# CSS

## Độ ưu tiên

* <https://htmlcss.fullstack.edu.vn/?id=0313423e-f68a-46ca-b745-0b6d29fd335b&t=335>
* <https://www.w3schools.com/css/css_specificity.asp>

## Inherent (Kế thừa)

Chú ý tính kế thừa

## CSS CheatSheet

* [CSS selectors cheatsheet](https://css-selectors-cheatsheet.fullstack.edu.vn/)
* [Game CSS Diner](https://css-diner.fullstack.edu.vn/)
* [Game CSS SCOOPS](https://css-scoops.fullstack.edu.vn/)



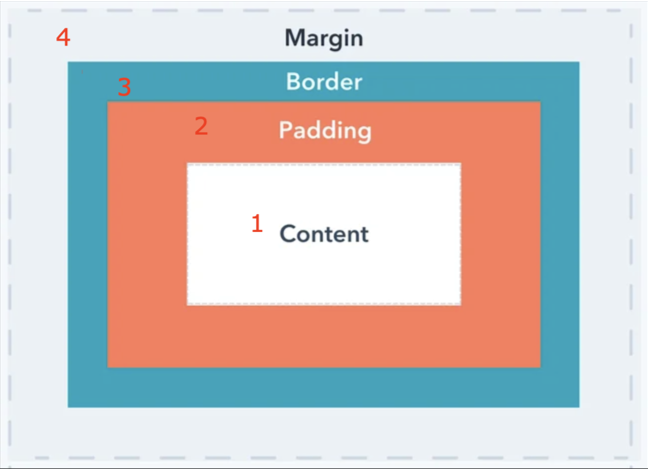
## Box model

* Trình duyệt luôn xem các phần tử là hình chữ nhật.
* Có 4 thành phần trong *Box Model*: *content, padding, border, margin*.
* Có 5 thuộc tính *CSS* trong *Box Model*: width, height, padding, border, margin.
* Kích thước được đặt bằng thuộc tính width và height chưa phải là kích thước thật của một phần tử. Kích thước thật của một phần tử luôn bao gồm: width, height, padding và border.
* *Margin* không tham gia làm tăng kích thước phần tử.

<https://htmlcss.fullstack.edu.vn/?id=24e8499b-994b-4802-9507-5203c06682f2>

### Padding

<https://htmlcss.fullstack.edu.vn/?id=0b2f5ae5-4c03-440e-9929-60aa462c13f8>



selector {

padding: 10px; /\* Đệm cả 4 hướng 10px \*/

padding: 10px 20px; /\* Đệm trên/dưới 10px, trái/phải 20px \*/

padding: 10px 20px 30px; /\* Đệm trên 10px, trái/phải 20px, dưới 30px \*/

padding: 10px 20px 30px 40px; /\* Đệm theo thứ tự: trên-phải-dưới-trái \*/

}

### Border

<https://htmlcss.fullstack.edu.vn/?id=f4b2095c-c115-4f3e-819d-9d145d7b6fce&t=110>

<https://www.w3schools.com/cssref/pr_border-style.asp>

.box {

width: 200px;

height: 200px;

padding: 20px;

background-color: blueviolet;

border: 10px solid pink; ( border-width, border-style ( required ) , border-color)

border-radius: 50%;

}

Width and style, color is four direction settings

border-radius : Chỉ tác động tới hiển thị chứ không tác động tới box-model

Thuộc tính border là cú pháp viết tắt của:

border-width: độ dày đường viền

border-style: kiểu dáng đường viền

border-color: màu sắc đường viền

Thuộc tính border-width lại là cú pháp viết tắt của:

border-top-width: Độ dày viền trên

border-bottom-width: Độ dày viền dưới

border-left-width: Độ dày viền trái

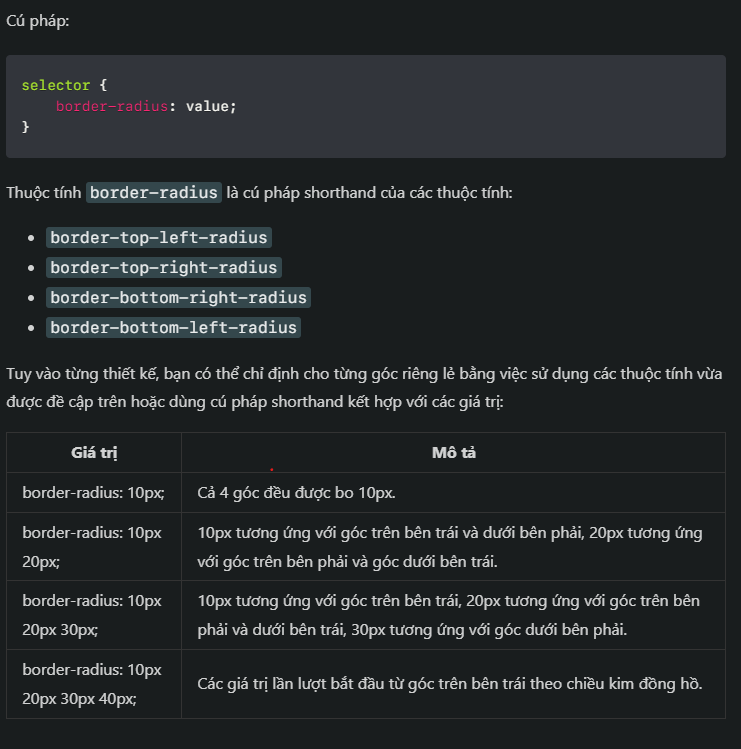
border-right-width: Độ dày viền phải

Các thuộc tính border-width, border-style, border-color đều có thể có từ 1 tới 4 giá trị tương ứng với các hướng: top, right, bottom, left, nên bạn có thể ghi nhớ tương tự với thuộc tính border-style và border-color.

Hướng dẫn làm trích dẫn với tag “blockquote”

<https://htmlcss.fullstack.edu.vn/?id=d5c038ad-5314-4110-ab22-dd2022bfae1e&t=270>

### Border-Radius



### Thuộc tính Outline

Outline tương tự như border là lớp viền. Nhưng nó không va chạm với các phần tử xung quanh như border. Nó ở mặt phẳng z cao hơn. Outline nằm ngoài border và khi không có border thì nó sẽ giống border

outline-offset

Outline cũng có thể sử dụng border-radius

### Margin

Margin là khoảng cách giữa phần tử hiện tại với các phần tử xung quanh nó. Trong Box Model, margin nằm ở vị trí thứ 4 tính từ trong ra ngoài và cũng là vị trí ngoài cùng của Box Model.

<https://htmlcss.fullstack.edu.vn/?id=8cfcfa2b-e332-420b-9866-2330a9f1c031>

margin: 0 auto; // Chỉ căn giữa cho block

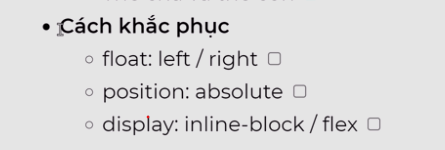
### Margin Collapse

<https://htmlcss.fullstack.edu.vn/?id=0c1db493-5c3d-4559-a7b7-3848860230c5&t=850>

Lấy giá trị lớn hơn

* Xảy ra với các thẻ con cùng cấp và margin ngược chiều
* Xảy ra giữa thẻ con và cha khi margin cùng chiều

Cách khắc phục:



float & width 100% (cùng cấp cả 2 thẻ, cha con 1 thẻ)

position: absolute width: 100% ( cùng cấp 1 thẻ, cha con ->thẻ con)

inline-block (thẻ con, thẻ cha thì thêm width 100%)

flex (thẻ cha: flex-direction: column)

### Box-sizing

* Content-box
* Border-box

## CSS Units

### Absolute units (tuyệt đối)

px (Pixel)

### Relative Units (Tướng đối)

* % : Từ thẻ cha gần nhất chứa nó có propertype font-size
* vw, vh : Phụ thuộc khung nhìn của trình duyệt
* em : Phụ thuộc kích cỡ phông chữ của chính nó
* rem : Phụ thuộc vào font-size của phần tử <html>

|  |  |
| --- | --- |
| **Unit** | **Description** |
| em | Relative to the font-size of the element (2em means 2 times the size of the current font) |
| ex | Relative to the x-height of the current font (rarely used) |
| ch | Relative to width of the "0" (zero) |
| rem | Relative to font-size of the root element |
| vw | Relative to 1% of the width of the viewport\* |
| vh | Relative to 1% of the height of the viewport\* |
| vmin | Relative to 1% of viewport's\* smaller dimension |
| vmax | Relative to 1% of viewport's\* larger dimension |
| % | Relative to the parent element |



Html {

Font-size: 62.5%

}

<https://htmlcss.fullstack.edu.vn/?id=9ed5261e-9f88-47bf-a031-860735889d09&t=485>

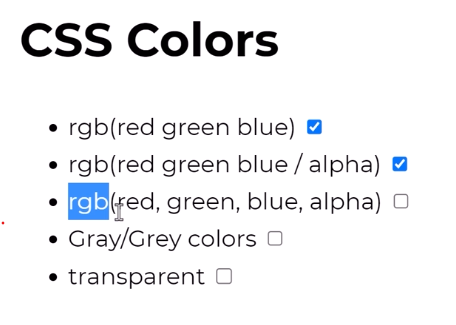
Đơn vị tương đối

* Hỗ trợ tốt cho giao diện trên nhiều kích thước màn hình.
* Sử dụng rem cho kích thước phông chữ, margin, padding, v.v
* Sử dụng % khi đặt chiều rộng cho phần tử.
* <https://www.javascriptwillrule.com/css-units-responsiveness-without-adding-media-queries>
* <https://www.7io.co/blog/why-we-use-rem-instead-of-px/>
* <https://www.sitepoint.com/understanding-and-using-rem-units-in-css/>
* <https://www.freecodecamp.org/news/what-is-rem-in-css/>

## Color

<https://fullstack.edu.vn/html-colors-tool>

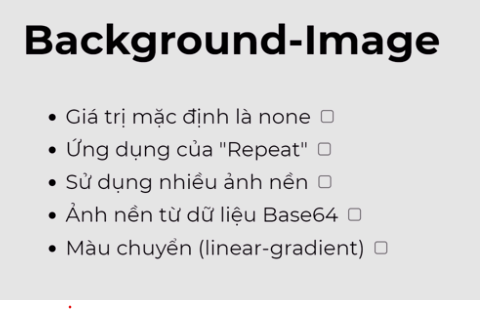
<https://www.rapidtables.com/convert/number/decimal-to-hex.html>

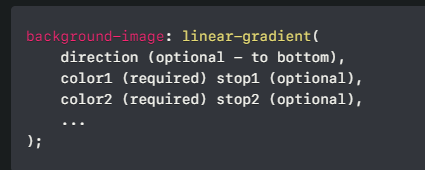


## Background

### Background-color

### Background-Image







* Linear-gradient



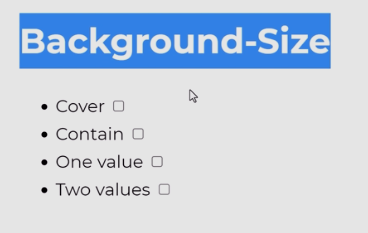
<https://cssgradient.io/>

Tài nguyên sử dụng trong bài học:

<https://cssgradient.io/gradient-backgrounds/>

<https://www.w3schools.com/cssref/func_linear-gradient.php>

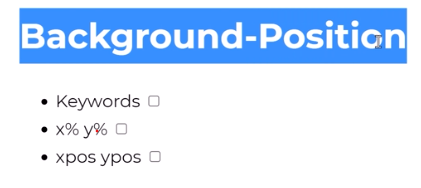
### Background-Size



<https://htmlcss.fullstack.edu.vn/?id=6c0a296e-9cc9-4e6e-8522-ace4e99e8485>

<https://htmlcss.fullstack.edu.vn/?id=6c0a296e-9cc9-4e6e-8522-ace4e99e8485&t=150>

### Background-position



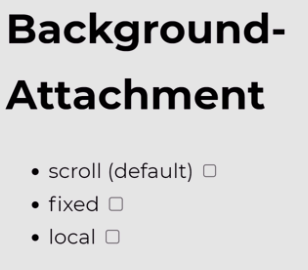
* top center, left center, right center, center center, left bottom, …

<https://htmlcss.fullstack.edu.vn/?id=20b6e6af-cfa6-4483-b01d-e947425f1155>

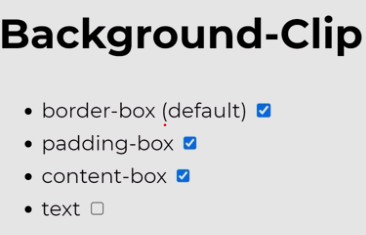
* 100% 0%
* 50px 50px
* Khi dùng 1 con số thì cái thứ 2 luôn là center

### Background-Attachment

<https://htmlcss.fullstack.edu.vn/?id=a7ec84a7-fcbf-4e1c-8595-a417cc3da431>



### Background-clip

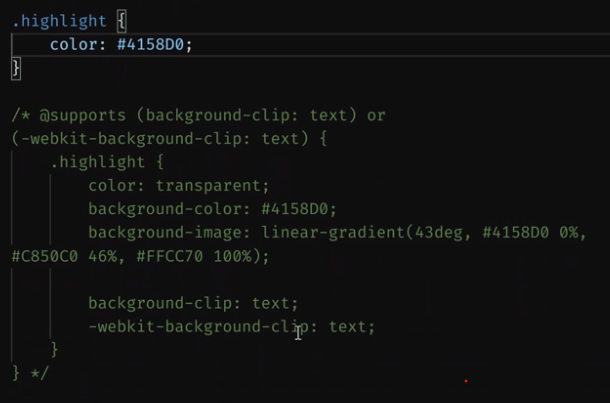


Thuộc tính background-clip quyết định màu nền/ảnh nền trong một phần tử được đổ từ thành phần nào.

Bao gồm 4 giá trị:

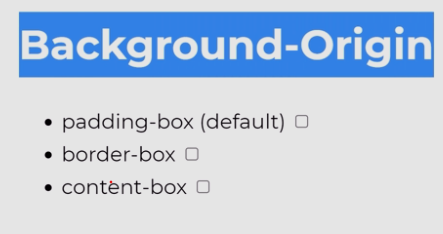
* border-box là giá trị mặc định. Ảnh nền/màu nền sẽ được đổ từ border vào
* padding-box sẽ đổ ảnh nền/màu nền từ padding vào
* content-box sẽ chỉ được đổ ở content
* text sẽ chỉ hiển thị được ở phần text

Làm chữ chuyển màu



### Background-origin

<https://htmlcss.fullstack.edu.vn/?id=ee50b333-4980-47f6-9ff9-6d4e38ec7932&t=350>



Quyết định gốc của ảnh nền sẽ hiển thị từ phần nào của phần tử và có tác động tới background-position

### Background shorthands

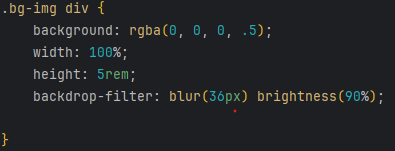
<https://www.w3schools.com/CSSref/css3_pr_background.php>

background: bg-color bg-image position/bg-size bg-repeat bg-origin bg-clip bg-attachment initial|inherit;

### Backdrop-filter

* <https://developer.mozilla.org/en-US/docs/Web/CSS/backdrop-filter>
* <https://www.w3schools.com/cssref/css3_pr_backdrop-filter.php>

Thêm hiệu ứng đồ họa cho phần tử



## Làm việc với Font



* Chủng font Serif:

Các font chữ thuộc chủng Serif là các font chữ có chân. Các font chữ phổ biến thuộc chủng Serif bao gồm: Times New Roman, Georgia, Garamond.

Xét về nguyên lý thiết kế, font chữ có chân giúp định hình mắt, hạn chế mỏi mắt khi đọc nhiều. Thường được dùng nhiều trong sách, báo, các bài blog, v.v.

* Chủng font San-Serif:

Font chữ thuộc chủng San-Serif là font chữ không có chân. Các font chữ phổ biến thường là: Arial, Verdana, Garamond, v.v. Chủng font San-Serif được xem là phổ biến nhất và được dùng trên hầu hết các trang web.

* Chủng font Monospace:

Các font chữ thuộc chủng Monospace thường sẽ có độ rộng các ký tự bằng nhau. Ví dụ như font Monaco thuộc chủng Monospace thường được dùng rất nhiều trong các trình viết code.

* Họ phông chữ được chia làm 2 loại:
* Họ phông theo tên - family-name: Arial, Helvetica, Times New Roman, v.v.
* Họ phông chữ chung - generic-family: serif, sans-serif, cursive, monospace, v.v.

### System font

* System fonts (*fonts chữ hệ thống*) là các fonts chữ đã được cài sẵn trên các thiết bị, hệ điều hành.
* Ưu điểm khi sử dụng: nhanh, an toàn và mang lại trải nghiệm quen thuộc.
* Để sử dụng system fonts, thêm font-family như sau vào selector body:
* /\* System Fonts with system-ui \*/
* body {
* font-family: system-ui, "Segoe UI", Roboto, Helvetica, Arial, sans-serif,
* "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol";

}

Có thể tìm kiếm Google với từ khóa “system fonts css” hoặc tham khảo ở các trang như:

* System Font Stack - css-tricks.com
* System Font Stack- digitalocean.com
* System Font Stack- https://systemfontstack.com/
* Thêm font-family như sau vào selector body:

/\* System Fonts with system-ui \*/

body {

font-family: system-ui, "Segoe UI", Roboto, Helvetica, Arial, sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol";

}

### Typography



<https://cssreference.io/typography/>

<https://www.w3schools.com/css/css_font_shorthand.asp>

* line-height:

Không nên dùng đơn vị tuyệt đối nên dùng số phụ thuộc font-size

1 = font-size

En: 1.4

Vi: 1.5 – 1.6

Chữ luôn nằm giữa line-height nên có thể ứng dụng căn chữ giữa 1 phần tử

* Font-style

Normal, italic, …

### Thuộc tính font – shorthand

font: [font-style] [font-variant] [font-weight] [font-size/line-height] [font-family];

Lưu ý:

* Thứ tự của các giá trị không nhất thiết phải theo đúng cú pháp, nhưng cần tuân theo thứ tự từ trái sang phải.
* Hai thuộc tính font-size và font-family bắt buộc phải có khi sử dụng shorthand, nếu thiếu trình duyệt sẽ sử dụng giá trị mặc định. Các thuộc tính khác không bắt buộc.
* Giữa giá trị của thuộc tính font-size và line-height cần có dấu /.

**Tổng kết**

* Trong CSS, thuộc tính line-height được sử dụng để điều chỉnh khoảng cách giữa các dòng chữ.
* Thuộc tính font-style dùng để xác định kiểu chữ của văn bản. Cho phép bạn chỉ định văn bản sẽ được hiển thị bằng chữ in nghiêng, chữ bình thường, v.v.
* Thuộc tính font-variant dùng để xác định cách hiển thị chữ hoa và chữ thường trong văn bản.
* Thuộc tính font-weight được sử dụng để điều chỉnh độ dày/mỏng của văn bản.
* Thuộc tính font trong CSS là cú pháp shorthand của: font-style, font-variant, font-weight font-size/line-height, font-family.

### Làm việc với kiểu chữ

Thuộc tính text-align căn nội dung chữ theo độ rộng của thẻ. Trong trường hợp sử dụng text-align cho các thẻ inline, bạn sẽ thấy không hoạt động vì độ rộng thẻ inline chỉ bằng nội dung mà nó chứa.

Để căn được nội dung của các thẻ inline, cần set cho thẻ đó có display: block.

Khi sử dụng text-align cho thẻ cha, các thẻ con là thẻ inline hoặc inline-block ở bên trong cũng sẽ được căn theo.

* Thuộc tính letter-spacing được sử dụng để điều chỉnh khoảng cách giữa các ký tự. Giá trị càng lớn khoảng cách giữa các ký tự càng xa nhau.
* Thuộc tính text-align dùng để căn nội dung theo các hướng như: căn trái, căn phải, căn giữa, v.v. Mặc định, nội dung đã được căn trái.
* Trong *CSS*, text-decoration được sử dụng để thêm gạch chân chữ hoặc bỏ gạch chân chữ.
* Thuộc tính text-indent được sử dụng để thụt lề dòng đầu tiên trong một đoạn văn bản.
* Thuộc tính text-transform được sử dụng để chuyển đổi từ chữ in hoa thành chữ thường và ngược lại.
* Thuộc tính text-transform được sử dụng để chuyển đổi từ chữ in hoa thành chữ thường và ngược lại.

Thuộc tính text-transform được sử dụng để chuyển đổi từ chữ in hoa thành chữ thường và ngược lại.

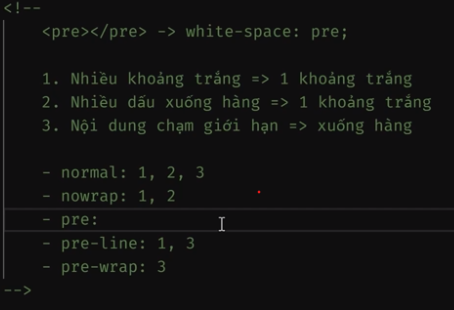
Bao gồm các giá trị:

* none là giá trị mặc định, văn bản hiển thị như văn bản gốc.
* uppercase biến đổi tất cả các chữ thành chữ in hoa.
* lowercase biến đổi tất cả các chữ thành chữ thường.
* capitalize biến đổi chữ cái đầu của mỗi từ thành chữ in hoa, các chữ cái còn lại ở dạng chữ thường.

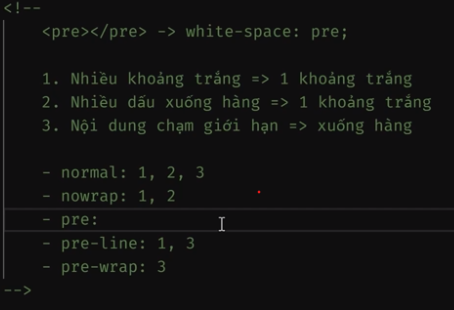
### Làm việc với chữ tiếp theo



* White-space



* Text-shadow



Thuộc tính text-shadow bao gồm 4 giá trị:

* Giá trị đầu tiên sẽ quyết định bóng đổ sang bên trái hay bên phải. Nếu giá trị âm sẽ đổ sang trái, giá trị dương sẽ đổ sang phải.
* Giá trị thứ 2 quyết định bóng đổ lên trên hay xuống dưới. Giá trị âm sẽ đổ lên trên, giá trị dương sẽ đổ xuống dưới.
* Giá trị thứ 3 quyết định độ nhòe (blur) của bóng đổ.
* Giá trị cuối cùng là màu của bóng đổ.

<https://cssgenerator.org/text-shadow-css-generator.html>

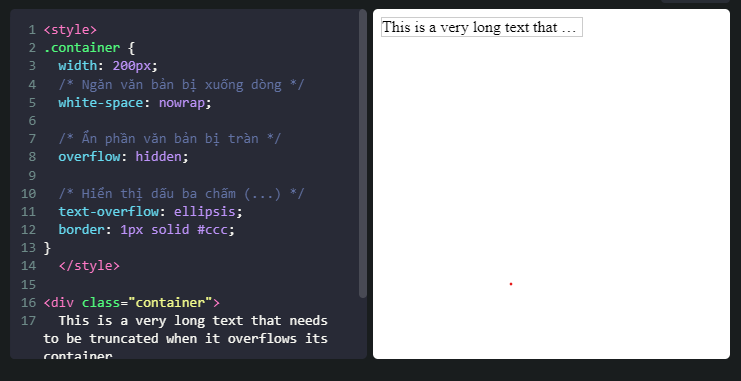
* text-overflow

Khi nội dung một phần tử có kích thước lớn vượt quá thẻ chứa thì sẽ bị tràn ra ngoài, tiếng Anh gọi là overflow.

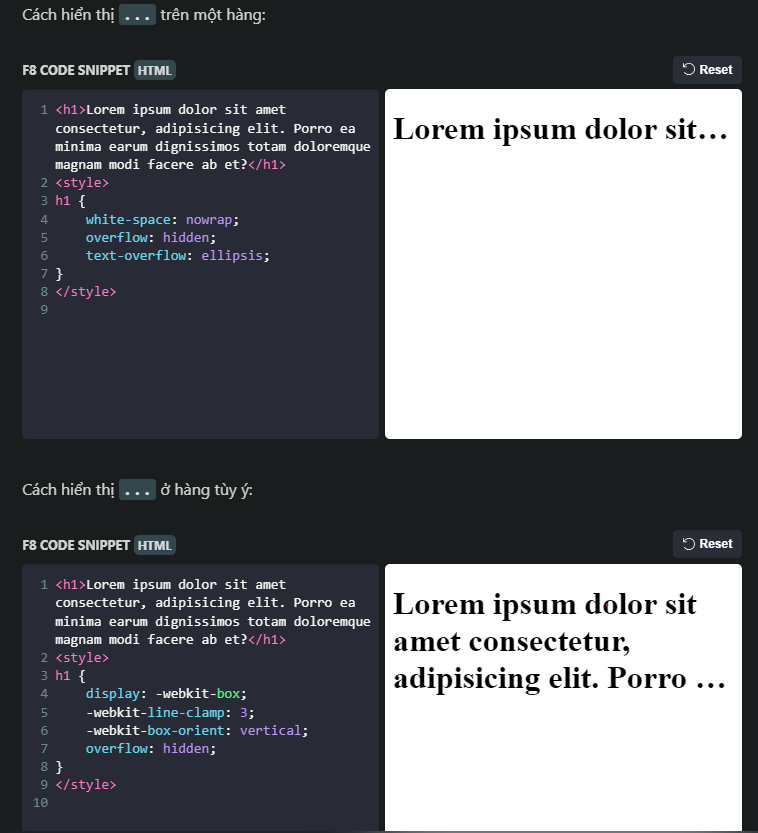
Thuộc tính text-overflow giúp bạn quyết định một nội dung văn bản được hiển thị như nào khi nó vượt quá thẻ cha chứa nó.  
Có thể sử dụng 2 giá trị:

* clip để cắt bỏ nội dung bị tràn ra ngoài
* ellipsis để hiển thị dấu ... ở cuối văn bản khi bị tràn

Lưu ý, khi sử dụng thuộc tính text-overflow, bắt buộc phải sử dụng thêm 2 thuộc tính:white-space: nowrap và overflow: hidden.



* Xử lý đấu 3 chấm trên hàng tùy ý



**Tóm tắt**

* Khi các thẻ có nhiều khoảng trắng, trình duyệt chỉ hiển thị 1 khoảng trắng (ngoại từ thẻ <pre>.
* Khi có nhiều dấu xuống hàng, trình duyệt chỉ hiển thị thành 1 khoảng trắng.
* Khi nội dung dài và chạm giới hạn, trình duyệt sẽ tự động cho xuống hàng.
* Thuộc tính white-space được sử dụng để xác định cách xử lý khoảng trắng (hoặc dấu cách) bên trong một phần tử HTML.
* Bao gồm 5 giá trị: normal, nowrap, pre, pre-line, pre-wrap.
* Thuộc tính word-break dùng để ngắt nội dung xuống hàng trong trường hợp nội dung quá dài và không tự xuống hàng. Nội dung chỉ được tự động xuống hàng khi gặp khoảng trắng hoặc dấu -.
* Trong CSS, word-spacing được sử dụng để tăng/giảm khoảng cách giữa các từ.
* Thuộc tính text-shadow được sử dụng để tạo ra bóng đổ cho nội dung chữ. Ví dụ: text-shadow: 4px 4px 2px rgba(0,0,0,0.6).
* Giá trị đầu tiên sẽ quyết định bóng đổ sang bên trái hay bên phải.
* Giá trị thứ 2 quyết định bóng đổ lên trên hay xuống dưới.
* Giá trị thứ 3 quyết định độ nhòe (blur) của bóng đổ.
* Giá trị cuối cùng là màu của bóng đổ.
* Thuộc tính text-overflow giúp bạn quyết định một nội dung văn bản được hiển thị như nào khi nó vượt quá thẻ cha chứa nó. Bao gồm 2 giá trị: clip và ellipsis.
* Lưu ý, khi sử dụng thuộc tính text-overflow, bắt buộc phải sử dụng thêm 2 thuộc tính:white-space: nowrap và overflow: hidden.

## Làm việc với hình ảnh

### Đặt kích thước tối đa cho hình ảnh

Max-width: 100%;

### Căn giữa hình ảnh

Tùy trường hợp: VD img được bao bởi thẻ figure thì có thể dùng text-align: center ở thẻ cha

Nhưng nếu bao bởi thẻ p thì tác động thẻ p có thể ảnh hưởng tới cấu trúc văn bản khi đó cần css cho img -> display: block; margin: 0 auto; (Vì thẻ cha p là thẻ block nên chiều rộng mặc định là 100%)

### Lazy load cho hình ảnh

Tăng tốc độ truy cập trang, giảm bằng thông chi phí cho máy chủ: Áp dụng cho các hình ảnh nằm xa khung hình hiển thị đầu tiên

Loading: lazy

### Giữ đúng tỉ lệ hình ảnh

Chỉ đặt 1 chiều width hoặc height (Phần lớn là đặt width)

Tuy nhiên khi ảnh bị lỗi hoặc mạng bị chậm => ta nên đặt 1 min chiều cao cho nó

Width: 100%;

Height: 350px;

Object-fit: cover;

Object-position: top center; /\* Thường để mặc định \*/

Thường được dùng để làm avatar chẳng hạn

### Bọc văn bản xung quanh hình ảnh

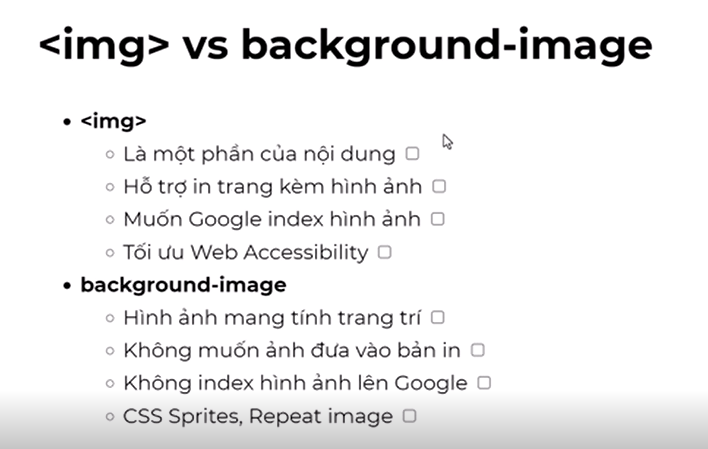
Có 1 thẻ bọc, trong có img và 1 nội dung: h3, p, ….

Img {

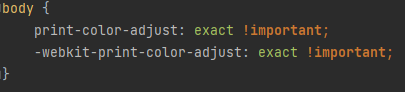
Float: left; /\* thẻ img trôi sang trái và anh em của nó sẽ bám sát nó nếu để right thì thường chữ cũng nên căn sang phải cho đẹp \*/

}

### Dùng img hay background-image?



### Hỗ trợ in background-image



## Thuộc tính vị trí

<https://htmlcss.fullstack.edu.vn/?id=c1535592-9ba3-433a-883b-c4efa5a03466>



* Thuộc tính position dùng để căn chỉnh vị trí hiển thị của phần tử trên trang web. Bao gồm các giá trị: static, relative, absolute, fixed, sticky.
* **Flow Layout** là bố cục luồng. Mặc định, các thẻ *HTML* sẽ luôn nằm trong **Flow Layout**.
* **Positioned** nghĩa là được định vị. Một phần tử được định vị khi phần tử đó được tác động bởi các thuộc tính: top, right, bottom, left.
* Sử dụng static, phần tử sẽ nằm trong **Flow Layout** nhưng sẽ không được định vị.
* Với position: relative, phần tử sẽ nằm trong bố cục luồng và được định vị theo vị trí ban đầu của nó.

### Position Relative

Nằm trong luồng và được định vị theo vị trí ban đầu của nó

### Position absolute

Được định vị nhưng không nằm trong bố cục luồng nữa. Nó mất thuộc tính block mặc định phải chủ động set width cho nó

* Nếu không có thẻ cha được định vị: Sẽ định vị theo vị trí của thẻ body
* Có thẻ cha được định vị: định vị theo thẻ cha gần nhất được định vị chứa nó

### Position Fixed

Không nằm trong flow layout nên không va chạm với các phần tử khác

Được định vị theo viewport và cố định không bị theo thanh cuộn

Top, right, left, bottom không dùng cả 4 với width, height được

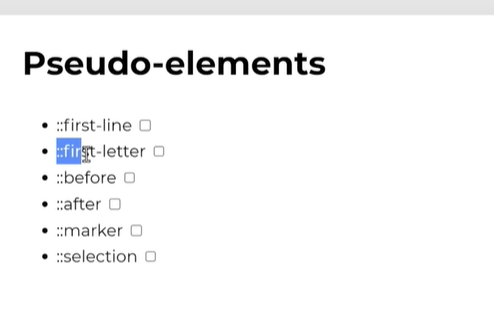
Inset: 0 0; tương tự margin, padding

### Position sticky

Nằm trong flow layout và có positioned, được định vị theo phần tử cha gần nhất có thanh cuộn ( scroll bar ) (Gần giống relative + fixed khi tới giới hạn)

Thường dùng cho thanh điều hướng

## Tìm hiểu pseudo-elements



### first-line, first-letter

first-line: Dùng với thẻ block, inline-block

first-letter

Chỉ hoạt động với 1 số thuộc tính CSS

### selection, marker

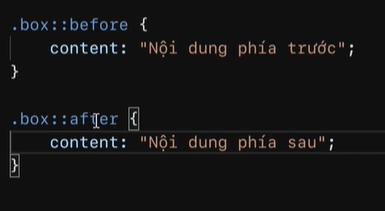
selection: chọn ra đoạn bôi đen

marker: list style thẻ li

Chỉ hoạt động với 1 số thuộc tính CSS

### before, after

Được dùng phổ biến nhất. Giúp chèn thêm nội dung vào phần tử



* Selector ::before được sử dụng để thêm một phần tử giả lên trước một phần tử.
* Selector ::after được sử dụng để thêm một phần tử giả vào sau phần tử.
* Để tạo một phần tử giả bằng ::before hoặc ::after, cần sử dụng thuộc tính content.
* Vì mặc định display của ::before, ::after là inline, nên muốn đặt width/height cần thêm display: inline-block hoặc display: block.

## Tìm hiểu overflow, scrollbar

* Sử dụng thuộc tính overflow để xử lý nội dung bị tràn ra bên ngoài.
* Các giá trị bao gồm: visible | hidden | scroll | auto.
  + visible: có thể nhìn thấy được
  + hidden: ẩn phần bị tràn
  + scroll: luôn hiển thị thanh cuộn
  + auto: chỉ hiển thị thanh cuộn khi nội dung bị tràn
* Thuộc tính overflow chỉ hoạt động với thẻ *block* và được chỉ định chiều cao cụ thể.

Trong trường hợp muốn chỉ định rõ ràng thanh cuộn ngang hoặc thanh cuộn dọc cho trang web, bạn có thể tự chỉ định bằng 2 thuộc tính:

overflow-y: scroll - hiển thị thanh cuộn dọc

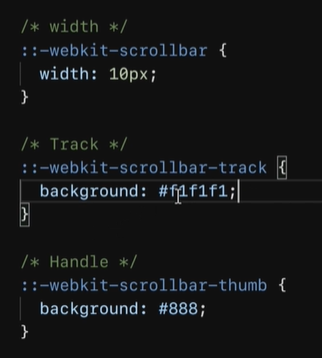
overflow-x: scroll - hiển thị thanh cuộn ngang

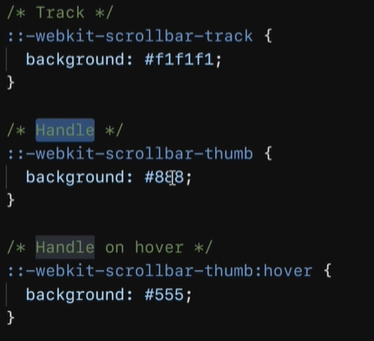
### overscroll-behavior

* *Cuộn trong cuộn* là khi có một thanh cuộn nằm trong một thanh cuộn khác.
* Cuộn chuỗi (*scroll chaining*) là khi bạn cuộn hết nội dung ở thanh cuộn bên trong, nội dung ở thanh cuộn bên ngoài sẽ được cuộn theo.
* Hiệu ứng cao su là hiệu ứng cho phép người dùng refresh lại trang khi cuộn đến cuộn tới đầu trang/cuối trang (chạm tới giới hạn).
* Thuộc tính overscroll-behavior có 3 giá trị:
  + auto là giá trị mặc định, làm xuất hiện tình trạng cuộn chuỗi
  + contain để loại bỏ tình trạng cuộn chuỗi
  + none để loại bỏ tình trạng cuộn chuỗi và hiệu ứng cao su

### Custom Scrollbar

Chỉ dùng được trên windows





### Overlay scrollbar (Thanh cuộn nổi lên)

html, body {

overflow: overlay;

}

::-webkit-scrollbar {

border-radius: 0;

width: 8px;

}

::-webkit-scrollbar-thumb {

border-radius: 4px;

background-color: rgba(22, 24, 35, .2);

}

::-webkit-scrollbar-track {

border-radius: 0;

background-color: rgba(0, 0, 0, 0);

}

### Box-shadow

<https://cssgenerator.org/>

Trong CSS, thuộc tính box-shadow dùng để tạo bóng đổ cho các phần tử.

Bao gồm 5 giá trị:

* Giá trị đầu tiên sẽ quyết định bóng đổ sang bên trái hay bên phải.
* Giá trị thứ 2 quyết định bóng đổ lên trên hay xuống dưới.
* Giá trị thứ 3 quyết định độ nhòe (blur) của bóng đổ.
* Giá trị thứ 4 quyết định độ phủ rộng.
* Giá trị cuối cùng là màu của bóng đổ.

<https://box-shadow.dev/>

<https://getcssscan.com/css-box-shadow-examples>

* Inset: Đổ vào bên trong
* Multi box-shadow
* Mặc định bóng sẽ được đổ bên ngoài phần tử. Giá trị inset trong thuộc tính box-shadow giúp bạn có thể đổ bóng bên trong phần tử.
* Có thể tạo nhiều bóng đổ (*multiple shadows*) cùng lúc chỉ bằng 1 thuộc tính box-shadow.
* Trong thuộc tính box-shadow, mỗi bóng đổ cách nhau một dấu phẩy. Sau mỗi dấu phẩy sẽ là một bóng đổ khác.

### Pointer-events

* Thuộc tính pointer-events quyết định một phần tử phản ứng với sự kiện của con trỏ chuột hay không.
* Bao gồm giá trị: auto và none.
* Sử dụng auto, phần tử sẽ phản ứng với các sự kiện như: di chuột, click.
* Sử dụng giá trị none, phần tử sẽ không phản ứng với sự kiện của chuột nữa. Được ứng dụng khi muốn vô hiệu hóa một phần tử, không cho người dùng tương tác với phần tử đó trên trang web.

### Mix-blend-mode

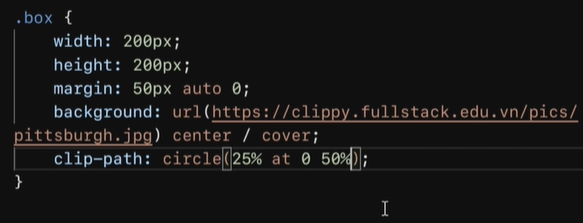
### CSS Counter

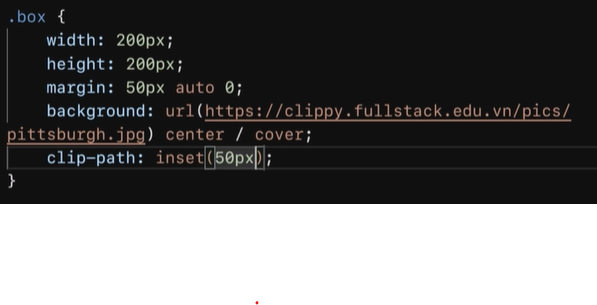
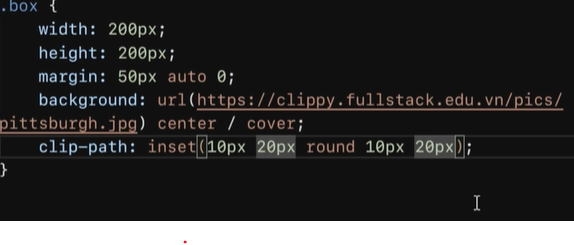
<https://www.w3schools.com/css/css_counters.asp>

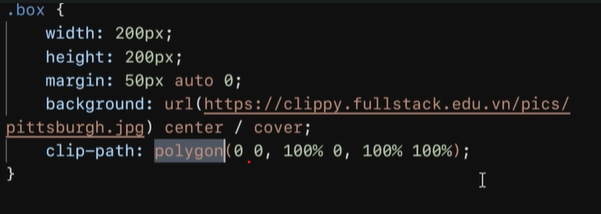
### clip-path



<https://clippy.fullstack.edu.vn/>





### Calc