**TỔNG HỢP KIẾN THỨC TRONG CODE WEB**

## 1.Thẻ Meta Viewport

Thẻ meta Viewport hướng dẫn trình duyệt cách điều chỉnh trang theo chiều rộng cửa từng thiết bị, khi thiếu thẻ meta Viewport các trình duyệt trên thiết bị điện thoại sẽ hiển thị các trang web có cài đặt màn hình mặc định

## 2.Thẻ ****Responsive Web****

## **Responsive Web hay còn gọi là thiết kế Web đáp ứng là cách tiếp cận trung vào môi trường của nguời dùng trang web ,** responsive web, máy chủ luôn gửi cùng một bộ code HTML đến tất cả các thiết bị và chỉ sử dụng CSS để thay đổi cách hiển thị của trang trên các thiết bị khác nhau.

## 3.Thẻ Overflow scroll

## Có thể có lúc bạn có nội dung tràn lớn, bị tràn khỏi khung nhìn.... hãy sử dụng overflow scroll.

* overflow-y: scroll; là thành phần chính của công thức này. Bằng cách chỉ định overflow-y, nó sẽ cắt nội dung ở trên và dưới. Giá trị scroll là để thêm thanh cuộn làm tràn trục hoành.
* Mặc dù bạn có thể nghĩ rằng overflow-y là đủ, nhưng bạn cũng phải yêu cầu trình duyệt không xuống hàng các phần tử con bằng thuộc tính white-space: nowrap;

## 4.Thẻ Media CSS

## [@media](https://viblo.asia/u/media) có thể được sử dụng để thay đổi kiểu dáng trang web dựa trên truy vấn thông tin về thiết bị và hai cách tiếp cận để giải quyết bố cục trang trong thiết kế đáp ứng.

## Media CSS là một "công nghệ" được giới thiệu trong Css3. Nguyên tắc nó sử dụng trên thông qua các thông số kích thước màn hình được khai báo thông qua [@media](https://viblo.asia/u/media) Với công cụ này chúng ta có thể phân đoạn Css chúng ta ra nhiều phần khác nhau tương ứng với kích thước của các loại thiết bị. Thuộc tính này có cú pháp như sau:

@media not|only mediatype and (media feature) {

CSS-Code;

## }

Trong đó mediatype gồm các thuộc tính hay sử dụng sau:

* all: Dùng cho mọi thiết bị
* print: Dùng cho máy in
* screen: Dùng cho máy tính và các thiết bị smart phone

## 5.Thẻ Box-sizing

## Thuộc tính ****box-sizing**** là một thuộc tính rất quan trọng trong việc thiết kế giao diện cho trang web, nó giúp chúng ta "đơn giản hóa" việc xác định kích thước của phần tử khi xây dựng bố cục trang.

|  |  |
| --- | --- |
| content-box | - Chiều rộng (chiều cao) được thiết lập từ thuộc tính width (height) chỉ là chiều rộng (chiều cao) của phần "nội dung" của phần tử. |
| border-box | - Chiều rộng (chiều cao) được thiết lập từ thuộc tính width (height) chính là chiều rộng (chiều cao) của nguyên cả phần tử. |
| initial | - Sử dụng giá trị mặc định của nó.  (mặc định thì thuộc tính box-sizing có giá trị là content-box) |
| Inherit | - Kế thừa giá trị thuộc tính box-sizing từ phần tử cha của nó. |

## 6.Thẻ Position

## Hiểu về thuộc tính position trong css – Khi bạn thiết kế các layout cho website việc chuyển từ file thiết kế theo định dạng HTML (Hypertext Markdown Language) bạn sẽ buộc phải sử dụng thuộc tính position trong [CSS](https://topdev.vn/viec-lam-it/css-kt3). Ví dụ như làm thanh tính năng trên menu, header, v.v. Có thể hiểu nó là thuộc tính xác định loại của phương pháp định vị trí cho thành phần.

## Với thuộc tính position chúng ta có 5 giá trị chính cần biết:

**Static:** Đây được xem là giá trị hiển thị Position trong css một cách mặc định (default), các thành phần sẽ nằm theo thứ tự của văn bản**.**

**Relative:**Định vị trí tuyệt đối cho các thành phần, không gây ảnh hưởng tới vị trí ban đầu hay các thành phần khác.

**Absolute:**Giá trị này sẽ giúp định vị trí tuyệt đối cho thành phần theo thành phần bao ngoài, hoặc ít nhất là theo cửa sổ trình duyệt. (**Fixed:**Định vị trí tương đối cho thành phần theo cửa sổ trình duyệt).

**Fixed:**Định vị và giúp cho phần tử luôn cố định một chỗ, ví dụ như khi bạn scroll trình duyệt chẳng hạn, phần tử sẽ không thay đổi.

**Inherit:**Xác định thừa hưởng thuộc tính từ thành phần cha (thành phần bao ngoài).

## 7.Thẻ Content

## Thuộc tính content sử dụng kèm với bộ chọn [:before](https://hocwebchuan.com/reference/cssSection/se_before.php), [:after](https://hocwebchuan.com/reference/cssSection/se_after.php) để chèn nội dung được tạo.

## 

## 8.Thẻ ****Transition****

## **Thuộc tính Transition trong CSS** được sử dụng khá rộng rãi trong thiết kế web để tạo ra các hiệu ứng chuyển đổi đẹp mắt trên website một cách dễ dàng.

Transition hoạt động bằng cách thay đổi giá trị thuộc tính một cách trơn tru từ giá trị này sang giá trị khác trong khoảng thời gian nhất định. Các tham số thường được sử dụng:

* **transition-delay:** khoảng thời gian dừng cho mỗi hiệu ứng chuyển đổi.
* **transition-duration:** khoảng thời gian chuyển đổi diễn ra.
* **transition-property:** thuộc tính cần chuyển đổi.
* **transition-timing-function:** tốc độ chuyển đổi diễn ra

## Tốc độ chuyển đổi

**Thuộc tính transition-timing-function** dùng để xác định tốc độ thay đổi khi chuyển đổi.

Các giá trị có sẵn như sau:

* **ease**: tạo hiệu ứng chuyển đổi khi bắt đầu thì chậm sau đó nhanh dần và gần kết thúc lại chậm từ từ (giá trị mặc định).
* **linear:** tạo hiệu ứng chuyển đổi từ lúc bắt đầu với lúc kết thúc tốc độ là như nhau.
* **ease-in:** tạo hiệu ứng chuyển đổi chậm ở lúc bắt đầu.
* **ease-out:** tạo hiệu ứng chuyển đổi chậm ở lúc kết thúc.
* **ease-in-out:** tạo hiệu ứng chuyển đổi chậm ở lúc bắt đầu và lúc kết thúc.

## 8.Thẻ ****Transform****

## Thuộc tính transform xác định một chuyển đổi 2 chiều, 3 chiều, có thể là xoay, tỷ lệ, di chuyển, nghiêng, ...

## 9.Thẻ Webkit

Đây là những tiền tố được cung cấp bởi các engine render khác nhau:

1. -webkit là của Chrome và Safari
2. -moz là của Firefox
3. -o là của Opera
4. -ms là của Internet Explorer

## 10.Thẻ Shadow

2 hiệu ứng được sử dụng nhiều nhất:

* **text-shadow** dành cho đoạn text.
* **box-shadow** dành cho phần tử dạng box.

## Text Shadow

Thuộc tính text-shadow sử dụng để tạo bóng cho các phần tử ở dạng text giúp văn bản hiển thị nổi bật tương tự kiểu chữ 3D.

### Cú pháp:

text-shadow: h-shadow v-shadow blur-radius color

Text-shadow gồm 4 tham số:

* **h-shadow:** vị trí bóng đổ theo chiều ngang, số âm đẩy bóng lên phía trên, số dương đẩy bóng xuống dưới (bắt buộc).
* **v-shadow:** vị trí bóng đổ theo chiều dọc, số âm đẩy bóng lùi ra sau, số dương đẩy bóng về trước (bắt buộc).
* **blur-radius:** độ nhòe của bóng (tùy chọn).
* **color:** màu sắc của bóng (tùy chọn).

## Box Shadow

**Thuộc tính box-shadow** sử dụng để tạo bóng cho các phần tử có dạng box, có tác dụng với các đường viền, giúp phần tử hiển thị nổi bật tương tự kiểu hộp 3D.

### Cú pháp box-shadow

box-shadow: h-offset v-offset blur spread color

## Box-shadow gồm 5 tham số:

* **h-offset:** vị trí bóng đổ theo chiều ngang, số dương đẩy bóng xuống dưới, số âm đẩy bóng lên phía trên (bắt buộc).
* **v-offset:** vị trí bóng đổ theo chiều dọc, số dương đẩy bóng về trước, số âm đẩy bóng lùi ra sau (bắt buộc).
* **spread:** kích thước của bóng (tùy chọn).
* **blur:** độ nhòe của bóng (tùy chọn).
* **color:** màu sắc của bóng (tùy chọn).

# **11.Thẻ object-fit**

## Với object-fit chúng ta có thể giải quyết như thế nào một hình ảnh (hoặc video) kéo dài hoặc co lại nó vào trong box. Điều này là cần thiết khi một bức ảnh chúng ta phải có một kích thước hoặc tỉ lệ khác nhau từ nơi chúng ta đã thiết kế cho nó trong một bố cục.

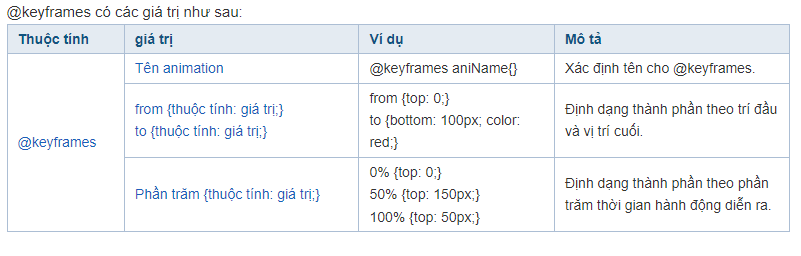
* fill (default): Chiều rộng và chiều cao của hình ảnh đầy box. Hầu hết các cái này sẽ mess lên tỉ lệ.
* cover : Những hình ảnh giữ tỉ lệ của nó và lấp đầy toàn bộ box, các bộ phận của nó thừa ra sẽ được cắt đi và sẽ không được hiển thị.
* contain : Những hình ảnh giữ nó tỉ lệ và phóng to / thu nhỏ lại để phù hợp với bên trong box.
* none : Lấy kích thước gốc và tỉ lệ.
* scale-down : hiển thị như 1 trong số những cái ở trên(fill,cover,contai...), phụ thuộc vào cái nào tạo ra size ảnh nhỏ nhất.

# **12.Thẻ Keyframes**

Thuộc tính @keyframes dùng để điều khiển diễn biến một hoạt động của thành phần, được dùng kèm với thuộc tính [animation](https://hocwebchuan.com/reference/cssSection/css3/pr_animation.php).

**Trong đó:**

* @-moz-keyframes hỗ trợ cho firefox.
* @-webkit-keyframes hỗ trợ cho Google Chrome và Safari.



# **13.Các ký tự**

**a. \***

Biểu tượng ngôi sao sẽ nhắm chọn mọi phần tử trên trang. Nhiều nhà phát triển sẽ sử dụng thủ thuật này để loại bỏ margin và padding. Mặc dù đây là công cụ để nhanh chóng kiểm tra, nhưng tôi muốn khuyên bạn đừng bao giờ sử dụng nó trong code của sản phẩm cuối. Nó thêm quá nhiều gánh nặng cho trình duyệt, và là không cần thiết.

# **13.Thẻ label**

* Tag <label> định nghĩa nhãn cho thành phần [<input />](https://hocwebchuan.com/reference/tag/tag_input.php).
* Tag <label> không hiển thị bất cứ gì đặc biệt cho người dùng, tuy nhiên nó cung cấp một cải thiện cho người sử dụng chuột, nếu click chuột vào nhãn, sẽ đưa con trỏ chuột vào vùng <input />.
* Muốn sử dụng hiệu quả <label>, cần thiết phải cho giá trị [id](https://hocwebchuan.com/reference/tag/attr/attr_standard.php) của <input /> và giá trị [for](https://hocwebchuan.com/reference/tag/tag_label.php#requireAttr) của <label>, hai giá trị này phải trùng nhau.

# 

## 

## 

# **13.JS**

## setInterval()

Hàm setInterval() như cái tên của nó, hàm này sẽ thường được sử dụng để thiết lập độ trễ cho các hàm sẽ được thực hiện lặp lại như là hiệu ứng. Hàm setInterval() có quan hệ gần gũi với setTimeout() - cú pháp của 2 hàm cũng có vẻ giống nhau

## Window.

Đối tượng window có tác dụng như với tên của nó là tác động đến của sổ trình duyệt