- Danh sách nhiều cấp:
 - Được tạo bằng cách lồng các thẻ ``, ``, `` với nhau

Bài giảng HTML

- I. Chương 1
 - Giới thiệu HTML
 - Cấu trúc tài liệu HTML
- II. Chương 2
 - Các thẻ cơ bản
 - Đa phương tiện
 - Forms

```
<h2>Bài giảng HTML</h2>
```

```
<ol type="I">
```

```
<li>Chương 1
```

```
<ul>
```

```
<li>Giới thiệu HTML</li>
```

```
<li>Cấu trúc tài liệu HTML</li>
```

```
</ul>
```

```
</li>
```

```
<li>Chương 2
```

```
<ul>
```

```
<li>Các thẻ cơ bản</li>
```

```
<li>Đa phương tiện</li>
```

```
<li>Forms</li>
```

```
</ul>
```

```
</li>
```

```
</ol>
```

Ứng dụng Web

60

60

Danh sách

- **Danh sách định nghĩa:**

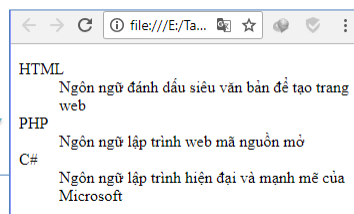
- Là loại danh sách mà mỗi phần tử là một từ/cụm từ và phần mô tả của chúng.
- Để tạo danh sách loại này, cần sử dụng ba loại thẻ sau:
 - `<dl></dl>`: thẻ mở đầu và kết thúc một danh sách.
 - `<dt></dt>`: thuật ngữ cho mỗi phần tử.
 - `<dd></dd>`: phần mô tả cho thuật ngữ.

61

Danh sách

- **Danh sách định nghĩa: ví dụ**

```
<dl>
  <dt>HTML</dt>
  <dd>Ngôn ngữ đánh dấu siêu văn bản để tạo trang web</dd>
  <dt>PHP</dt>
  <dd>Ngôn ngữ lập trình web mã nguồn mở </dd>
  <dt>C#</dt>
  <dd>Ngôn ngữ lập trình ứng dụng hiện đại và mạnh mẽ của Microsoft</dd>
</dl>
```



62

NỘI DUNG CHƯƠNG 2

1. Giới thiệu HTML
2. Các thẻ HTML cơ bản
3. Siêu liên kết
4. Danh sách
5. Bảng
6. Multimedia
7. Form và Control
8. Giới thiệu HTML5

63

2.5 Bảng (Table)

- Thường được sử dụng để trình bày dữ liệu lấy từ cơ sở dữ liệu:
 - Giỏ hàng trong các website thương mại điện tử
 - Hóa đơn mua hàng
 - ...
- Cấu trúc bảng bao gồm:
 - Các hàng (rows)
 - Các cột (columns)
 - Các ô (cells): là giao giữa hàng và cột

64

Bảng (Table)

- Các thẻ tạo bảng
 - **<table>** **</table>**: thẻ bắt đầu và kết thúc một bảng.
 - **<tr>** **</tr>**: thẻ tạo các hàng (dòng).
 - **<td>** **</td>**: thẻ tạo các ô trong mỗi hàng.
 - **<th>** **</th>**:
 - Thẻ tạo các ô tiêu đề (thường làm tiêu đề cột trong bảng).
 - Nội dung nằm trong thẻ này được tự động canh giữa và in đậm.
 - **<caption>** **</caption>**: thẻ tạo tiêu đề cho bảng

Bảng (Table)

- Các thẻ tạo bảng (tt)
 - **<tbody>** **</tbody>**: đặt bên trong thẻ **<table>**, dùng để nhóm các nội dung trong bảng, thường dùng kết hợp với hai thẻ **<thead>** và **<tfoot>**
 - **<thead>** **</thead>**: được dùng để nhóm các hàng là tiêu đề đầu trong bảng.
 - **<tfoot>** **</tfoot>**: được dùng để nhóm các hàng là tiêu đề cuối trong bảng
 - Trình duyệt dựa vào các phần tử này để cho phép cuộn phần nội dung trong **<tbody>** độc lập với phần **<thead>** và **<tfoot>**.
 - Khi in một bảng có nhiều trang, nội dung trong phần **<thead>** và **<tfoot>** luôn được in ở trên và dưới của mỗi trang.

Bảng (Table)

```
<table border = "1">
  <caption><h2>Kết quả học tập</h2></caption>
  <thead>
    <tr> <th>Họ</th> <th>Tên</th> <th>Điểm</th> </tr>
  </thead>
  <tbody>
    <tr> <td>Trần Văn</td> <td>Thành</td> <td>9</td> </tr>
    <tr> <td>Lý Thị</td> <td>Hoa</td> <td>7</td> </tr>
    <tr> <td>Hồ Thanh</td> <td>Trúc</td> <td>4</td> </tr>
  </tbody>
  <tfoot>
    <tr> <td>Điểm cao nhất:</td> <td></td> <td>9</td> </tr>
  </tfoot>
</table>
```

Kết quả học tập

Họ	Tên	Điểm
Trần Văn	Thành	9
Lý Thị	Hoa	7
Hồ Thanh	Trúc	4
Điểm cao nhất:		9

Ứng dụng Web

67

67

Bảng (Table)

• Các thuộc tính định dạng cho bảng

- **border**: độ dày của đường viền bao xung quanh bảng, mặc định bảng không có đường viền.
- **cellpadding**: khoảng cách giữa văn bản trong ô và đường viền của ô.
- **cellspacing**: khoảng cách giữa các ô.
- **bgcolor**: màu nền của bảng, hàng, ô.
 - Trong thẻ <table>: thiết lập màu nền cho toàn bảng.
 - Trong thẻ <tr>: thiết lập màu nền cho hàng.
 - Trong thẻ <td> hoặc <th>: thiết lập màu nền cho ô.

Ứng dụng Web

68

68

Bảng (Table)

• Các thuộc tính định dạng cho bảng (tt)

- **width**: độ rộng của bảng, dòng, ô với đơn vị là px (pixel) hoặc % (tỉ lệ % của phần tử so với phần tử chứa nó)
- **height**: chiều cao của bảng, dòng, ô.
- **align**: chỉ dùng trong thẻ <th> và <td>, dùng để canh lề nội dung trong ô, gồm các giá trị:
 - **left** (canh trái, mặc định)
 - **right** (canh phải)
 - **center** (canh giữa)
 - **justify** (canh đều hai bên)

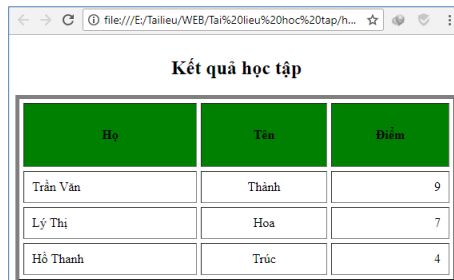
Ứng dụng Web

69

69

Bảng (Table)

```
<table border="5px" cellpadding="10px" cellspacing="5px" width="100%">
  <caption><h2>Kết quả học tập</h2></caption>
  <tr bgcolor="green" height="80px">
    <th>Họ</th>
    <th>Tên</th>
    <th>Điểm</th>
  </tr>
  <tr>
    <td>Trần Văn</td>
    <td align="center">Thành</td>
    <td align="right">9</td>
  </tr>
  <tr>
    <td>Lý Thị</td>
    <td align="center">Hoa</td>
    <td align="right">7</td>
  </tr>
  ...
</table>
```



Họ	Tên	Điểm
Trần Văn	Thành	9
Lý Thị	Hoa	7
Hồ Thanh	Trúc	4

Ứng dụng Web

70

70

Bảng (Table)

- Có thể sử dụng CSS để định dạng bảng với thuộc tính **style**. Ví dụ:

```
<table border = "5" style="width:100%; border-collapse: collapse;">
```

Kết quả học tập		
Họ	Tên	Điểm
Trần Văn	Thành	9
Lý Thị	Hoa	7
Hồ Thanh	Trúc	4

71

Bảng (Table)

- Trộn ô trong bảng:** sử dụng các thuộc tính **rowspan**, **colspan** đặt trong thẻ <th> hoặc <td>.
 - rowspan: trộn các ô liên kề trong một hàng.
 - colspan: trộn các ô liên kề trong một cột.

72

Bảng (Table)

• Trộn ô (tt):

Kết quả học tập			
Họ	Tên	Kết quả	
		Đầu	Rớt
Trần Văn	Thành	9	
Lý Thị	Hoa	7	
Hồ Thanh	Trúc		4

```

<table border = "1" width="100%">
<caption><h2>Kết quả học tập</h2></caption>
<tr>
  <th rowspan="2">Họ</th>
  <th rowspan="2">Tên</th>
  <th colspan="2">Kết quả</th>
</tr>
<tr>
  <th>Đầu</th>
  <th>Rớt</th>
</tr>
<tr>
  <td>Trần Văn</td>
  <td align="center">Thành</td>
  <td align="center">9</td>
  <td></td>
</tr>
</table>

```

Ứng dụng Web

73

73

NỘI DUNG CHƯƠNG 2

1. Giới thiệu HTML
2. Các thẻ HTML cơ bản
3. Siêu liên kết
4. Danh sách
5. Bảng
6. Multimedia
7. Form và Control
8. Giới thiệu HTML5

Ứng dụng Web

74

74

2.6 Multimedia

- Multimedia thường được dùng trong trang web để:
 - Tăng tính thẩm mỹ và thu hút người xem.
 - Quảng bá thương hiệu cho công ty, doanh nghiệp
 - Minh họa cho các bài báo, bài tham luận
 - Quảng cáo sản phẩm trong thương mại điện tử,
 - ...
- Multimedia phổ biến nhất là hình ảnh, audio, video

75

75

Hình ảnh

- Các trình duyệt ngày nay hỗ trợ khá nhiều loại định dạng hình ảnh như:
 - GIF (8 bits)
 - Animated GIF
 - JPG (24 bits)
 - PNG (48 bits)
 - JFIF
- Cần chú ý đến sự cân đối giữa chất lượng và dung lượng của tập tin: tập tin kích thước lớn → load chậm

76

76

Hình ảnh

- **Sử dụng liên kết ngoài:**

- Trên trang web chứa liên kết đến tập tin hình ảnh.
- Click vào liên kết → hình ảnh sẽ thay thế nội dung trang hiện hành trong trình duyệt web.
- Ví dụ: ` Hoa hồng `

77

77

Hình ảnh (tt)

- **Nhúng hình ảnh vào trang web:**

- Hình ảnh được nhúng vào trang web, hiển thị cùng với văn bản.
- Sử dụng thẻ ``

```

<p>
```

Với vẻ đẹp, hình dáng cũng hương thơm nổi bật, hoa hồng là hoa biểu trưng hay được dùng nhất ở phương Tây, tượng ứng tổng thể với hình tượng hoa sen ở châu Á. Bên cạnh đó, hoa hồng còn tượng trưng cho phần thưởng cuộc sống, tình hân, trái tim, tình yêu.

```
</p>
```



Với vẻ đẹp, hình dáng cũng hương thơm nổi bật, hoa hồng là hoa biểu trưng hay được dùng nhất ở phương Tây, tượng ứng tổng thể với hình tượng hoa sen ở châu Á. Bên cạnh đó, hoa hồng còn tượng trưng cho phần thưởng cuộc sống, tình hân, trái tim, tình yêu.

78

78

Hình ảnh (tt)

• Các thuộc tính của phần tử ``:

- **src**: địa chỉ URL của tập tin hình ảnh.
- **height**: chiều cao của hình với đơn vị là px (pixel) hoặc %.
- **width**: chiều rộng của hình.
- **align**: canh lề cho hình ảnh và văn bản xung quanh, gồm các giá trị: top, middle, bottom, left, right.
- **alt**: chuỗi văn bản hiển thị để thay thế khi URL của tập tin sai
- **border**: độ dày của đường viền với đơn vị là px, giá trị mặc định là 0.

79

79

Hình ảnh (tt)

• Ví dụ:

`<h3>Không thiết lập thuộc tính align</h3>`

`<p> Táo tây, còn gọi là bôm ...
`

``

`Đây là một trong những loại cây ăn trái phổ biến nhất... </p>`

`<h3>Thiết lập thuộc tính align="left"</h3>`

`<p> Chuối là một loại cây vô cùng quen thuộc...
`

``

`Cây chuối là loại cây vô cùng phổ biến ở quốc gia nhiệt đới ẩm gió mùa như Việt Nam...</p>`

`<h3>Thiết lập thuộc tính align="right"</h3>`

`<p> Cam được trồng rộng rãi ở những nơi có khí hậu ẩm áp...
`

``

`Vỏ cam dày, có vị đắng,... </p>`

80

80

Hình ảnh (tt)

Không thiết lập thuộc tính align

Táo tây, còn gọi là bôm (phiên âm từ tiếng Pháp: pomme).



```

```

Đây là một trong những loại cây ăn trái phổ biến nhất. Táo thường được ăn sống, nhưng cũng có thể được chế biến thành nhiều thực phẩm khác và thức uống

Thiết lập thuộc tính align="left"

Chuối là một loại cây vô cùng quen thuộc trong đời sống hàng ngày, đặc biệt là trong các vùng nông thôn Việt Nam.



Cây chuối là loại cây dễ trồng, dễ sống và thường mang lại năng suất cao cho những người nông dân

```

```

Thiết lập thuộc tính align="right"

Cam được trồng rộng rãi ở những nơi có khí hậu ẩm áp, và vị cam có thể biến đổi từ ngọt đến chua. Vỏ cam dày, có vị đắng, có thể chế biến thành thức ăn cho súc vật bằng cách rút nước bằng sức ép và hơi nóng. Nó cũng được dùng làm gia vị hay đồ trang trí trong một số món ăn.



```

```

81

81

Hình ảnh (tt)

• Sử dụng hình ảnh như một siêu liên kết:

```
<a href="../../index.html" title="Trở về trang chủ" >
  
</a>
```

82

82

Hình ảnh (tt)

- **Sử dụng hình ảnh làm ảnh nền:**

- Hình ảnh có thể được sử dụng làm nền cho toàn bộ trang web hoặc cho một phần tử trên trang.
- Sử dụng thuộc tính background.
- Ví dụ:

```
<body background=" ../Hinh/bg1.jpg">
  <h1>Sử dụng ảnh nền</h1>
</body>
```



83

83

Hình ảnh (tt)

- **ImageMap:**

- Tạo các vùng liên kết trên ảnh (hotspots), nếu click chuột lên sẽ được chuyển đến URL chỉ định trong liên kết.
- Để tạo một ImageMap, ta sử dụng các thẻ sau:
 - Thẻ : dùng để nhúng ảnh vào trang web
 - Thẻ <map>: định nghĩa một bản đồ ảnh
 - Thẻ <area>: định nghĩa vị trí các vùng liên kết

84

84

Hình ảnh (tt)

• ImageMap (tt):

– Các thuộc tính của thẻ <area>:

- shape: định nghĩa hình dạng của vùng với ba giá trị:
 - circ hoặc circle (hình tròn)
 - rect hoặc rectangle (hình chữ nhật)
 - poly hoặc polygon (hình đa giác)
- coords: tọa độ trên hình.
 - circle: tọa độ (x,y) của tâm và bán kính (px)
 - rect: tọa độ (x,y) của 2 điểm góc trên trái và góc dưới phải
 - poly: tập các điểm của đa giác.

85

85

Hình ảnh (tt)

• ImageMap (tt): ví dụ

```

<map name="animals">
  <area shape="circle" coords="100,110, 20"
    href="rabits.html"
    target="_blank" title="Giới thiệu về loài thỏ">
  <area shape="poly"
    coords="32,20,90,15,100,60,75,105,60,150,30,160,32,20"
    href="cats.html"
    target="_blank" title="Giới thiệu về loài mèo">
  <area shape="rect" coords="190,46,290,150"
    href="dogs.html"
    target="_blank" title="Giới thiệu về các giống chó kiểng">
</map>
```



86

86

Audio

- Có nhiều cách để nhúng một tập tin audio/video vào trang web.
- HTML5: sử dụng thẻ `<audio></audio>`
- Hỗ trợ các định dạng:
 - MP3 (audio/mpeg)
 - OGG (audio/ogg)
 - WAV (audio/wav)

87

87

Audio

- Cách thực hiện:
 - Mở đầu và kết thúc với cặp thẻ `<audio></audio>`
 - Có thể thiết lập các thuộc tính luận lý trong thẻ `<audio>`
 - controls: hiển thị bảng điều khiển
 - loop: lặp lại
 - Trong thẻ `<audio>` có thể có nhiều cặp thẻ `<source></source>`, mỗi thẻ có thuộc tính src chỉ định đường dẫn đến tập tin audio. Trình duyệt sẽ chọn tập tin âm thanh hợp lệ theo thứ tự từ trên xuống để phát.
 - Có thể bổ sung chuỗi thông báo trong trường hợp các trình duyệt cũ không hỗ trợ HTML5 hoặc định dạng file audio không phù hợp.

88

88

Audio

- Ví dụ:

```
<h2> Chương trình nghe nhạc </h2>
```

```
<audio controls loop >
```

```
  <source src = "../video-audio/audio.mp3" type = "audio/mpeg">
```

```
  <source src = "../video-audio/audio.ogg" type = "audio/ogg">
```

Trình duyệt của bạn không hỗ trợ HTML5 Audio, vui lòng truy cập link:

```
<a href = "../video-audio/audio.mp3">audio.mp3</a>
```

```
</audio>
```

Chương trình nghe nhạc

▶ 0:09 / 5:10 ————— 🔊

89

89

Video

- Sử dụng thẻ <video> </video>
- Hỗ trợ các định dạng:
 - MP4 (video/mp4)
 - WebM (video/webm)
 - Ogg (video/ogg)

90

90

Video

• Ví dụ

```
<h2 align="center">CHƯƠNG TRÌNH XEM VIDEO</h2>
<video loop controls width="320" height="240">
  <source src="../../video-audio/small.mp4" type=video/mp4>
  <source src="../../video-audio/small.ogv" type=video/ogv>
  <source src="../../video-audio/small.webm" type=video/webm>
  <source src="../../video-audio/small.flv" type=video/flv>
  <source src="../../video-audio/small.3gp" type=video/3gp>
  Nếu trình duyệt của bạn xem được, vui lòng truy cập link:
  <a href= "http://techslides.com/sample-webm-ogg-and-mp4-video-files-for-html5">
techslides </a>
</video>
```

CHƯƠNG TRÌNH XEM VIDEO



91

91

NỘI DUNG CHƯƠNG 2

1. Giới thiệu HTML
2. Các thẻ HTML cơ bản
3. Siêu liên kết
4. Danh sách
5. Bảng
6. Multimedia
7. Form và Control
8. Giới thiệu HTML5

Ứng dụng Web

92

92

2.7 Form và controls

- **Form - biểu mẫu:**

- Là một thành phần trong tài liệu HTML chứa các phần tử đặc biệt gọi là các phần tử điều khiển (controls).
- Cung cấp một giao diện cho phép người dùng tương tác với ứng dụng
 - Cung cấp công cụ tìm kiếm cho website.
 - Thu thập thông tin của người dùng như tên, địa chỉ, số điện thoại, địa chỉ email,... khi đăng ký sử dụng dịch vụ.
 - Thu thập thông tin đăng ký mua bán hàng hoá.
 - Thu thập thông tin phản hồi từ người dùng.
 - ...

93

93

Form (tt)

Create your Google Account

First name Last name

Username @gmail.com

You can use letters, numbers & periods

[Use my current email address instead](#)

Password Confirm password

Use 8 or more characters with a mix of letters, numbers & symbols

[Sign in instead](#)

NEXT



One account. All of Google working for you.

94

94

Form (tt)

- HTML có thể tạo form, nhưng để làm cho form hoạt động cần phải có một ngôn ngữ lập trình như JavaScript.
- Form là phần tử HTML có thể chứa các điều khiển như:
 - Hộp văn bản (textbox)
 - Hộp kiểm (checkbox)
 - Các nút lệnh.
 - ...
- Dữ liệu do người dùng nhập từ form → gửi đến máy chủ xử lý → gửi trả về mã HTML → trình duyệt hiển thị

95

95

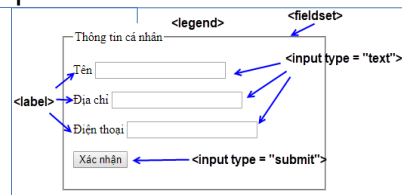
Form (tt)

- Tạo form:
 - Sử dụng thẻ `<form>` `</form>`
 - Bên trong thẻ `<form>` `</form>` là các phần tử điều khiển.

```

<form action = "">
<fieldset>
<legend>Thông tin cá nhân</legend>
<p><label>Tên</label>
  <input type = "text" /> </p>
<p><label>Địa chỉ</label>
  <input type = "text" /> </p>
<p><label>Điện thoại</label>
  <input type = "text" /> </p>
<p><input type = "submit" value="Xác nhận" /> </p>
</fieldset>
</form>

```



96

96

Form (tt)

- **Một số thuộc tính cơ bản của form:**

- **id, name:** định danh/ tên của form, được sử dụng để truy xuất đến form.
- **action:** quy định một hành động với dữ liệu khi biểu mẫu được gửi, giá trị là URL hoặc đoạn mã script

```
< form action=http://www.example.org/login.aspx method='get' >
<input type = "submit" value="Đăng nhập" />
</form>

<form action = "javascript:window.open('form-basic.html')">
<input type = "submit" value="Xác nhận" />
</form>
```

97

97

Form (tt)

- **method:** phương thức gửi dữ liệu
 - **get:** mặc định
 - Dữ liệu trên form được gửi nằm trong một phần của URL (dạng **?name=value&name=value...**)
 - Dạng chuỗi ký tự, có giới hạn (khoảng 1024 ký tự)
 - Có thể cached, bookmark
 - Ví dụ: **<form action="TopicList.aspx" method="get">**
 - Khi chuyển đến trang TopicList.aspx, URL hiển thị trên trình duyệt có dạng: <http://cunghoclaptrinh.com/TopicList.aspx?subjectid=1>
 - **post:** bảo mật hơn get
 - Dữ liệu được gửi thông qua HTTP header
 - Có thể truyền dữ liệu nhị phân, kích thước lớn
 - Không thể cache, bookmark
 - Ví dụ: **<form id = 'frmLogin' action = "login.php" method="post">**

98

98

Form (tt)

- **onsubmit**: sự kiện xảy ra khi form được submit, thường là một hàm/đoạn mã script được thực thi khi dữ liệu trên form được gửi đi (người dùng click lên nút submit của form). Ví dụ:
`<form action="" method="post" onsubmit="Validate();" >`
- **onreset**: là đoạn mã lệnh được thực thi khi dữ liệu trên form được trả lại trạng thái ban đầu (người dùng click lên nút reset của form).
- **autocomplete = on/off**: Gợi ý tự động khi nhập liệu

99

99

Controls

- Là các thành phần trên form được dùng để trình bày dữ liệu và cho phép người dùng tương tác với ứng dụng web
 - Text boxes: được dùng để nhập một dòng văn bản.
 - Password boxes: được dùng để nhập mật khẩu.
 - Text areas: được dùng để nhập văn bản nhiều dòng.
 - Select lists: cung cấp một danh sách cho phép người dùng lựa chọn.
 - Check boxes: cung cấp danh sách các hộp kiểm cho phép người dùng có thể chọn nhiều tùy chọn.
 - Radio buttons: cung cấp danh sách các tùy chọn mà tại một thời điểm chỉ cho phép chọn một.
 - Buttons: các nút lệnh cho phép thực thi một tác vụ trên form.
 - Labels: hiển thị văn bản tĩnh, thường dùng để gán nhãn chú thích cho các phần tử khác trên form.
 - Fieldsets và legends: cung cấp một khung chứa các phần tử khác trên form.

100

100

Controls (tt)

- Một số thuộc tính chung của các control:
 - **type**: thuộc tính quy định loại control.
 - **id**: định danh của control.
 - **name**: tên của control.
 - **value**: chuỗi hiển thị trên control.

101

101

Các phần tử nhập văn bản

- **<input type="text">**: ô nhập 1 dòng văn bản, có các thuộc tính:
 - **size**: quy định chiều rộng của ô, mặc định là 20 ký tự.
 - **maxlength**: quy định số ký tự tối đa có thể nhập.
 - **readonly**: chỉ đọc, không cho phép nhập.
 - **placeholder**: chuỗi gợi ý.

102

102

Các phần tử nhập văn bản (tt)

- **<input type="text">**, ví dụ

```
<form>
Họ:
<br><input type="text" name="txtHo" placeholder="Họ và chữ lót">
<br>Tên:
<br><input type="text" name="txtTen" maxlength="10">
<br>Địa chỉ:
<br><input type="text" name="txtDiachi" size="50">
</form>
```

Họ:
Nguyễn Văn
Tên:
Thanh
Địa chỉ:
371, Nguyễn Kiệm, Phường 3, TP HCM

103

103

Các phần tử nhập văn bản (tt)

- **<input type="password">**: tạo ô nhập mật khẩu

```
<form>
Tên đăng nhập:
<br><input type="text" name="txtTendn">
<br>Mật khẩu:
<br><input type="password" name="txtMatkhau">
</form>
```

Tên đăng nhập:

Mật khẩu:

104

104

Các phần tử nhập văn bản (tt)

- **Phần tử <textarea>:**

- Hộp văn bản cho phép nhập trên nhiều dòng.
- Văn bản bên trong phần tử <textarea> được xử lý tương tự như văn bản nằm bên trong phần tử <pre>.
- Các thuộc tính:
 - rows: chiều cao của hộp văn bản tính theo số dòng.
 - cols: độ rộng của hộp văn bản, tính theo độ rộng trung bình của ký tự.

105

105

Các phần tử nhập văn bản (tt)

- **Phần tử <textarea>, ví dụ:**

```
<form>
  Vui lòng nhập ý kiến của bạn:
  <br><textarea rows="5" cols="50"></textarea>
</form>
```

Vui lòng nhập ý kiến của bạn:

Dòng thứ nhất
Dòng thứ hai
Dòng thứ ba
Dòng thứ tư
Dòng thứ năm

106

106

Các nút lệnh (buttons)

- Là các nút nhấn trên trang web, khi người dùng click vào sẽ thực thi một tác vụ nào đó
- Một số thuộc tính:
 - onclick: chuỗi mã lệnh hoặc hàm xử lý khi click lên phần tử.
 - disabled: nếu được chỉ định, phần tử sẽ bị vô hiệu hóa
 - accesskey: chỉ định phím tắt (shortcut key) thay vì dùng chuột click lên button.

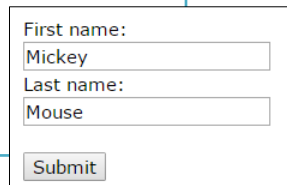
107

107

Các nút lệnh (tt)

- **<input type="submit">**: tạo submit button, gửi thông tin đến web server

```
<form action="action_page.php">
  First name:<br>
  <input type="text" name="firstname" value="Mickey"><br>
  Last name:<br>
  <input type="text" name="lastname" value="Mouse">
  <br><br>
  <input type="submit" value="Submit">
</form>
```



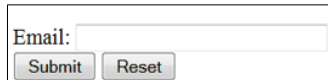
108

108

Các nút lệnh (tt)

- **<input type="reset">**: hủy giá trị nhập, trả trang web về trạng thái ban đầu

```
<form action="demo_form.asp">
  Email: <input type="text" name="CustEmail"> <br>
  <input type="submit" value="Submit">
  <input type="reset" />
</form>
```



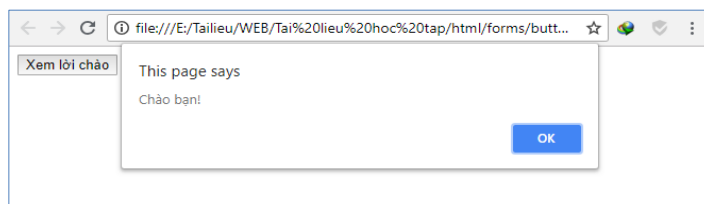
109

109

Các nút lệnh (tt)

- **<input type="button">**: tạo nút dùng để xử lý các script

```
<form>
  <input type="button" value="Xem lời chào" onClick="alert('Chào bạn!')">
</form>">
```



110

110

Các nút lệnh (tt)

- **<input type="image">**:
 - Tạo một nút submit hiển thị hình ảnh
 - Hai thuộc tính thường được sử dụng:
 - src: địa chỉ url của tập tin hình ảnh.
 - alt: chuỗi thay thế khi đường dẫn tập tin hình ảnh không đúng.
 - Khi được nhấn, tọa độ của vị trí nhấn chuột được chuyển đến máy chủ dưới dạng
 - name.x = value_of_x
 - name.y = value_of_y
 - Trong đó: name là tên của nút ảnh

111


111

Các nút lệnh (tt)

- **<input type="image">**: ví dụ

```
<form>
  Vui lòng nhập ý kiến của bạn:
  <br><textarea rows="5" cols="50"></textarea>
  <br>
  <input type="image" src="sendmail.png" alt="Gửi mail"
    width="100px" height="40px" onClick="SendMail()">
</form>
```

Vui lòng nhập ý kiến của bạn:

 Send

112

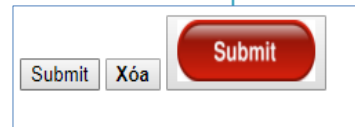
112

Các nút lệnh (tt)

- **Phần tử <button> </button>:**

- Tạo nút với các chức năng xử lý cũng tương tự phần tử <input>.
- Ưu điểm: nội dung nằm giữa <button> và </button> có thể chứa hình ảnh, văn bản và có thể định dạng linh hoạt hơn.

```
<form>
  <button type="submit"> Submit </button>
  <button type="reset"><b>Xóa</b> </button>
  <button type="button">
    
  </button>
</form>
```



113

113

Check boxes

- **<input type="checkbox">:**

- Tạo các ô cho phép chọn bằng cách đánh dấu (check)
- Có thể chọn nhiều giá trị cùng một lúc
- Một số thuộc tính :
 - **name:** tên, các checkbox cùng nhóm có cùng giá trị cho thuộc tính này.
 - **value:** giá trị được gửi đến server nếu checkbox được chọn.
 - **checked:** đây là thuộc tính luận lý, nếu được chỉ định, checkbox sẽ được đánh dấu khi nạp trang web.

114

114

Check boxes (tt)

- **<input type="checkbox">**: ví dụ

```
<form action="http://www.abc.com/cv.aspx" method="get" name="frmCV">
  Bạn thành thạo các kỹ năng nào sau đây?
  <br><input type="checkbox" name="chkSkills" value="html" checked/> HTML
  <br><input type="checkbox" name="chkSkills" value="xhtml"/> XHTML
  <br><input type="checkbox" name="chkSkills" value="css"/> CSS
</form>
```

Bạn thành thạo các kỹ năng nào sau đây?

- ☒ HTML
☐ XHTML
☐ CSS

115

115

Radio buttons

- **<input type="radio">**: tương tự checkbox, nhưng tại một thời điểm chỉ có thể chọn một radio trong nhóm.

```
<form>
  <input type="radio" name="sex" value="male" checked> Male
  <br>
  <input type="radio" name="sex" value="female"> Female
</form>
```

- ☒ Male
☐ Female

116

116

Danh sách

- Có hai loại danh sách:

Danh sách xổ xuống
(dropdown list)

Vui lòng chọn một màu:

Xanh lục	▼
Đỏ	
Xanh lục	
Xanh dương	

Danh sách hiển thị
nhiều phần tử (listbox)

Chọn các ngày trong tuần:

Thứ hai	▲
Thứ ba	
Thứ tư	
Thứ năm	▼

117

117

Danh sách (tt)

- Phần tử **<select>**: tạo một hộp danh sách các lựa chọn
 - Mỗi phần tử trong danh sách được hiển thị bởi phần tử **<option>**.
 - Các thuộc tính:
 - **name**: Tên của phần tử select, khi form được gửi đi, thuộc tính này được gán với giá trị của phần tử option (thuộc tính value)
 - **size**: số dòng được hiển thị, dùng khi tạo một listbox
 - **multiple**: cho phép chọn một hay nhiều chọn lựa

118

118

Danh sách (tt)

- Phần tử **<option>**: tạo một mục chọn trong danh sách
 - Các thuộc tính:
 - **value** : giá trị chọn lựa
 - **selected** : giá trị được chọn sẵn
 - **disable** : không cho phép lựa chọn

119

119

Danh sách (tt)

- Ví dụ tạo dropdown list:

```
<form action="" method="post">
  Vui lòng chọn một màu:<br>
  <select name="selColor" >
    <option value="red" > Đỏ </option >
    <option value="green" selected > Xanh lục </option>
    <option value="blue"> Xanh dương </option>
  </select >
</form>
```

Vui lòng chọn một màu:

Xanh lục	▼
Đỏ	
Xanh lục	
Xanh dương	

120

120

Danh sách (tt)

- Tạo listbox: bổ sung thuộc tính size

```
<form action="" name="frmDays" method="post">
  Chọn các ngày trong tuần:<br><br>
  <select size="4" name="selDay" multiple>
    <option value="Mon">Thứ hai</option>
    <option value="Tue">Thứ ba</option>
    <option value="Wed">Thứ tư</option>
    <option value="Thu">Thứ năm</option>
    <option value="Fri">Thứ sáu</option>
    <option value="Sat">Thứ bảy</option>
    <option value="Sun">Chủ nhật</option>
  </select>
  <br><br><input type="submit" value="Xác nhận" />
</form>
```

Chọn các ngày trong tuần:

Thứ hai
Thứ ba
Thứ tư
Thứ năm

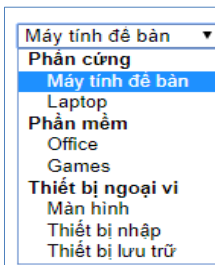
Xác nhận

121

121

Danh sách (tt)

- Phần tử **<optgroup>**: đặt trong phần tử **<select>** để tạo danh sách có phân nhóm



```
<select name="selProducts">
  <optgroup label="Phần cứng">
    <option value="Desktop">Máy tính để bàn</option>
    <option value="Laptop">Laptop</option>
  </optgroup>
  <optgroup label="Phần mềm">
    <option value="OfficeSoftware">Office</option>
    <option value="Games">Games</option>
  </optgroup>
  <optgroup label="Thiết bị ngoại vi">
    <option value="Monitors">Màn hình</option>
    <option value="InputDevices">Thiết bị nhập</option>
    <option value="Storage">Thiết bị lưu trữ</option>
  </optgroup>
</select>
```

122

122

File upload

- **<input type="file">**: cho phép chọn một tập tin để upload lên server.
 - Thuộc tính **accept**: chỉ định loại tập tin cho phép chọn

```
<form action="http://www.abc.com/upload.asp" method="post"
  name="formUpload" enctype="multipart/form-data">
  Chọn file upload:<br>
  <input type="file" name="fileUpload" accept="image/*" />
  <br><input type="submit" value="Tải lên" />
</form>
```

Chọn file upload:

Choose File photo 3.JPG

Tải lên

123

123

Phần tử label

- **<label></label>**:
 - Gắn nhãn tĩnh cho các control trên form.
 - Thiết lập thuộc tính **for** của label = **id** của control

```
<label for="Uname">Tên đăng nhập</label>
<input type="text" id="Uname" name="txtUserName"/>
<label for="Pwd">Mật khẩu</label>
<input type="password" id="Pwd" name="txtPassword"/>
```

Tên đăng nhập

Mật khẩu

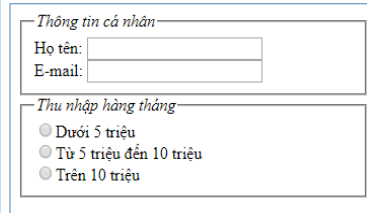
124

124

Phần tử Fieldset và Legend

- **<fieldset>**: tạo một đường viền xung quanh một nhóm các control
- **<legend>**: tạo tiêu đề chú thích cho phần tử <fieldset>

```
<fieldset>
  <legend><em>Thông tin cá nhân</em></legend>
  <!-- các control đặt nơi đây -->
</fieldset>
<fieldset>
  <legend><em>Thu nhập hàng tháng</em></legend>
  <!-- các control đặt nơi đây -->
</fieldset>
```



125

125

Thao tác trên Form

- Xếp thứ tự các phần tử trên form theo phím tab
 - Thường áp dụng cho các phần tử có thể nhận tiêu điểm như Textboxes, Buttons, RadioButton, Checkbox, danh sách,...
 - Sử dụng thuộc tính **tabindex**: giá trị nguyên quy định thứ tự các control khi nhấn phím Tab, Shift + Tab.
 - Khi form được nạp, phần tử có giá trị tabindex thấp nhất sẽ được nhận tiêu điểm (focus)
 - Phần tử không gán thiết lập giá trị tabindex sẽ nhận tiêu điểm cuối cùng

```
<input tabindex=1 type =“text”/>
<input tabindex=2 type =“password”/>
...
<input tabindex=5 type =“submit” value=“Log in” />
```

126

126

Thao tác trên Form (tt)

- Vô hiệu hóa phần tử
 - **disabled**: không cho truy cập. Khi form được submit, giá trị của phần tử có thiết lập thuộc tính disabled sẽ không được gửi đến máy chủ
- **readonly**: ngăn người dùng thay đổi giá trị của phần tử bằng thao tác trực quan

```
<input name="realname" value="Hi There" disabled>
<input type="submit" value="GO">
```

```
<input name="realname" value="Hi There" readonly>
<input type="submit" value="GO">
```

127

127

Thao tác trên Form (tt)

- **Thiết lập phím tắt để truy cập nhanh phần tử**
 - Một số phần trên form có thuộc tính **accesskey** cho phép gán ký tự bàn phím, dùng kết hợp với phím **Alt** (hoặc **Alt + Shift**, tùy theo trình duyệt)
 - Khi nhấn phím tắt, control sẽ nhận tiêu điểm
 - Ví dụ:

```
<input accesskey = "U" tabindex=2 type="text" id="username"/>
```

128

128

NỘI DUNG CHƯƠNG 2

1. Giới thiệu HTML
2. Các thẻ HTML cơ bản
3. Siêu liên kết
4. Danh sách
5. Bảng
6. Multimedia
7. Form và Control
8. Giới thiệu HTML5

Ứng dụng Web

129

129

2.8. Giới thiệu HTML5

- HTML5 là phiên bản mới của HTML, được bắt đầu nghiên cứu vào tháng 6/2004.
- Ưu điểm:
 - Cho phép ứng dụng web có thể chạy được trên nhiều trình duyệt khác nhau của bất kỳ thiết bị nào mà không cần cài đặt thêm plugin.
 - Tốc độ truy cập nhanh.
 - Bổ sung nhiều thẻ mới, cung cấp giao diện trực quan, tăng tính tương tác.
 - Hỗ trợ CSS3.

130

130

2.8. Giới thiệu HTML5

- Ưu điểm (tt):
 - Hỗ trợ API (Application Programming Interface) và DOM (Document Object Model), cung cấp các API xử lý cơ sở dữ liệu với SQLite.
 - Bổ sung thẻ <canvas>, <svg> cho phép vẽ/hiển thị các đối tượng đồ họa, mà không cần Flash hay các plug-in.
 - Hiển thị video, audio đơn giản hơn, không cần cài đặt các công cụ phát triển.
 - Các phần tử input mới với chức năng kiểm chứng tự động, làm giảm nhu cầu tải mã javascript, giúp truyền thông tốt hơn giữa các thiết bị di động và các máy chủ điện toán đám mây.

131

131

Giới thiệu HTML5 (tt)

- Các tính năng mới trong HTML5
 - Khai báo <!DOCTYPE html>
 - Khai báo Character Set đơn giản hơn: <meta charset="UTF-8">
- Không còn sử dụng một số thẻ và thuộc tính định dạng cũ vì có thể thay bằng mã CSS: <basefont>, <big>, <center>, , <s>, <strike>, <tt>, <u>
- Bổ sung các thành phần mới như thẻ <header>, <footer>, <article>, <section>, <nav> hỗ trợ thiết kế bố cục trang web dễ dàng hơn

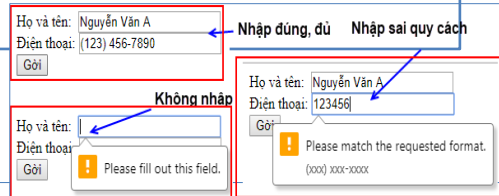
132

132

Giới thiệu HTML5 (tt)

- **<input type="text">** với thuộc tính **required**, **pattern**: tạo ô bắt buộc nhập dữ liệu đúng cách.

```
<label for="ten" >Họ và tên:</label>
<input type="text" name="ten" id="ten" required>
<br><label for="dienthoai" >Số điện thoại:</label>
<input type="text" name="dienthoai" id="dienthoai" required pattern="^\d{3}\d{3}-\d{4}$" title="(xxx) xxx-xxxx">
<br><input type="submit" value="Gửi">
```



Three screenshots of a form demonstrating HTML5 input validation:

- Top:** A valid form with "Họ và tên: Nguyễn Văn A" and "Điện thoại: (123) 456-7890". A blue arrow points to the phone number field with the label "Nhập đúng, đủ" (Entered correctly, fully).
- Bottom Left:** An empty "Họ và tên:" field with a red border and a yellow error message: "Please fill out this field." A blue arrow points to the field with the label "Không nhập" (Not entered).
- Bottom Right:** A form with "Họ và tên: Nguyễn Văn A" and "Điện thoại: 123456". A red border and a yellow error message: "Please match the requested format. (xxx) xxx-xxxx". A blue arrow points to the phone number field with the label "Nhập sai quy cách" (Entered incorrectly).

133

Giới thiệu HTML5 (tt)

- **<input type="number">**: tạo ô chỉ nhập số

```
<form>
  Chọn một giá trị:
  <input type="number" name="txtNumber" min="0" max="100"
    step="10" value="30">
</form>
```

Chọn một giá trị:

134

134

Giới thiệu HTML5 (tt)

- **<input type="date">**: phần tử cho phép nhập ngày

```
<form>
  Chọn một ngày trước ngày 1/1/2018:<br>
  <input type="date" name="day" max="2018-01-01">
<br>
  Chọn một ngày sau ngày 31/12/2010:<br>
  <input type="date" name="day" min="2010-12-31">
<br>
  <input type="submit" value="Xác nhận">
</form>
```

Chọn một ngày trước ngày 1/1/2018:
 12/11/2017
 Chọn một ngày sau ngày 31/12/2010:
 08/10/2017
 Xác nhận

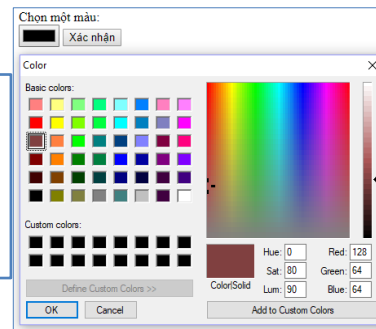
135

135

Giới thiệu HTML5 (tt)

- **<input type="color">**: phần tử cho phép chọn màu

```
<form>
  Chọn một màu:<br>
  <input type="color" name="mycolor">
  <input type="submit" value="Xác nhận">
</form>
```



136

136

Giới thiệu HTML5 (tt)

- **<input type="range">**: phần tử hiển thị một thanh trượt, cho phép chọn giá trị trong một vùng giá trị định trước

```
<form>
Số điểm của bạn:
<input type="range" name="diem" min="0" max="10" value="8">
<input type="submit" value="Xác nhận">
</form>
```

Số điểm của bạn: 

137

137

Giới thiệu HTML5 (tt)

- **<input type="month">**: phần tử hiển thị một quyền lịch cho phép chọn tháng, năm

```
<form>
Sinh nhật của bạn (tháng, năm):<br>
<input type="month" name="ngaysinh" >
<input type="submit" value="Xác nhận">
</form>
```

Sinh nhật của bạn (tháng, năm):

May 2000

May 2000

Sun	Mon	Tue	Wed	Thu	Fri	Sat
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3

138


138

Giới thiệu HTML5 (tt)

- **<input type="email">**: phần tử cho phép nhập và kiểm tra tính hợp lệ của email

```
<form action="xulymail.php">
  E-mail:
  <input type="email" name="email">
  <input type="submit" value="Xác nhận">
</form>
```

E-mail:

 Please include an '@' in the email address. 'trangouit' is missing an '@'.

139


139

Giới thiệu HTML5 (tt)

- **<input type="url">**: phần tử cho phép nhập và kiểm tra tính hợp lệ của địa chỉ url

```
<form >
  Web site:
  <input type="url" name="homepage">
  <input type="submit" value="Submit">
</form>
```

Web site:

 Please enter a URL.

140

140

Giới thiệu HTML5 (tt)

- **<datalist>**: kết hợp với phần tử **<input type="text">** để tạo một danh sách tùy chọn cho phép nhập với khả năng tự điền dữ liệu (autocomplete).

```
<p>Các môn học yêu thích của bạn: <br>
<input type="text" list="favorite"></p>
<datalist id="favorite" >
  <option> Cơ sở lập trình </option>
  <option> Lập trình web </option>
  <option> Cơ sở dữ liệu </option>
  <option> Lập trình giao diện </option>
</datalist>
```

Các môn học yêu thích của bạn:

- Cơ sở lập trình
- Lập trình web
- Cơ sở dữ liệu
- Lập trình giao diện

141

141

Giới thiệu HTML5 (tt)

- **<canvas>**:
 - Là phần tử được dùng kết hợp với JavaScript để vẽ các đối tượng đồ họa trên trang web.
 - Canvas: chứa đối tượng đồ họa
 - JavaScript: thực hiện thao tác vẽ
 - Phần tử Canvas cung cấp một đối tượng thiết bị ngữ cảnh và các phương thức để vẽ các đối tượng trong không gian hai chiều như các đường, hình chữ nhật, hình tròn, bitmap ...



142

142

Giới thiệu HTML5 (tt)

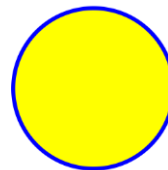
- **<svg>**:
 - SVG (Scalable Vector Graphics): một định dạng hình ảnh theo cấu trúc XML để hiển thị hình ảnh dưới dạng vector.
 - Phần tử <svg> được dùng để chứa các đối tượng đồ họa như đường thẳng, hình hộp, hình tròn, văn bản, hình ảnh,... bằng cách khai báo các thuộc tính của đối tượng theo XML
 - Các thuộc tính thường được sử dụng để vẽ:
 - Width, height, viewBox: kích thước của svg
 - stroke: màu nét vẽ
 - stroke-width: độ dày nét vẽ
 - fill: màu nền
 - style

143

143

Giới thiệu HTML5 (tt)

- **<svg>**: đường tròn
 - Đặt thẻ <circle> bên trong thẻ <svg> với các thuộc tính
 - cx, cy: vị trí tọa độ tâm đường tròn
 - r: bán kính



```
<svg width="300" height="300">
  <circle cx="200" cy="100" r="80" stroke="blue" stroke-width="4" fill="yellow" />
</svg>
```

144

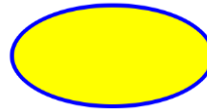
144

Giới thiệu HTML5 (tt)

- **<svg>**: ellipse

– Đặt thẻ **<ellipse>** bên trong thẻ **<svg>** với các thuộc tính

- cx, cy: vị trí tọa độ tâm đường tròn
- rx: bán kính chiều ngang
- ry: bán kính chiều dọc



```
<svg width="300" height="300">
  <ellipse cx="150" cy="70" rx="100" ry="50" stroke="blue" stroke-width="4" fill="yellow" />
</svg>
```

145

145

Giới thiệu HTML5 (tt)

- **<svg>**: hình chữ nhật

– Đặt thẻ **<rect>** bên trong thẻ **<svg>** với các thuộc tính

- x, y: vị trí tọa độ điểm góc trên trái
- width: chiều ngang
- height: chiều cao



```
<svg width="300" height="200">
  <rect x="50" y="50" width="200" height="100" style="fill:orange; stroke:black; stroke-width:3;" />
</svg>
```

146

146

Giới thiệu HTML5 (tt)

- **<svg>**: đoạn thẳng

– Đặt thẻ **<line>** bên trong thẻ **<svg>** với các thuộc tính

- x1, y1, x2, y2: vị trí hai điểm



```
<svg width="300" height="200">
  <line x1="50" y1="50" x2="250" y2="150" style="stroke:red; stroke-width:3;" />
</svg>
```

147

147

Giới thiệu HTML5 (tt)

- **<svg>**: đa giác

– Đặt thẻ **<polygon>** bên trong thẻ **<svg>** với các thuộc tính

- points: danh sách các điểm
x1, y1, x2, y2,...



```
<svg viewBox="0 0 200 100">
  <polygon points="0,100 50,25 50,75 100,0" />
  <polygon points="100,100 150,25 150,75 200,0" fill="none" stroke="black" />
</svg>
```

148

148

- **<svg>**: ví dụ

```

<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" />
</svg>
<svg width="400" height="180">
  <rect x="50" y="20" rx="20" ry="20" width="150" height="150"
  style="fill:red;stroke:black;stroke-width:5;opacity:0.5" />
</svg>
<svg width="300" height="200">
  <polygon points="100,10 40,198 190,78 10,78 160,198"
  style="fill:lime;stroke:purple;stroke-width:5;fill-rule:evenodd;" />
</svg>
  
```



149