**CSS**

Dùng để tạo kiểu hiển thị cho các phần tử trong html như màu, chữ, hoạt ảnh,...

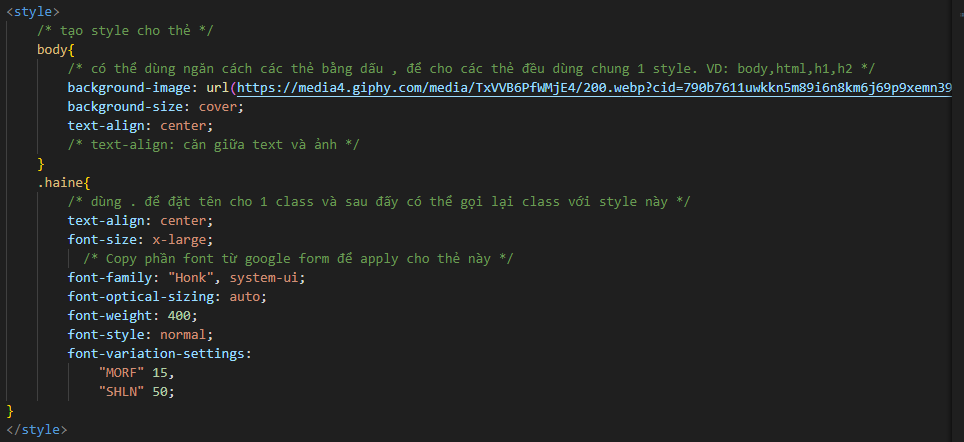
# Có 3 cách sử dụng CSS

## Inline

Lồng ngay trong dòng của các phần tử trong dòng đó, thêm đặc tính style.



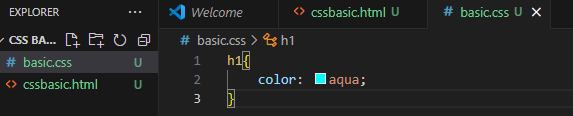
## Internal

Dùng ở ngay đầu trước khi code, viết trong cặp thẻ style và quy định cho tất cả các thẻ có cùng tên.

## External

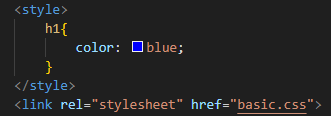
Viết ở 1 file ngoài có đuôi .css, thực hiện kết nối với file html bằng thẻ “link” (đặt ở đầu hoặc trong cặp thẻ <head></head đều được> và với 1 file CSS có thể sử dụng cho nhiều file html.





Độ ưu tiên: Inline sẽ có độ ưu tiên cao nhất, sau đó sẽ theo thứ tự xuất hiện.

VD:

 Nếu link external xuất hiện sau cặp thẻ style, sẽ ưu tiên cặp thẻ style được kích hoạt.

# Bộ chọn CSS (Selecter)

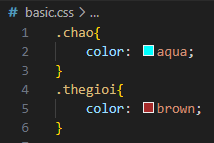
Bộ chọn giúp ta có thể chọn và thao tác chỉnh sửa các phần tử theo ý muốn. Có 3 bộ chọn:

## Chọn theo class

Định nghĩa 1 class bằng phần tử class



Trong CSS, gọi class muốn thao tác bằng dấu “.” sau đó là tên class



## Chọn theo id

Cách sử dụng gần như tương tự theo class, nhưng thay dấu “.” bằng dấu “#”



Trong CSS, gọi id muốn thao tác bằng “#” sau đó là tên id

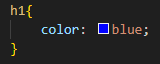


Tuy nhiên id không thông dụng bằng class, id phục vụ chủ yếu cho các trường hợp đặc biệt, id mang tính duy nhất chứ ko chỉ để nhóm class phục vụ cho css. VD:



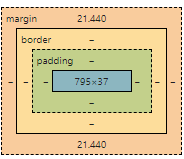
Với ví dụ trên, khi id được gán vào và cặp thẻ label được sử dụng cho id đó thì sẽ giúp ta ấn vào không chỉ trong phần text mà ấn cả vào **tên người dùng:** cũng sẽ trỏ về phần trong text. Chung quy sẽ giúp cải thiện cho thao tác người dùng vì id là để đại diện cho phần trong text đó.

## Chọn theo tên thẻ

 phân biệt bằng cách ko có định dạng dấu “.” hoặc dấu “#”. Cách sử dụng tương tự.

# Box model

Là hình dạng sẵn có của các thẻ html, mỗi 1 thẻ html đều có 1 box model.



Mỗi 1 box model đều có đầy đủ các yếu tố:

## Margin

khoảng cách từ viên đến các thẻ khác

Có thể tùy chỉnh margin qua các thẻ thuộc tính:

1. **margin-top**: Xác định khoảng cách giữa phần tử và phần tử phía trên.
2. **margin-right**: Xác định khoảng cách giữa phần tử và phần tử phía bên phải.
3. **margin-bottom**: Xác định khoảng cách giữa phần tử và phần tử phía dưới.
4. **margin-left**: Xác định khoảng cách giữa phần tử và phần tử phía bên trái.
5. **margin :** Xác định khoảng cách của cả 4 cạnh.

Có thể căn chỉnh bằng giá trị auto. Khi đó, margin sẽ được căn giữa cả 4 chiều so với phần tử cha

Hoặc sử dụng margin: 20px 10px để căn chỉnh theo cặp trên dưới, trái phải

## Border

Khoảng cách từ padding đến margin

Có thể tùy chỉnh border qua các thẻ thuộc tính:

1. **border-width**: Xác định độ dày của border.
2. **border-style**: Xác định kiểu của border, như solid, dashed, dotted, và nhiều hơn nữa.
3. **border-color**: Xác định màu sắc của border.
4. **border-radius**: Xác định độ cong của border, tạo ra góc bo tròn.
5. **border**: Thuộc tính rút gọn để thiết lập border-width, border-style và border-color cùng một lúc.

Các chức năng top, bot, left, right sử dụng như margin.

## Padding

Khoảng cách từ nội dung đến viền

Thẻ padding có cách điều chỉnh y như margin, thay tên thuộc tính thành padding-top/bottom/left/right

## Phần nội dung (content)

nội dung, ví dụ như hình ảnh, văn bản,…

Trong CSS, không có một thuộc tính cụ thể nào được sử dụng để tùy chỉnh nội dung (content) của một phần tử HTML. Nội dung của một phần tử HTML thường được quy định bằng các thẻ HTML như **<div>**, **<p>**, **<span>**, **<h1>** đối với văn bản, hoặc **<img>**, **<video>**, **<audio>** đối với các phần tử đa phương tiện như hình ảnh, video, và âm thanh.

Tuy nhiên, bạn có thể sử dụng các thuộc tính CSS khác để tùy chỉnh vị trí, kích thước và hiển thị của nội dung bên trong một phần tử. Dưới đây là một số thuộc tính CSS thường được sử dụng để tùy chỉnh nội dung:

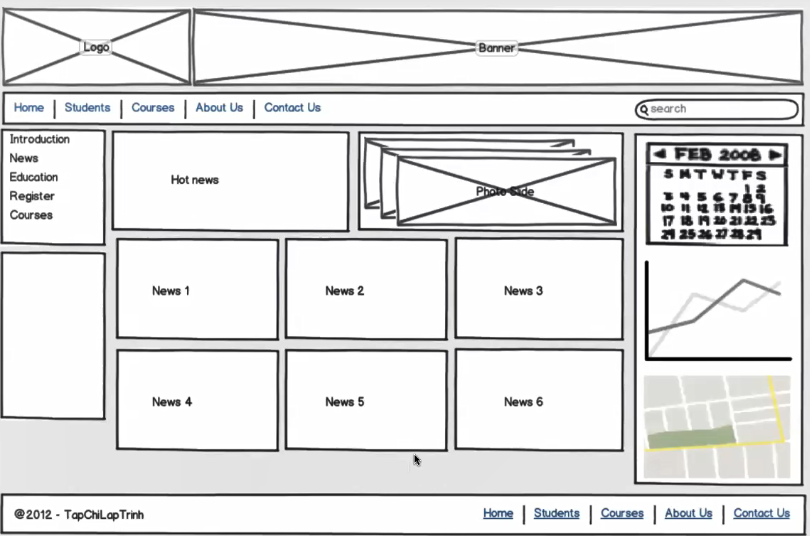
1. **font-size**: Xác định kích thước của văn bản.
2. **font-family**: Xác định font chữ được sử dụng cho văn bản.
3. **color**: Xác định màu sắc của văn bản.
4. **text-align**: Xác định căn chỉnh của văn bản (left, center, right).
5. **line-height**: Xác định độ cao của dòng văn bản.
6. **background-color**: Xác định màu nền của phần tử.

Với các phần tử đa phương tiện như hình ảnh, video và âm thanh, bạn có thể sử dụng các thuộc tính như **width**, **height**, và **object-fit** để tùy chỉnh kích thước và hiển thị của chúng.

# CSS Layout

Là giao diện cơ bản của một trang web, bao gồm có:

Header, Navtive, Main, Footer



Để chia được các bố cục, ta tận dụng các thuộc tính trong CSS như layout hoặc flexbox để chia cho đẹp

Dưới đây sẽ giới thiệu về 1 vài thuộc tính.

## Float

Trong HTML và CSS, thuộc tính **float** được sử dụng để di chuyển một phần tử (thường là phần tử block) sang trái hoặc phải của phần tử cha của nó, và cho phép các phần tử khác trôi lấp xung quanh nó.

Dưới đây là cách sử dụng thuộc tính **float**:

<div style="float: left;">Phần tử được di chuyển sang trái</div>

hoặc

<div style="float: right;">Phần tử được di chuyển sang phải</div>

Khi bạn áp dụng thuộc tính **float** cho một phần tử, phần tử đó sẽ không chiếm toàn bộ không gian của phần tử cha mà nó đang nằm trong. Thay vào đó, các phần tử khác sẽ trôi lấp xung quanh phần tử đó.

Lưu ý rằng việc sử dụng **float** có thể gây ra một số vấn đề liên quan đến bố cục của trang web, đặc biệt là khi không quản lý được các phần tử chồng lên nhau. Để xử lý các vấn đề này, thường được khuyến khích sử dụng các kỹ thuật bố cục hiện đại như Flexbox hoặc Grid Layout.

Box sizing: