

DHTML

JAVA  
SCRIPT

LẬP TRÌNH  
WEB

PHP

## Chương 4-CSS

Trình bày: Nguyễn Phú Trường

Giới thiệu CSS

Các loại CSS

Cách dùng CSS

Các CSS cơ bản

Các CSS nâng cao

- ❖ **CSS** viết tắt từ **Cascading Style Sheets**
- ❖ Các kiểu (Styles) định nghĩa cách hiển thị các phần tử HTML
- ❖ Các kiểu được bổ sung vào HTML 4.0 để giải quyết vấn đề về hiển thị
- ❖ **Các bảng kiểu ngoài (External Style Sheets)** có thể tiết kiệm nhiều việc
- ❖ Các bảng kiểu ngoài được lưu vào các tập tin **.CSS**
- ❖ Cùng một nội dung dữ liệu có thể hiển thị theo nhiều kiểu khác nhau
- ❖ Demo minh họa

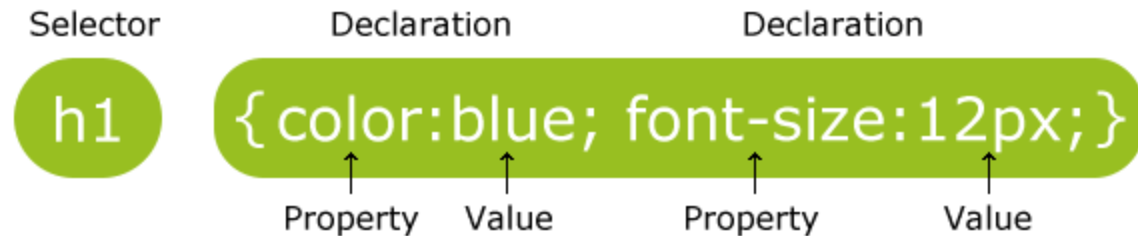
# TẠI SAO CSS?



- ❖ HTML không chứa các thẻ để định dạng tài liệu.
- ❖ HTML định nghĩa nội dung của tài liệu như:
  - `<h1>This is a heading</h1>`
  - `<p>This is a paragraph.</p>`
- ❖ Khi thẻ `<font>`, và thuộc tính `color` được thêm tới đặc tả HTML 3.2
  - Nó sinh ra ra một vấn đề cho người phát triển web.
  - Việc phát triển các website lớn chứa thông tin fonts và color được thêm tới mỗi trang web riêng tốn nhiều thời gian và chi phí
- ❖ Giải quyết vấn đề này, W3C tạo ra CSS.
- ❖ Trong HTML 4.0, tất cả định dạng được gỡ khỏi tài liệu HTML, và được lưu vào một tập tin CSS riêng.
- ❖ Tất cả trình duyệt ngày nay đều hỗ trợ CSS.

- ❖ CSS định nghĩa cách các phần tử HTML được hiển thị.
- ❖ Các kiểu thường được lưu vào tập tin .CSS.
- ❖ Các bảng kiểu ngoài cho phép thay đổi diện mạo và canh lề cả tất cả Web site chỉ bằng hiệu chỉnh một tập tin đơn!

❖ Cú pháp CSS gồm 3 phần: bộ chọn, thuộc tính và giá trị



❖ Ở đây:

- Bộ chọn (selector): thường là các thành phần/thẻ HTML
- Thuộc tính (property): là thuộc tính muốn thay đổi, mỗi thuộc tính có thể mang 1 giá trị.
- Thuộc tính và giá trị được ngăn cách bởi dấu “:” và được bao quanh dấu ngoặc nhọn “{}”.

❖ Thí dụ:

- `body {color:black}`
- `p {font-family:"sans serif"}`
- `p {text-align:center;color:red}`

❖ **Chú ý:**

- Nếu giá trị có nhiều từ thì đặt chúng trong dấu nháy đôi (").
- Nếu muốn xác định nhiều hơn một thuộc tính, phải ngăn cách mỗi thuộc tính với dấu chấm phẩy (;)

❖ Để tạo các định nghĩa kiểu dễ đọc hơn, có thể mô tả một thuộc tính trên một dòng

- ```
p
{
color:red;
text-align:center;
}
```



- ❖ Có thể nhóm các bộ chọn.
- ❖ Ngăn cách mỗi bộ chọn với dấu phẩy (,).
- ❖ Thí dụ:

*h1,h2,h3,h4,h5,h6*

{

*color:green*

}

- ❖ Để bổ sung thiết lập một kiểu cho phần tử HTML, CSS cho phép bạn xác định bộ chọn của chính mình.
- ❖ Gồm 2 loại
  - Bộ chọn lớp- class selector
  - Bộ chọn định danh-id selector

- ❖ Được dùng để xác định kiểu cho 1 phần tử đơn, duy nhất
- ❖ Dùng thuộc tính ID của phần tử HTML và được định nghĩa với “#”
- ❖ Thí dụ:
  - #para1
    - {
    - text-align:center;
    - color:red;
    - }
  - Được áp dụng tới phần tử với id="para1"
- ❖ Chú ý: không được bắt đầu tên id với 1 số, nó sẽ không làm việc trong Mozilla và FireFox

- ❖ Được dùng để xác định 1 kiểu cho 1 nhóm các phần tử.
- ❖ Không giống bộ chọn định danh, bộ chọn lớp dùng cho nhiều phần tử
- ❖ Thiết lập một kiểu xác định cho bất cứ phần tử HTML với cùng lớp.
- ❖ Dùng thuộc tính class của phần tử HTML và được định nghĩa với một dấu “.”
- ❖ Thí dụ:
  - `.center {text-align:center;}`
  - Tất cả phần tử HTML với `class="center"` sẽ được canh giữa

- ❖ Với bộ chọn lớp có thể định nghĩa các kiểu khác nhau cho cùng loại phần tử HTML.
- ❖ Thí dụ: nếu chúng ta muốn có hai kiểu đoạn tài liệu: một canh lề phải và một canh giữa, chúng ta có kiểu sau:
  - `p.right {text-align:right}`
  - `p.center {text-align:center}`
- ❖ Dùng thuộc tính class trong tài liệu HTML:
  - `<p class="right">This paragraph will be right-aligned.</p>`
  - `<p class="center">This paragraph will be center-aligned.</p>`

- ❖ **Chú ý:** Để áp dụng nhiều hơn một class trên phần tử được cho, cú pháp như sau:
  - `<p class="center bold">This is a paragraph.</p>`
- ❖ Đoạn trên sẽ được định kiểu bởi class "**center**" AND class "**bold**".
- ❖ Cũng có thể bỏ tên thẻ trong bộ chọn để định nghĩa kiểu sẽ được dùng bởi tất cả thành phần HTML có lớp cụ thể.
- ❖ Trong thí dụ sau, tất cả thành phần HTML với `class="center"` sẽ được canh giữa:
  - `.center {text-align:center}`

❖ Trong mã dưới đây cả hai phần tử `h1` và `p` có `class="center"`. Điều này có nghĩa là cả hai phần tử sẽ tuân theo qui cách trong bộ chọn `center`:

- `<h1 class="center">`This heading will be center-aligned`</h1>`  
`<p class="center">`This paragraph will also be center-aligned.`</p>`

### ❖ Chú ý:

- Không bắt đầu tên lớp với một số! Điều này chỉ được hỗ trợ trong Internet Explorer.

- ❖ Cũng có thể áp dụng các kiểu tới các phần tử HTML với các thuộc tính xác định
- ❖ Qui cách kiểu dưới đây sẽ phù hợp cho tất cả phần tử **input** có thuộc tính **type** có giá trị “**text**”:
- ❖ **input**[type="**text**"] {background-color:blue}



- ❖ Các chú thích được dùng để giải thích mã của bạn và có thể giúp bạn khi hiệu chỉnh mã nguồn sau đó.
- ❖ Một chú thích sẽ được bỏ qua bởi trình duyệt
- ❖ Chú thích trong CSS bắt đầu với "/\*", và kết thúc với "\*/", như thí dụ:
  - ```
/*This is a comment*/  
p  
{  
text-align:center;  
/*This is another comment*/  
color:black;  
font-family:arial  
}
```

- ❖ Khi trình duyệt đọc một style sheet, nó sẽ định dạng tài liệu dựa theo style sheet đó.
- ❖ Có ba cách để chèn một style sheet:
  - External style sheet (bảng kiểu ngoài)
  - Internal style sheet (bảng kiểu trong)
  - Inline style (bảng kiểu trên dòng)

- ❖ Bảng kiểu ngoài là lý tưởng khi kiểu được áp dụng tới nhiều trang.
- ❖ Với bảng kiểu ngoài, bạn có thể thay đổi diện mạo của toàn bộ website bằng cách thay đổi một tập tin.
- ❖ Mỗi trang phải liên kết tới bảng kiểu dùng thẻ `<link>`.
- ❖ Thẻ `<link>` nằm bên trong phần head:
  - `<head>`  
`<link rel="stylesheet" type="text/css" href="mystyle.css" />`  
`</head>`

- ❖ Bảng kiểu ngoài có thể được viết trong bất cứ trình soạn thảo nào.
- ❖ Tập tin này sẽ không chứa bất cứ thẻ HTML. Bảng kiểu nên được lưu trong tập tin có phần mở rộng .css.
- ❖ Thí dụ, một tập tin bảng kiểu nên được hiển thị như sau:
  - `hr {color:sienna}`  
`p {margin-left:20px}`  
`body {background-image:url("images/back40.gif")}`
- ❖ Đừng để khoảng trống giữa giá trị thuộc tính và đơn vị tính!
  - "margin-left:20 px" (thay vì "margin-left:20px") sẽ làm việc trong IE, nhưng không làm việc trong Firefox hay Opera.

- ❖ Bảng kiểu trong nên được dùng khi một tài liệu đơn có kiểu duy nhất.
- ❖ Bạn định nghĩa các kiểu bên trong trong phần head của trang HTML, bằng cách dùng thẻ `<style>`, như sau:
  - `<head>`  
`<style type="text/css">`  
`hr {color:sienna}`  
`p {margin-left:20px}`  
`body {background-image:url("images/back40.gif")}`  
`</style>`  
`</head>`

- ❖ Một kiểu trên dòng mất nhiều lợi điểm của bảng kiểu bằng cách xen lẫn nội dung với biểu diễn.
- ❖ Dùng phương pháp này một cách “hạn chế”!
- ❖ Để dùng các kiểu trên dòng, bạn dùng thuộc tính style trong thẻ tương ứng.
- ❖ Thuộc tính style có thể chứa bất cứ thuộc tính CSS.
- ❖ Thí dụ hiển thị cách đổi màu và canh trái của một đoạn
  - `<p style="color:sienna;margin-left:20px">`  
This is a paragraph.  
`</p>`

- ❖ Nếu có vài thuộc tính được thiết lập cho cùng một bộ chọn trong các bảng kiểu khác nhau, các giá trị này sẽ được kế thừa từ nhiều hơn bảng kiểu xác định.
- ❖ Thí dụ, 1bảng kiểu ngoài có các thuộc tính này cho bộ chọn h3
  - h3

```
{
color:red;
text-align:left;
font-size:8pt
}
```

- ❖ Và 1bảng kiểu trong có các thuộc tính này cho bộ chọn h3:
  - h3
    - {
    - text-align:right;
    - font-size:20pt
    - }
- ❖ Nếu trang này với bảng kiểu trong cũng liên kết tới bảng kiểu ngoài, các thuộc tính cho h3 sẽ là:
  - color:red;
  - text-align:right;
  - font-size:20pt
- ❖ Màu sẽ được kế thừa từ bảng kiểu ngoài và canh lề văn bản và kích thước font bị thay thế bởi bảng kiểu trong.



## ❖ Các kiểu có thể được xác định

- Bên trong một phần tử
- Bên trong phần head của một trang HTML
- Trong tập tin CSS ngoài

## ❖ Chú ý:

- Nhiều bảng kiểu có thể được tham chiếu bên trong một tài liệu HTML.

# THỨ TỰ PHÂN TẦNG



- ❖ Kiểu gì sẽ được dùng khi có nhiều hơn một kiểu xác định cho một phần tử HTML?
- ❖ Thường thì tất cả các kiểu sẽ được “xếp” thành một bảng kiểu mới “ảo” theo những qui tắc mà ở đó 4 có độ ưu tiên cao nhất
  - Browser default
  - External style sheet
  - Internal style sheet (trong phần head )
  - Inline style (bên trong phần tử HTML)

- ❖ Vì thế, kiểu trên dòng(bên trong phần tử HTML) có độ ưu tiên cao nhất, có nghĩa nó sẽ đè:
  - Kiểu được định nghĩa bên trong thẻ <head>, hay
  - Trong bảng kiểu bên ngoài, hay
  - Trong trình duyệt (giá trị mặc định).
- ❖ **Chú ý:**
  - Nếu liên kết tới bảng kiểu ngoài được thay thế sau khi bảng kiểu bên trong thẻ HTML <head>, thì bảng kiểu ngoài sẽ đè bảng kiểu trong!

- ❖ Các thuộc tính nền CSS được dùng để định nghĩa các hiệu ứng nền của một phần tử.
- ❖ Các thuộc tính CSS được dùng cho hiệu ứng nền:
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position

- ❖ Thuộc tính **background-color** xác định màu nền của một phần tử.
- ❖ Màu nền của trang được định nghĩa trong bộ chọn body:
- ❖ Thí dụ
  - `body {background-color:#b0c4de}`
- ❖ Màu nền có thể được xác định bởi:
  - name – tên một màu, như "red" cho màu đỏ
  - RGB – giá trị RGB, như "rgb(255,0,0)"
  - Hex – giá trị thập lục phân, như "#ff0000"

❖ Trong thí dụ dưới đây phần tử h1, p, và div có màu nền khác nhau:

❖ Thí dụ:

- h1 {background-color:#6495ed}

p {background-color:#e0ffff}

div {background-color:#b0c4de}

- ❖ Thuộc tính background-image xác định ảnh dùng làm nền cho một phần tử.
- ❖ Mặc định, ảnh được lặp để mà nó phủ toàn bộ phần tử.
- ❖ Ảnh nền đối với một trang có thể được thiết lập như sau:
- ❖ Thí dụ:
  - `body {background-image:url('paper.gif')}`  
Below is an example of a bad combination of text and background image. The text is almost not readable:
- ❖ **Chú ý: cần có sự kết hợp giữa văn bản và màu nền để văn bản có thể đọc được**
  - Thí dụ:

- ❖ Mặc định, thuộc tính background-image lặp ảnh cả chiều ngang và chiều đứng.
- ❖ Một vài ảnh nên được lặp chỉ ngang hay đứng, hay chúng sẽ trông rất lạ như thế này:
  - Body { background-image:url('gradient2.png'); }
- ❖ Nếu ảnh được lặp chỉ theo chiều ngang (repeat-x), nền sẽ trông tốt hơn:
  - body  
{  
background-image:url('gradient2.png');  
**background-repeat:repeat-x;**  
}



- ❖ Khi sử dụng một ảnh nền, dùng một ảnh mà không biến dạng văn bản.
- ❖ Hiển thị hình ảnh chỉ một lần được xác định bởi thuộc tính background-repeat

### ❖ Thí dụ

- body

```
{    background-image:url('img_tree.png');
    background-repeat:no-repeat;
}
```
- Trong thí dụ trên, ảnh nền được hiển thị trong cùng vị trí với văn bản. Chúng ta phải chuyển vị trí của ảnh để mà nó không biến dạng văn bản quá nhiều

❖ Vị trí của ảnh được xác định bởi thuộc tính background-position:

❖ Thí dụ

- body

```
{
background-image:url('img_tree.png');
background-repeat:no-repeat;
background-position:top right;
}
```

- ❖ Như đã thấy trong các thí dụ trên, có nhiều thuộc tính để xem xét liên quan tới nền.
- ❖ Để đơn giản mã, có thể xác định tất cả thuộc tính trong một thuộc tính đơn. Thuộc tính này là shorthand (nhanh).
- ❖ Thuộc tính nhanh cho nền đơn giản là "background":
- ❖ Thí dụ:
  - `body {background:#ffffff url('img_tree.png') no-repeat top right}`

- ❖ Khi dùng thuộc tính nhanh thứ tự của các giá trị thuộc tính là:
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position
- ❖ Không có vấn đề nếu một trong các giá trị thuộc tính bị thiếu, miễn là chúng được hiện diện trong thứ tự này.

## TẤT CẢ THUỘC TÍNH NỀN CSS



Thuộc tính	Mô tả	Giá trị	CS S
<u>background</u>	Sets all the background properties in one declaration	<i>background-color</i> <i>background-image</i> <i>background-repeat</i> <i>background-attachment</i> <i>background-position</i> inherit	1
<u>background-attachment</u>	Sets whether a background image is fixed or scrolls with the rest of the page	scroll fixed inherit	1
<u>background-color</u>	Sets the background color of an element	<i>color-rgb</i> <i>color-hex</i> <i>color-name</i> transparent inherit	1

## TẤT CẢ THUỘC TÍNH NỀN CSS



Thuộc tính	Mô tả	Giá trị	CSS
<u>background-image</u>	Sets the background image for an element	url(URL) none inherit	1
<u>background-position</u>	Sets the starting position of a background image	top left top center top right center left center center center right bottom left bottom center bottom right x% y% xpos ypos inherit	1
<u>background-repeat</u>	Sets if/how a background image will be repeated	repeat repeat-x repeat-y no-repeat inherit	1

- ❖ Các thuộc tính CSS text định nghĩa diện mạo của văn bản
- ❖ Thí dụ gồm vài thuộc tính định dạng văn bản:
  - Tiêu đề dùng:
  - Text-align
  - Text-transform
  - Color

- ❖ Thuộc tính màu được dùng để thiết lập màu của văn bản.
- ❖ Màu có thể được xác định bởi:
  - name – tên màu, như màu đỏ là "red"
  - RGB – giá trị RGB, như "rgb(255,0,0)"
  - Hex – giá trị thập lục phân, như "#ff0000"
- ❖ Màu mặc định cho một trang được định nghĩa trong bộ chọn body.



❖ Thí dụ:

- `body {color:blue}`  
`h1 {color:#00ff00}`  
`h2 {color:rgb(255,0,0)}`

❖ Để W3C tương thích CSS: nếu bạn định nghĩa thuộc tính màu, bạn cũng phải định nghĩa thuộc tính background-color.

- ❖ Thuộc tính text-align được dùng để thiết lập canh lề ngang cho văn bản.
- ❖ Văn bản có thể được canh lề: giữa, trái, phải hay cả hai trái và phải.
- ❖ Khi text-align được đặt tới "justify", mỗi dòng bị kéo giãn để mà mỗi dòng có độ dày bằng nhau và lề trái, phải là thẳng (như tạp chí hay báo).
- ❖ Thí dụ
  - `h1 {text-align:center}`  
`p.date {text-align:right}`  
`p.main {text-align:justify}`

- ❖ Thuộc tính text-decoration được dùng để thiết lập hay xóa thiết lập từ văn bản.
- ❖ Hầu hết, thuộc tính text-decoration được dùng để bỏ gạch dưới từ những liên kết cho mục đích thiết kế:
  - a {text-decoration:none}
- ❖ Nó cũng có thể được dùng để trang trí văn bản
  - h1 {text-decoration:overline}
  - h2 {text-decoration:line-through}
  - h3 {text-decoration:underline}
  - h4 {text-decoration:blink}
- ❖ Văn bản không phải là liên kết không yêu cầu gạch dưới vì sẽ thường làm cho người dùng nhầm lẫn.

- ❖ Thuộc tính text-transform được dùng để xác định chữ hoa và thường trong văn bản.
- ❖ Nó có thể được dùng để chuyển mọi thứ thành hoa hay thường, hay ký tự đầu tiên hoa.
- ❖ Thí dụ:
  - p.uppercase {text-transform:uppercase}
  - p.lowercase {text-transform:lowercase}
  - p.capitalize {text-transform:capitalize}

- ❖ Thuộc tính text-indentation được dùng để xác định lề của dòng đầu tiên trong văn bản.
- ❖ Thí dụ:
  - `p {text-indent:50px}`

Thuộc tính	Mô tả	Giá trị	CSS
<u>color</u>	Sets the color of a text	<i>color</i>	1
<u>direction</u>	Sets the text direction	ltr rtl	2
<u>line-height</u>	Sets the distance between lines	normal <i>number</i> <i>length</i> %	1
<u>letter-spacing</u>	Increase or decrease the space between characters	normal <i>length</i>	1
<u>text-align</u>	Aligns the text in an element	left right center justify	1

Thuộc tính	Mô tả	Giá trị	CSS
<u>text-decoration</u>	Adds decoration to text	none underline overline line-through blink	1
<u>text-indent</u>	Indents the first line of text in an element	<i>length</i> %	1
text-shadow		none <i>color</i> <i>length</i>	
<u>text-transform</u>	Controls the letters in an element	none capitalize uppercase lowercase	1

## TẤT CẢ THUỘC TÍNH VĂN BẢN



Thuộc tính	Mô tả	Giá trị	CSS
unicode-bidi		normal embed bidi-override	2
<u>vertical-align</u>	Sets the vertical alignment of an element	baseline sub super top text-top middle bottom text-bottom <i>length</i> %	1
<u>white-space</u>	Sets how white space inside an element is handled	normal pre nowrap	1
<u>word-spacing</u>	Increase or decrease the space between words	normal <i>length</i>	1



- ❖ Các thuộc tính CSS font định nghĩa: font family, boldness, size, và loại của văn bản the style of a text.
- ❖ Sự khác nhau của font Sans-serif và Serif

F

Sans-serif

F

Serif

F

Serif  
(red serifs)

- ❖ Trên màn hình máy tính, các font Sans-serif được xem là dễ đọc hơn font Serif

❖ Trong CSS, có hai loại tên họ phông:

- **Họ chung** (generic family) – nhóm các họ font trông tương tự nhau (như "Serif" hay "Monospace")
- **Họ phông** (font family) – một họ phông xác định (như "Times New Roman" hay "Arial")

Họ chung	Họ phông	Mô tả
Serif	Times New Roman Georgia	Phông Serif có các dòng nhỏ tại điểm kết thúc trên 1 vài ký tự
Sans-serif	Arial Verdana	"Sans" nghĩa là không- các phông này không có các dòng tại điểm cuối mỗi ký tự
Monospace	Courier New Lucida Console	Tất cả ký tự có cùng độ rộng

- ❖ Họ phông của văn bản được thiết lập với thuộc tính font-family.
- ❖ Thuộc tính font-family có thể quản lý nhiều tên phông như hệ thống "fallback".
- ❖ Nếu browser không hỗ trợ phông thứ nhất, nó thử phông kế.
- ❖ Bắt đầu với phông bạn muốn, và kết thúc với phông họ chung, để browsers chọn một phông tương tự trong họ chung nếu không có phông nào khác sẵn có.

- ❖ **Chú ý:** Nếu tên của họ phông nhiều hơn một từ, nó phải đặt trong dấu “”.
- ❖ Thí dụ: font-family: "Times New Roman".
- ❖ Nhiều hơn một họ phông được xác định trong danh sách được ngăn cách bởi dấu phẩy (,)
- ❖ Thí dụ:
- ❖ p{font-family:"Times New Roman",Georgia,Serif}

- ❖ Thuộc tính font-style thường được dùng để xác định văn bản in nghiêng.
- ❖ Thuộc tính có 3 giá trị:
  - normal – Văn bản hiển thị bình thường
  - italic – Văn bản được hiển thị in nghiêng
  - oblique – Văn bản được hiển thị in xiên (xiên rất giống nghiêng, nhưng được hỗ trợ ít hơn)
- ❖ Thí dụ:
  - p.normal {font-style:normal}
  - p.italic {font-style:italic}
  - p.oblique {font-style:oblique}

- ❖ Thuộc tính font-size thiết lập kích cỡ văn bản.
- ❖ Quản lý kích thước văn bản là quan trọng trong thiết kế web.
- ❖ Tuy nhiên, bạn không nên dùng các hiệu chỉnh để làm các đoạn trông giống như các heading, hay các heading giống như các đoạn.
- ❖ Luôn dùng các thẻ HTML phù hợp, như `<h1>` - `<h6>` cho heading và `<p>` đoạn.
- ❖ Giá trị font-size có thể là tuyệt đối hay tương đối.

### ❖ Kích thước tuyệt đối:

- Thiết lập văn bản tới một kích thước xác định
- Không cho phép người dùng thay đổi kích thước văn bản trong tất cả trình duyệt (hạn chế cho lý do khả năng truy cập)
- Kích thước tuyệt đối có ích khi kích thước vật lý của dữ liệu xuất được biết

### ❖ Kích thước tương đối:

- Thiết lập kích thước tương đối so với các thành phần phụ cận
- Cho phép người dùng thay đổi kích thước văn bản trong trình duyệt
- Nếu không xác định kích thước phông, kích thước mặc định cho văn bản thông thường như đoạn là 16px (16px=1em).

- ❖ Thiết lập kích cỡ văn bản với pixels, cho bạn điều khiển hoàn toàn kích cỡ văn bản:
- ❖ Thí dụ
  - `h1 {font-size:40px}`  
`h2 {font-size:30px}`  
`p {font-size:14px}`
- ❖ Thí dụ trên cho phép Firefox, Chrome, và Safari định lại kích thước văn bản, **nhưng Internet Explorer thì không**
- ❖ Văn bản có thể được định lại kích thước trong tất cả trình duyệt dùng công cụ phóng to/thu nhỏ (tuy nhiên, điều này định vị toàn bộ trang, nhưng chỉ văn bản).



- ❖ Để tránh vấn đề định lại kích cỡ với IE, nhiều người phát triển dùng em thay cho pixel.
- ❖ Đơn vị kích thước em được đề nghị bởi W3C.
- ❖ 1em bằng kích thước phong hiện hành. Kích thước văn bản mặc định là 16px. Vì thế, mặc định  $1\text{ em} = 16\text{px}$ .
- ❖ Kích thước có thể được tính từ pixels tới em dùng công thức:  $\text{pixels}/16 = \text{em}$

### ❖ Thí dụ:

- `h1 {font-size:2.5em} /* 40px/16=2.5em */`  
`h2 {font-size:1.875em} /* 30px/16=1.875em */`  
`p {font-size:0.875em} /* 14px/16=0.875em */`
- Trong thí dụ trên, kích thước văn bản tính bằng em bằng với thí dụ trước tính bằng pixel. Tuy nhiên, với kích thước em, kích thước văn bản có thể hiệu chỉnh trong tất cả browser.
- ❖ Tuy nhiên, vấn đề vẫn còn với IE. Khi định lại kích thước văn bản, văn bản trở nên lớn hơn khi được làm lớn và nhỏ hơn khi được làm nhỏ hơn.

- ❖ Giải pháp làm việc trong tất cả browser là thiết lập kích thước phông mặc định bằng % cho phần tử body.
- ❖ **Thí dụ**
  - `body {font-size:100%}`  
`h1 {font-size:2.5em}`  
`h2 {font-size:1.875em}`  
`p {font-size:0.875em}`
  - Mã bây giờ làm việc tốt! Hiển thị cùng kích thước văn bản trong tất cả browser, và cho phép tất cả browser phóng to/thu nhỏ hay định lại kích thước văn bản!

## TẤT CẢ THUỘC TÍNH PHÒNG



Thuộc tính	Mô tả	Giá trị	CSS
<u>font</u>	Sets all the font properties in one declaration	<i>font-style</i> <i>font-variant</i> <i>font-weight</i> <i>font-size/line-height</i> <i>font-family</i> caption icon menu message-box small-caption status-bar inherit	1

## TẤT CẢ THUỘC TÍNH PHÒNG



Thuộc tính	Mô tả	Giá trị	CSS
<u>font-family</u>	Specifies the font family for text	<i>family-name</i> <i>generic-family</i> inherit	1
<u>font-size</u>	Specifies the font size of text	xx-small x-small small medium large x-large xx-large smaller larger <i>length</i> % inherit	1

## TẤT CẢ THUỘC TÍNH PHÒNG



Thuộc tính	Mô tả	Giá trị	CSS
<u>font-style</u>	Specifies the font style for text	normal italic oblique inherit	1
<u>font-variant</u>	Specifies whether or not a text should be displayed in a small-caps font	normal small-caps inherit	1

- ❖ Tất cả phần tử HTML có thể được xem như các hộp (boxes). Trong CSS, thuật ngữ “mô hình hộp” (box model) được dùng khi nói về thiết kế và sắp xếp (layout).
- ❖ Mô hình hộp CSS là 1 hộp mà bao bọc các phần tử HTML, và chứa: lề, đường viền, padding, và nội dung thật.
- ❖ Mô hình hộp cho phép thay thế đường viền bao quanh các phần tử và các phần tử trống liên quan với các phần tử khác.
- ❖ Hình ảnh bên hiển thị mô hình hộp:

# MÔ HÌNH HỘP CSS





❖ Giải thích các phần khác nhau:

- **Lề (Margin)** – chiếm 1 vùng quanh đường viền. Lề không có màu nền và nó hoàn toàn trong suốt
- **Đường viền (Border)** – Đường viền nằm giữa padding và nội dung. Đường viền bị ảnh hưởng bởi màu nền của hộp
- **Padding** – chiếm một vùng quanh nội dung. Padding bị ảnh hưởng bởi màu nền của hộp
- **Nội dung (Content)** – nội dung của hộp, nơi văn bản và hình ảnh xuất hiện

❖ Để thiết lập độ rộng và chiều cao của một phần tử chính xác trong tất cả trình duyệt, bạn cần biết cách mô hình hộp làm việc.

- ❖ **Quan trọng:** khi bạn xác định thuộc tính độ rộng và chiều cao của một phần tử với CSS, bạn chỉ thiết lập độ rộng và chiều cao của vùng nội dung. Để biết kích thước đầy đủ của phần tử, bạn phải thêm vào padding, đường viền và lề.
- ❖ Toàn bộ độ rộng của phần tử trong thí dụ dưới đây là 300px:
  - width:250px;  
padding:10px;  
border:5px solid gray;  
margin:10px;

- ❖ Nếu bạn đã kiểm tra thí dụ trước trong IE, bạn thấy rằng độ rộng tổng cộng không bằng 250px.
- ❖ IE chứa padding và đường viền trong độ rộng, khi thuộc tính độ rộng được thiết lập, **trừ khi DOCTYPE được khai báo.**
- ❖ Để giải quyết vấn đề này, chỉ cần thêm DOCTYPE tới mã:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html>  
<head>  
<style type="text/css">  
div.ex  
{  
width:220px;  
padding:10px;  
border:5px solid gray;  
margin:0px;  
}  
</style>  
</head>
```

- ❖ Các thuộc tính đường viền CSS định nghĩa các đường viền xung quanh phần tử.
- ❖ Các thuộc tính đường viền CSS cho phép bạn đặc tả kiểu và màu sắc của đường viền của phần tử.
- ❖ Các loại đường viền
  - Thuộc tính loại xác định loại đường viền gì được hiển thị.
  - Không có thuộc tính đường viền nào khác sẽ có ảnh hưởng ngoài trừ kiểu đường viền được đặt
- ❖ Giá trị loại đường viền

- none: Defines no border
- dotted: Defines a dotted border
- dashed: Defines a dashed border
- solid: Defines a solid border
- double: Defines two borders. The width of the two borders are the same as the border-width value
- groove: Defines a 3D grooved border. The effect depends on the border-color value
- ridge: Defines a 3D ridged border. The effect depends on the border-color value
- inset: Defines a 3D inset border. The effect depends on the border-color value
- outset: Defines a 3D outset border. The effect depends

# ĐỘ RỘNG ĐƯỜNG VIỀN



- ❖ Thuộc tính border-width được dùng để đặt độ rộng đường viền.
- ❖ Độ rộng được đặt theo đơn vị pixels, hay sử dụng một trong ba giá trị được định nghĩa: thin, medium, hay thick.
- ❖ **Chú ý:** Thuộc tính "border-width" không làm việc nếu nó được dùng một mình. Dùng thuộc tính "border-style" để đặt đường viền trước.

❖ **Thí dụ**

## ❖ p.one

```
{  
border-style:solid;  
border-width:5px;  
}
```

## p.two

```
{  
border-style:solid;  
border-width:medium;  
}
```



- ❖ Thuộc tính border-color được dùng để thiết lập màu của đường viền. Màu có thể được đặt bởi:
  - name- xác định tên màu, như "red"
  - RGB – xác định giá trị RGB, như "rgb(255,0,0)"
  - Hex – xác định giá trị thập lục phân, như "#ff0000"
- ❖ Bạn cũng có thể thiết lập màu đường viền trong suốt "transparent".
- ❖ **Chú ý:** thuộc tính "border-color" không làm việc nếu nó được dùng một mình. Dùng thuộc tính "border-style" để thiết lập đường viền trước.

```
❖ p.one
{
border-style:solid;
border-color:red;
}
p.two
{
border-style:solid;
border-color:#98bf21;
}
```

❖ Trong CSS có thể xác định các đường viền khác nhau cho các phía khác nhau:

❖ Thí dụ:

```
p
{
border-top-style:dotted;
border-right-style:solid;
border-bottom-style:dotted;
border-left-style:solid;
}
```

❖ Trong thí dụ trên cũng có thể thiết lập một thuộc tính đơn:

- border-style:dotted solid;

- ❖ Thuộc tính border-style có thể có từ 1 đến 4 giá trị.
- ❖ **border-style:dotted solid double dashed;**
  - top border là dotted
  - right border là solid
  - bottom border là double
  - left border là dashed
- ❖ **border-style:dotted solid double;**
  - top border là dotted
  - right và left borders là solid
  - bottom border là double

- ❖ Thuộc tính border-style có thể có từ 1 đến 4 giá trị.
- ❖ **border-style:dotted solid;**
  - top và bottom borders là dotted
  - right và left borders là solid
- ❖ **border-style:dotted;**
  - Bốn borders là dotted
- ❖ Thuộc tính border-style được dùng trong thí dụ ở trên. Tuy nhiên, nó cũng làm việc với border-width và border-color.

- ❖ Như bạn thấy trong thí dụ trên, có nhiều thuộc tính xem xét khi làm việc với các đường viền.
- ❖ Để làm ngắn mã, nó cũng có thể xác định tất cả thuộc tính đường viền trong một thuộc tính. Điều này được gọi là thuộc tính **shorthand**.
- ❖ Thuộc tính shorthand cho các thuộc tính đường viền là "border":
- ❖ **Thí dụ:**
  - border:5px solid red;

- ❖ Khi sử dụng thuộc tính border, thứ tự của các giá trị là:
  - border-width
  - border-style
  - border-color
- ❖ Nó không là vấn đề nếu một trong các giá trị trên bị mất (mặc dù, border-style được yêu cầu), miễn là phần còn lại trong thứ tự xác định.

## TẤT CẢ THUỘC TÍNH CSS BORDER



Thuộc tính	Mô tả	Giá trị	CSS
<u><a href="#">border</a></u>	Sets all the border properties in one declaration	<i>border-width</i> <i>border-style</i> <i>border-color</i>	1
<u><a href="#">border-bottom</a></u>	Sets all the bottom border properties in one declaration	<i>border-bottom-width</i> <i>border-bottom-style</i> <i>border-bottom-color</i>	1
<u><a href="#">border-bottom-color</a></u>	Sets the color of the bottom border	<i>border-color</i>	2
<u><a href="#">border-bottom-style</a></u>	Sets the style of the bottom border	<i>border-style</i>	2
<u><a href="#">border-bottom-width</a></u>	Sets the width of the bottom border	<i>border-width</i>	1
<u><a href="#">border-color</a></u>	Sets the color of the four borders	<i>color_name</i> <i>hex_number</i> <i>rgb_number</i> transparent inherit	1



## TẤT CẢ THUỘC TÍNH CSS BORDER



Thuộc tính	Mô tả	Giá trị	CSS
<a href="#">border-left</a>	Sets all the left border properties in one declaration	<i>border-left-width</i> <i>border-left-style</i> <i>border-left-color</i>	1
<a href="#">border-left-color</a>	Sets the color of the left border	<i>border-color</i>	2
<a href="#">border-left-style</a>	Sets the style of the left border	<i>border-style</i>	2
<a href="#">border-left-width</a>	Sets the width of the left border	<i>border-width</i>	1
<a href="#">border-right</a>	Sets all the right border properties in one declaration	<i>border-right-width</i> <i>border-right-style</i> <i>border-right-color</i>	1
<a href="#">border-right-color</a>	Sets the color of the right border	<i>border-color</i>	2
<a href="#">border-right-style</a>	Sets the style of the right border	<i>border-style</i>	2
<a href="#">border-right-width</a>	Sets the width of the right border	<i>border-width</i>	1

## TẤT CẢ THUỘC TÍNH CSS BORDER



Thuộc tính	Mô tả	Giá trị	CSS
<a href="#">border-style</a>	Sets the style of the four borders	none hidden dotted dashed solid double groove ridge inset outset inherit	1
<a href="#">border-top</a>	Sets all the top border properties in one declaration	<i>border-top-width</i> <i>border-top-style</i> <i>border-top-color</i>	1
<a href="#">border-top-color</a>	Sets the color of the top border	<i>border-color</i>	2
<a href="#">border-top-style</a>	Sets the style of the top border	<i>border-style</i>	2
<a href="#">border-top-width</a>	Sets the width of the top border	<i>border-width</i>	1
<a href="#">border-width</a>	Sets the width of the four borders	thin medium thick <i>length</i> inherit	1

- ❖ Outline là đường được vẽ xung quanh các phần tử, ngoài rìa đường viền, để làm thành phần “nổi bật”.
- ❖ Các thuộc tính outline xác định kiểu, màu, và độ rộng của outline.
- ❖ Ví dụ:
  - Vẽ đường quanh phần tử outline
  - Thiết lập kiểu outline
  - Thiết lập màu của outline
  - Thiết lập độ rộng của outline

## TẤT CẢ THUỘC TÍNH OUTLINE



Thuộc tính	Mô tả	Giá trị	CSS
<u>outline</u>	Sets all the outline properties in one declaration	<i>outline-color</i> <i>outline-style</i> <i>outline-width</i> inherit	2
<u>outline-color</u>	Sets the color of an outline	<i>color_name</i> <i>hex_number</i> <i>rgb_number</i> invert inherit	2

## TẤT CẢ THUỘC TÍNH OUTLINE



Thuộc tính	Mô tả	Giá trị	CSS
<u>outline-style</u>	Sets the style of an outline	none dotted dashed solid double groove ridge inset outset inherit	2
<u>outline-width</u>	Sets the width of an outline	thin medium thick <i>length</i> inherit	2

- ❖ Các CSS margin (lề) định nghĩa khoảng trống xung quanh các phần tử.
- ❖ Lề xóa vùng xung quanh một phần tử (bên ngoài đường viền).
- ❖ Lề không có màu nền và hoàn toàn trong suốt.
- ❖ Lề trên, phải, dưới, và trái có thể được thay đổi độc lập sử dụng các thuộc tính riêng.
- ❖ Thuộc tính shorthand cũng có thể được dùng, để thay đổi tất cả lề một lần

# CÁC GIÁ TRỊ CÓ THỂ



Giá trị	Mô tả
auto	The browser sets the margin. The result of this is dependant of the browser
<i>length</i>	Defines a fixed margin (in pixels, pt, em, etc.)
%	Defines a margin in % of the containing element
<ul style="list-style-type: none"><li>• Có thể sử dụng các giá trị âm để chồng lấp nội dung</li></ul>	

❖ Trong CSS, có thể xác định các lề khác nhau cho mỗi phía khác nhau:

❖ **Thí dụ**

- `margin-top:100px;`  
`margin-bottom:100px;`  
`margin-right:50px;`  
`margin-left:50px;`



- ❖ Để làm ngắn mã, có thể xác định tất cả thuộc tính lề trong một thuộc tính. Điều này được gọi là thuộc tính shorthand.
- ❖ Thuộc tính shorthand cho tất cả thuộc tính lề là "margin":
- ❖ **Thí dụ**
  - `margin:100px 50px;`
- ❖ Thuộc tính margin có thể có một tới bốn giá trị.
  - **`margin:25px 50px 75px 100px;`**
    - top margin là 25px
    - right margin là 50px
    - bottom margin là 75px
    - left margin là 100px

### ❖ **margin:25px 50px 75px;**

- top margin là 25px
- right và left margin là 50px
- bottom margin là 75px

### ❖ **margin:25px 50px;**

- top và bottom margins là 25px
- right và left margins là 50px

### ❖ **margin:25px;**

- Tất cả 4 margins là 25px

## TẤT CẢ THUỘC TÍNH CSS MARGIN



Thuộc tính	Mô tả	Giá trị	CSS
<u>margin</u>	A shorthand property for setting the margin properties in one declaration	<i>margin-top</i> <i>margin-right</i> <i>margin-bottom</i> <i>margin-left</i>	1
<u>margin-bottom</u>	Sets the bottom margin of an element	auto <i>length</i> %	1
<u>margin-left</u>	Sets the left margin of an element	auto <i>length</i> %	1
<u>margin-right</u>	Sets the right margin of an element	auto <i>length</i> %	1
<u>margin-top</u>	Sets the top margin of an element	auto <i>length</i> %	1

❖ Các thuộc tính CSS padding định nghĩa khoảng trống giữa đường viền thành phần và nội dung thành phần

## ❖ **Padding**

- Padding xóa một vùng xung quanh nội dung (bên trong) thành phần. Padding bị ảnh hưởng bởi màu nền của thành phần.
- Top, right, bottom, và left padding có thể được thay đổi độc lập sử dụng các thuộc tính riêng rẽ.
- Thuộc tính shorthand padding cũng có thể được dùng để thay đổi tất cả paddings một lần.

❖ Trong CSS có thể padding từ phía khác nhau

❖ Thí dụ:

- padding-top:25px;  
padding-bottom:25px;  
padding-right:50px;  
padding-left:50px;

- ❖ Để là ngắn mã, có thể xác định tất cả thuộc tính padding trong một thuộc tính. Điều này được gọi là thuộc tính shorthand.
- ❖ Thuộc tính shorthand cho tất cả thuộc tính padding là "padding".
- ❖ Ví dụ:
  - padding:25px 50px;

- ❖ Thuộc tính padding có thể có từ một tới bốn giá trị.
- ❖ **padding:25px 50px 75px 100px;**
  - top padding là 25px
  - right padding là 50px
  - bottom padding là 75px
  - left padding là 100px
- ❖ **padding:25px 50px 75px;**
  - top padding là 25px
  - right và left paddings là 50px
  - bottom padding là 75px

### ❖ **padding:25px 50px;**

- top và bottom paddings là 25px
- right và left paddings là 50px

### ❖ **padding:25px;**

- Tất cả 4 paddings là 25px

### ❖ **Thí dụ:**

- Tất cả padding trong một khai báo
- Thiết lập padding-left
- Thiết lập padding-right
- Thiết lập padding-top



## TẤT CẢ THUỘC TÍNH PADDING



Thuộc tính	Mô tả	Giá trị	CSS
<u>padding</u>	A shorthand property for setting all the padding properties in one declaration	<i>padding-top</i> <i>padding-right</i> <i>padding-bottom</i> <i>padding-left</i>	1
<u>padding-bottom</u>	Sets the bottom padding of an element	<i>length</i> <i>%</i>	1
<u>padding-left</u>	Sets the left padding of an element	<i>length</i> <i>%</i>	1
<u>padding-right</u>	Sets the right padding of an element	<i>length</i> <i>%</i>	1
<u>padding-top</u>	Sets the top padding of an element	<i>length</i> <i>%</i>	1

- ❖ Thuộc tính CSS list cho phép đặt bộ đánh dấu thành phần danh sách, thay đổi giữa các bộ đánh dấu khác nhau hay thiết lập hình ảnh như bộ đánh dấu thành phần danh sách
- ❖ Danh sách: Trong HTML, có hai loại danh sách
  - Danh sách không thứ tự
  - Danh sách có thứ tự
- ❖ Với CSS, các danh sách có thể được phân kiểu xa hơn, và hình ảnh có thể được dùng như bộ đánh dấu thành phần danh sách

- ❖ Có thể xác định loại danh sách với thuộc tính list-style-type
- ❖ Thí dụ:
  - `ul.circle {list-style-type:circle}`  
`ul.square {list-style-type:square}`  
  
`ol.upper-roman {list-style-type:upper-roman}`  
`ol.lower-alpha {list-style-type:lower-alpha}`

Giá trị	Mô tả
none	No marker
disc	Default. The marker is a filled circle
circle	The marker is a circle
square	The marker is a square

Giá trị	Mô tả
none	No marker
circle	The marker is a circle
disc	The marker is a filled circle. This is default
square	The marker is a square
armenian	The marker is traditional Armenian numbering
decimal	The marker is a number
decimal-leading-zero	The marker is a number padded by initial zeros (01, 02, 03, etc.)

Giá trị	Mô tả
georgian	The marker is traditional Georgian numbering (an, ban, gan, etc.)
lower-alpha	The marker is lower-alpha (a, b, c, d, e, etc.)
lower-greek	The marker is lower-greek (alpha, beta, gamma, etc.)
lower-latin	The marker is lower-latin (a, b, c, d, e, etc.)
lower-roman	The marker is lower-roman (i, ii, iii, iv, v, etc.)
upper-alpha	The marker is upper-alpha (A, B, C, D, E, etc.)
upper-latin	The marker is upper-latin (A, B, C, D, E, etc.)
upper-roman	The marker is upper-roman (I, II, III, IV, V, etc.)

- ❖ Thuộc tính `list-style-position` xác định căn lề của một danh sách.
- ❖ "outside" là giá trị mặc định. "inside" là giá trị khác để căn lề danh sách
- ❖ **Thí dụ**
  - `ul.inside {list-style-position:inside}`
  - `ul.outside {list-style-position:outside}`

❖ Có thể dùng một ảnh như bộ đánh dấu thành phần danh sách marker:

❖ **Thí dụ**

- `ul`  
`{`  
`list-style-image:url('arrow.gif');`  
`}`



- ❖ Thí dụ trên sẽ không hiển thị kết quả chính xác trong tất cả trình duyệt
- ❖ IE và Opera sẽ hiển thị ảnh cao hơn một chút so với Firefox, Chrome, và Safari.
- ❖ Thí dụ trên sẽ tốt cho hầu hết các trường hợp. Tuy nhiên, có một cách để định vị hình ảnh chính xác hơn
- ❖ Để có cùng kết quả trong tất cả trình duyệt, bạn phải dùng một ảnh nền trên mỗi thành phần danh sách, như thế này:

❖ **Thí dụ**

```
ul
{
list-style-type:none;
padding:0px;
margin:0px;
}
li
{
background-image:url(arrow.gif);
background-repeat:no-repeat;
background-position:0px 5px;
padding-left:14px;
}
```

## ❖ Giải thích thí dụ

## ❖ Đối với ul:

- Đặt list-style-type tới none để bỏ bộ đánh dấu thành phần danh sách
- Cả hai padding và margin phải được đặt tới 0px để tương thích trình duyệt

## ❖ Đối với li:

- Đặt URL của ảnh, và hiển thị nó chỉ một lần (không lặp)
- Dùng thuộc tính background-position để đặt hình nơi bạn muốn (left 0px and down 5px)
- Dùng thuộc tính padding-left để định vị văn bản trong danh sách

- ❖ Có thể xác định tất cả các thuộc tính danh sách trong một thuộc tính đơn dùng thuộc tính shorthand.
- ❖ Thuộc tính shorthand cho danh sách là "list-style":
- ❖ **Thí dụ:**
  - list-style:square inside;
- ❖ Khi dùng thuộc tính shorthand, thứ tự của các giá trị là:
  - list-style-type
  - list-style-position
  - list-style-image
- ❖ Không vấn đề nếu một trong các giá trị bị mất, miễn là phần còn lại trong thứ tự xác định.

## TẤT CẢ THUỘC TÍNH



Thuộc tính	Mô tả	Giá trị	CSS
<u>list-style</u>	Sets all the properties for a list in one declaration	<i>list-style-type</i> <i>list-style-position</i> <i>list-style-image</i> inherit	1
<u>list-style-image</u>	Specifies an image as the list-item marker	URL none inherit	1
<u>list-style-position</u>	Specifies where to place the list-item marker	inside outside inherit	1

Thuộc tính	Mô tả	Giá trị	CSS
<u>list-style-type</u>	Specifies the type of list-item marker	none disc circle square decimal decimal-leading-zero armenian georgian lower-alpha upper-alpha lower-greek lower-latin upper-latin lower-roman upper-roman inherit	1

❖ Thuộc tính css table cho phép thiết lập bố trí bảng

❖ Thí dụ:

- Thiết lập bố trí bảng
- Hiển thị ô trống của bảng
- Phủ lấp đường viền của bảng
- Thiết lập không gian giữa các đường viền
- Thiết lập chú thích cho bảng

## TẤT CẢ THUỘC TÍNH CHO BẢNG



Thuộc tính	Mô tả	Giá trị	CSS
<u><a href="#">border-collapse</a></u>	Specifies whether or not table borders should be collapsed	collapse separate inherit	2
<u><a href="#">border-spacing</a></u>	Specifies the distance between the borders of adjacent cells	<i>length length</i> inherit	2
<u><a href="#">caption-side</a></u>	Specifies the placement of a table caption	top bottom inherit	2
<u><a href="#">empty-cells</a></u>	Specifies whether or not to display borders and background on empty cells in a table	show hide inherit	2
<u><a href="#">table-layout</a></u>	Sets the layout algorithm to be used for a table	auto fixed inherit	2