

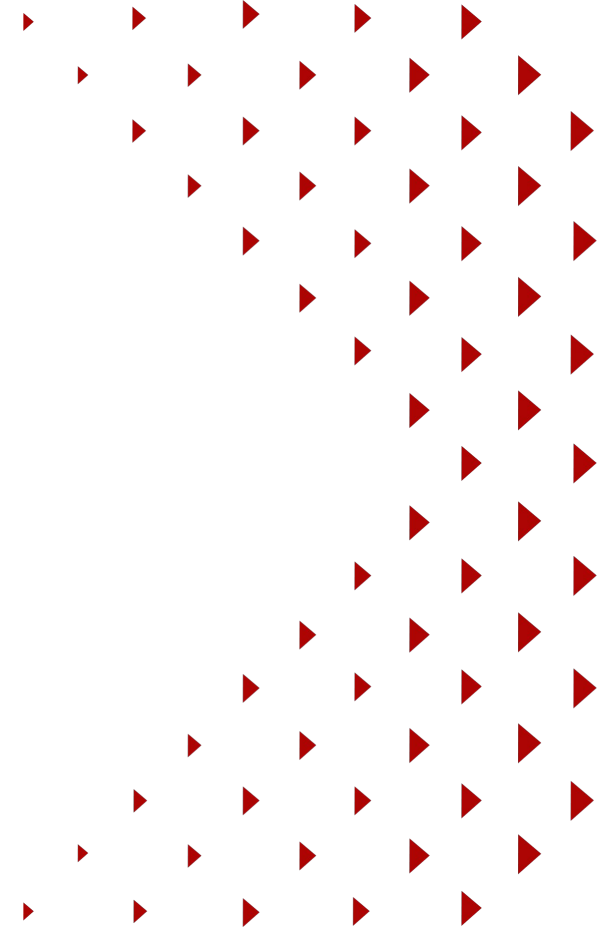


SESSION 03:

CSS BASIC

Module 1: Web Application UI Design

Version: 1.0



Mục tiêu

- Nhúng được CSS vào trang web
- Sử dụng được CSS selector
- Sử dụng được CSS để tùy chỉnh văn bản
- Giải thích được Box Model
- Trình bày được mô hình hộp (box model) trong CSS
- Sử dụng được CSS để điều chỉnh kích thước các thành phần
- Sử dụng được CSS tạo layout đơn giản

1. **CSS - Cascading Style Sheets**
2. **Cú pháp sử dụng CSS**
3. **Nhúng CSS vào trang web**
4. **Các loại bộ chọn (selector)**
5. **Các thuộc tính định dạng văn bản**
6. **Mô hình hộp trong CSS**

CSS là gì ?

- CSS là viết tắt của cụm từ Cascading Style Sheets
- Ngôn ngữ định kiểu, đưa ra các chỉ thị để miêu tả cách trình bày nội dung trong trang HTML
- Trang trí, định dạng, bố cục cho trang web
- File có phần mở rộng là .css với nhiệm vụ tách riêng phần định dạng (style) ra khỏi nội dung HTML
- Sử dụng CSS giúp dễ dàng quản lý nội dung trang HTML, điều khiển định dạng, tốn ít thời gian khi code hay chỉnh sửa



ĐẶC ĐIỂM CỦA CSS

- Tái sử dụng mã (Code): CSS tiết kiệm thời gian bằng việc viết chỉ thị định dạng một lần và áp dụng cho nhiều trang
- Giảm mã HTML: CSS giúp giảm kích thước mã trong file HTML do các chỉ thị định dạng được đặt trong file khác
- Độc lập thiết bị: CSS được thiết kế để cung cấp cho người đọc web có cái nhìn giống nhau khi trang được hiển thị trên nhiều thiết bị khác nhau
- CSS có thể được nhúng cùng với mã HTML hoặc link giữa file HTML với file CSS

CÁCH SỬ DỤNG CSS

Có 3 cách sử dụng CSS trong trang web

- Khai báo trực tiếp ở thẻ HTML (Inline style)

```
<h1 style="color: red; font-size: 100px">Learning IT is not difficult</h1>
```

- Khai báo trong thẻ <style> của tài liệu HTML (Internal style sheet)

```
<style>
  h1 {
    color: yellow;
    font-size: "100px";
  }
</style>
```

- Khai báo trong file .css riêng biệt (External style sheet)

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

CÁCH SỬ DỤNG CSS

Độ ưu tiên khi sử dụng (nhúng) CSS trong trang web:

Khi có nhiều khai báo CSS cho cùng một thẻ HTML thì các khai báo này sẽ tuân thủ theo độ ưu tiên như sau:

1. Inline style
2. External và Internal Style
3. Kiểu mặc định của trình duyệt

4. Các loại bộ chọn (selector)

Hiểu đơn giản CSS Selector là thứ cho phép bạn nhắm mục tiêu tới các phần tử HTML để áp dụng các thuộc tính CSS cho chúng.

CSS Selector giống như là đường dẫn, chỉ định để cho CSS biết bạn đang muốn điều chỉnh, tạo kiểu cho phần tử HTML nào vậy.

- **Tag selector**: Chọn tất cả các phần tử
- **Id selector**: Chọn phần tử với id xác định
- **Class selector**: Chọn tất cả các phần tử xác định



4. Các loại bộ chọn (selector)

Bộ chọn	Ví dụ	Mô tả
*	*	Chọn tất cả thành phần
tag	p	Chọn tất cả các thành phần cùng loại.
tag01 tag02	div p	Chọn tất cả thành phần <tag02> bên trong thành phần <tag01>
tag01 , tag02	div , p	Chọn tất cả thành phần <tag01> và <tag02>.
tag01 + tag02	div + p	Chọn tất cả thành phần <tag02> được đặt kế và sau thành phần <tag01>.
tag01 > tag02	div > p	Chọn tất cả thành phần <tag02> có thành phần cha là <tag01>.
.class	.className	Chọn tất cả các thành phần có cùng tên class.
#id	#idName	Chọn tất cả các thành phần có cùng tên id.
:active	a:active	Chọn các liên kết được kích hoạt
:after	p:after	Thêm nội dung ngay phía sau thành phần.
:before	p:before	Thêm nội dung ngay phía trước thành phần.

4. Các loại bộ chọn (selector)

Bộ chọn	Ví dụ	Mô tả
:first-child	p:first-child	Chọn thành phần đầu tiên của thành phần cha chứa nó.
:first-letter	p:first-letter	Chọn ký tự đầu tiên của một thành phần.
:first-line	p:first-line	Chọn dòng đầu tiên của một thành phần.
:focus	input:focus	Thành phần sẽ focus khi được chọn.
:hover	a:hover	Chọn các liên kết được hover (di chuyển chuột lên thành phần).
:lang(mã ngôn ngữ)	p:lang(vi)	Chọn thành phần với thuộc tính ngôn ngữ có giá trị bắt đầu với "mã ngôn ngữ".
:link	a:link	Định dạng cho tất cả liên kết khi chưa được click.
:visited	a:visited	Định dạng cho các thành phần đã được click.
[attribute]	[target]	Chọn tất cả các thành phần có cùng thuộc tính (attribute).
[attribute]=value	[lang]=vi	Được sử dụng để chọn tất cả các thành phần với thuộc tính có giá trị = "ngôn ngữ".
[attribute]~=value	[title~=myWeb]	Chọn tất cả các thành phần với thuộc tính chứa một giá trị.
[attribute]=value	[target=_blank]	Chọn tất cả các thành phần với thuộc tính bằng giá trị.

4. Các loại bộ chọn (selector)

Bộ chọn	Ví dụ	Mô tả
tag01 ~ tag02	ul ~ p	Chọn tất cả thành phần 02 khi có thành phần 01 ở trước.
[attribute^=value]	img[src^="bnr_"]	Chọn tất cả thành phần với thuộc tính có giá trị bắt đầu bằng "value".
[attribute\$=value]	img[src\$=".gif"]	Chọn tất cả thành phần với thuộc tính có giá trị kết thúc bằng "value".
[attribute*=value]	img[src*=""]	Chọn tất cả thành phần với thuộc tính có giá trị chứa "value".
:first-of-type	p:first-of-type	Chọn thành phần con đầu tiên hoặc duy nhất trong các thành phần cha.
:last-of-type	p:last-of-type	Chọn thành phần con cuối cùng hoặc duy nhất trong các thành phần cha.
:only-of-type	p:only-of-type	Chọn thành phần con trong các thành phần cha, khi thành phần cha có một thành phần con là chính nó. Trong thành phần cha có thể chứa nhiều thành phần con, tuy nhiên thành phần con được chọn phải là duy nhất, không được có từ 2 trở lên.
:only-child	p:only-child	Chọn thành phần con trong các thành phần cha, khi thành phần cha có mỗi thành phần con là chính nó, không được chứa thành phần con khác. Chỉ chọn thành phần có thành phần cha, những thành phần độc lập sẽ không được chọn.

4. Các loại bộ chọn (selector)

Bộ chọn	Ví dụ	Mô tả
:nth-child(n)	p:nth-child(3)	Chọn thành phần thứ "n" trong thành phần cha. Chỉ chọn thành phần có thành phần cha, những thành phần độc lập sẽ không được chọn.
:nth-last-child(n)	p:nth-last-child(3)	Chọn thành phần thứ "n" tính từ thành phần cuối trong thành phần cha. Chỉ chọn thành phần có thành phần cha, những thành phần độc lập sẽ không được chọn.
:nth-of-type(n)	p:nth-of-type(3)	Chọn thành phần thứ "n".
:nth-last-of-type(n)	p:nth-last-of-type(3)	Chọn thành phần thứ "n" từ thành phần cuối trong thành phần cha.
:last-child	p:last-child	Chọn thành phần cuối cùng trong thành phần cha. Chỉ chọn thành phần có thành phần cha, những thành phần độc lập sẽ không được chọn.
:root	:root	Chọn thành phần gốc của văn bản.
:empty	p:empty	Chọn thành phần không chứa thành phần khác.
:target	#event:target	Sử dụng trong liên kết anchor name (link neo).

4. Các loại bộ chọn (selector)

Bộ chọn	Ví dụ	Mô tả
:enabled	input:enabled	Chọn thành phần <i>được kích hoạt</i> (enabled), thường sử dụng cho các thành phần của form
:disabled	input:disabled	Chọn thành phần <i>bị vô hiệu hóa</i> (disabled), thường sử dụng cho các thành phần của form
:checked	input:checked	Chọn thành phần <i>được check</i> (checked), thường sử dụng cho các thành phần của form
:not(bộ chọn)	:not(p)	Chọn tất cả ngoại trừ <i>bộ chọn</i> trong ngoặc..
::selection	::selection	Chọn phần tử được người dùng chọn.

5. Các bộ thuộc tính cơ bản trong CSS

Bộ thuộc tính màu sắc cho text (color):

color: *value*

value có các dạng set màu như sau:

- *plain english* (red, blue, green, ...)
- *hex code* (#fff, #000, ...)
- *rgb* (*rgb*(red,green,blue)) với các giá trị red, green, blue nằm trong khoảng [0, 255]

Bộ thuộc tính về font:

font-size, font-weight, font-family, ...

5. Các bộ thuộc tính cơ bản trong CSS

Bộ thuộc tính về căn chỉnh text:

text-align (right, left, center, ...)

text-decoration(none, ...)

text-transform(uppercase, lowercase, ...)

Bộ thuộc tính về background

background-color

background-image: url("path")

background-position (center, left, right, ...)

background-size (cover, ...)

background-repeat (no-repeat, ...)

5. Các bộ thuộc tính cơ bản trong CSS

Bộ thuộc tính về box model

- *width (px, rem, em, ...)*
- *height (px, rem, em, ...)*
- *margin (px, rem, em, ...)*
- *padding (px, rem, em, ...)*
- *border (px, rem, em, ...)*

Bộ thuộc tính display

display (none, block, inline, inline-block, ...)

6. Mô hình hộp trong CSS

Mỗi phần tử HTML có thể được coi như là một chiếc hộp

Trong CSS thuật ngữ mô hình hộp (box model) được dùng để nói về việc thiết kế bố cục

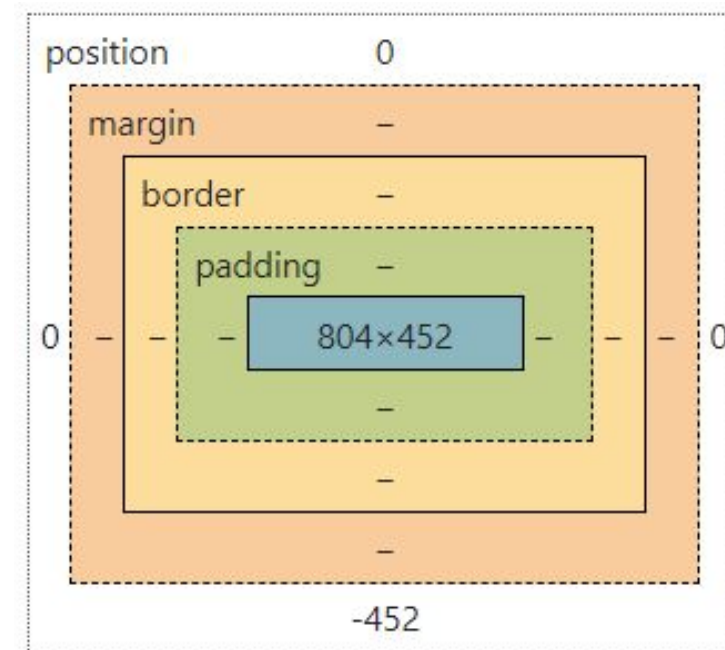
Về cơ bản mô hình hộp CSS là một chiếc hộp bao quanh phần tử HTML như:

→ **Content**: nội dung trong hộp, nơi hiển thị văn bản và hình ảnh

→ **Padding**: Khoảng cách từ nội dung đến viền, chính là phần khoảng trắng xung quanh nội dung, trong suốt

→ **Border**: Đường viền xung quanh nội dung và padding

→ **Margin**: Phần lề bên ngoài đường viền, trong suốt



6. Mô hình hộp trong CSS

Độ rộng của phần tử thực tế:

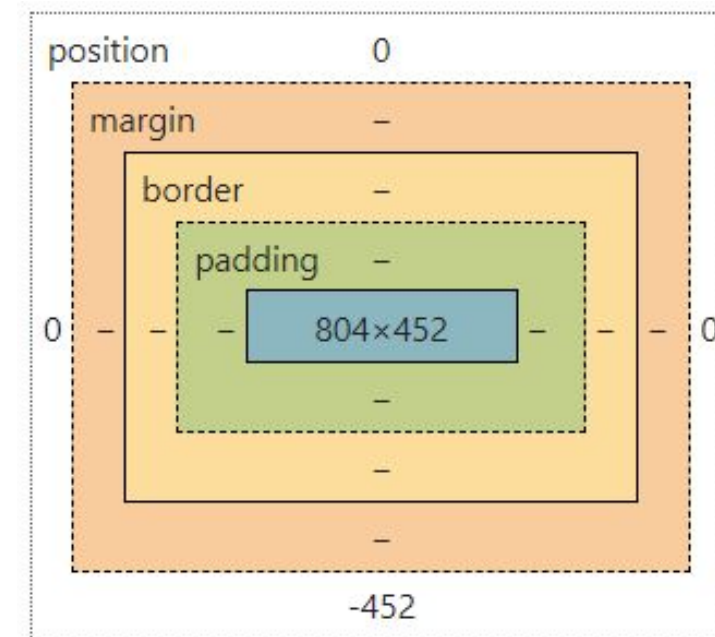
Chiều rộng tổng thể = chiều rộng Content + padding trái + padding phải +
đường viền trái + đường viền phải + margin trái + margin phải

Chiều cao của phần tử thực tế:

Chiều cao tổng thể = chiều cao Content + padding

trên + padding dưới + đường viền trên + đường

viền dưới + margin trên + margin dưới



- ❑ CSS là gì?
- ❑ Tại sao lại cần đến CSS?
- ❑ CSS có thể thực hiện được những hiệu ứng gì?
- ❑ Làm thế nào để nhúng CSS vào trong tài liệu HTML?
- ❑ Khi nào thì sử dụng inline-style, internal style hoặc external style?
- ❑ Cú pháp khai báo bộ quy tắc (rule) trong CSS là như thế nào?
- ❑ Làm thế nào để ghi chú trong CSS?
- ❑ Trong CSS thuật ngữ mô hình hộp (box model) được dùng để nói về việc thiết kế bố cục (layout). Mỗi phần tử HTML có thể được coi như là một chiếc hộp



KẾT THÚC

HỌC VIỆN ĐÀO TẠO LẬP TRÌNH CHẤT LƯỢNG NHẬT BẢN