

# BUỔI 3: SỬ DỤNG CSS

Ban WEB - SFIT

# Một số thuộc tính cơ bản

# Thuộc tính Padding

## (Lớp đậm)

Có 4 thuộc tính: padding-top, padding-right, padding-bottom, padding-left  
Sử dụng để căn chỉnh nội dung so với các viền tương ứng: trên, phải, dưới, trái.

Có làm thay đổi kích thước của phần tử

127.0.0.1:5500/VD.html

127.0.0.1:5500/VD.html

DevTools is now available in Vietnamese! Always match Chrome's language Switch DevTools to Vietnamese Don't show again

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights ▾

Styles Computed Layout Event Listeners >

margin -  
border -  
padding -  
100x100 -  
-

Filter Show all Group

align-items center  
background-color #rgb(255, 165, 0)  
color #rgb(255, 255, 255)  
display flex  
height 100px  
justify-content center  
unicode-bidi isolate  
width 100px

Rendered Fonts

Family name: Times New Roman  
PostScript name: TimesNewRomanPSMT  
Font origin: Local file (3 glyphs)

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <style>
      .box{
        width:100px;
        height:100px;
        color:white;
        display: flex;
        align-items:center;
        justify-content:center;
        background-color:orange;
      }
    </style>
  </head>
  <body>
    <div class="box">box</div> flex == $0
    <!-- Code injected by live-server -->
    <script>...</script>
  </body>
</html>
```

html body div.box

chưa có padding

127.0.0.1:5500/VD.html

127.0.0.1:5500/VD.html

DevTools is now available in Vietnamese! Always match Chrome's language Switch DevTools to Vietnamese Don't show again

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder > | X

Styles Computed Layout Event Listeners >

```
<!DOCTYPE html>
<html lang="en">
  <head></head>
  <body>
    <div class="box">box</div> flex == $0
    <!-- Code injected by live-server -->
    <script></script>
  </body>
</html>
```

margin -  
border -  
padding 30  
100×100  
-

Filter Show all Group

- align-items center
- background-color #rgb(255, 165, 0)
- color #rgb(255, 255, 255)
- display flex
- height 100px
- justify-content center
- padding-top 30px
- unicode-bidi isolate
- width 100px

Rendered Fonts

Family name: Times New Roman  
PostScript name: TimesNewRomanPSMT  
Font origin: Local file (3 glyphs)

html body div.box

padding phía trên 30px

# Thuộc tính Border

## Lớp viền

Có làm thay đổi kích thước phần tử

Border-width: Độ dày của đường viền

Border-style: Kiểu đường viền

Border-color: Màu đường viền

The screenshot shows a code editor and a browser window side-by-side. The code editor displays the following HTML and CSS:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<style>
.box{
width:100px;
height:100px;
color:white;
display: flex;
align-items:center;
justify-content:center;
background-color:orange;
padding: 30px;
border-width: 10px;
border-style:solid;
border-color:red;
}
</style>
</head>
<body>
<div class="box">box</div>
</body>
</html>
```

The browser window shows a yellow square with the word "box" in white text in the center. The DevTools inspection pane on the right shows the element's computed styles:

Property	Value
margin	-9.6px -9.6px -9.6px -9.6px
border	9.6px solid red
padding	30px
width	100px
height	100px
color	white
display	flex
align-items	center
justify-content	center
background-color	orange
border-width	10px
border-style	solid
border-color	red

# Border màu đỏ 10px

# Thuộc tính margin

Lớp lè



Tạo ra khoảng cách xung quanh phần tử

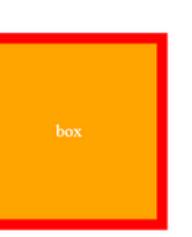
không làm thay đổi kích thước phần tử

Tương tự Padding, Margin cũng có 4 thuộc tính để căn  
chỉnh 4 vị trí: top, right, bottom, left hay auto

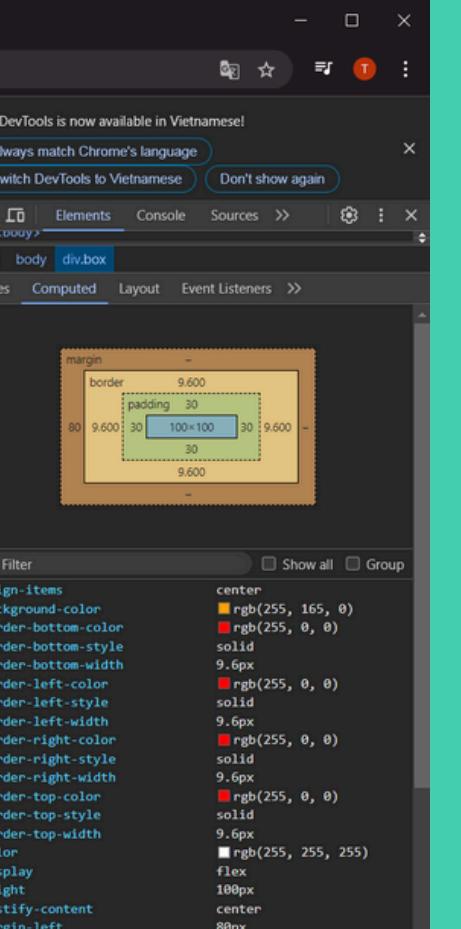
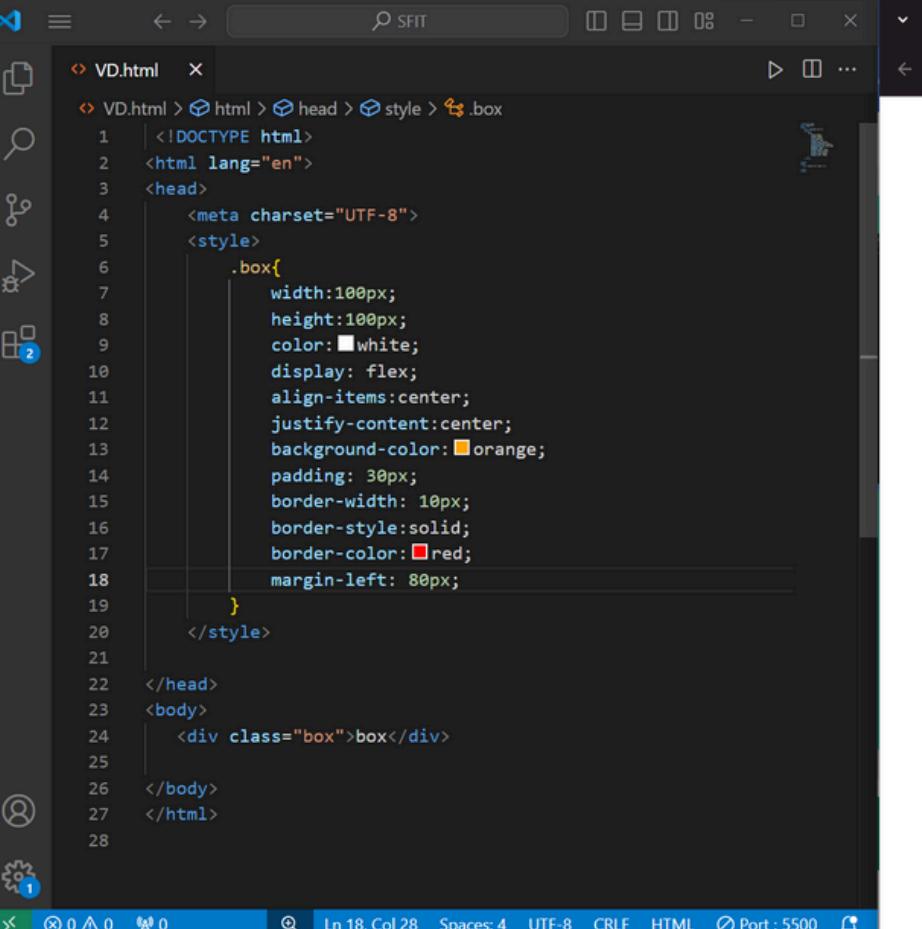
Lề trên cách 50 px



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <style>
        .box{
            width:100px;
            height:100px;
            color: white;
            display: flex;
            align-items:center;
            justify-content:center;
            background-color: orange;
            padding: 30px;
            border-width: 10px;
            border-style:solid;
            border-color: red;
            margin-top:50px;
        }
    </style>
</head>
<body>
    <div class="box">box</div>
</body>
</html>
```



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <style>
        .box{
            width:100px;
            height:100px;
            color: white;
            display: flex;
            align-items:center;
            justify-content:center;
            background-color: orange;
            padding: 30px;
            border-width: 10px;
            border-style:solid;
            border-color: red;
            margin-left: 80px;
        }
    </style>
</head>
<body>
    <div class="box">box</div>
</body>
</html>
```



Lề bên trái cách 80px

# Background trong CSS

## 01 Background-color

Là thuộc tính sử dụng để định dạng màu nền cho website

Giá trị:+ Màu theo tên tiếng anh: red, green, blue, white, black,..

+ Là mã màu theo hệ hex.

## 02 Background-image

Chọn hình nền là ảnh

background : url (liên kết ảnh);

## 03 Background-repeat

lặp lại ảnh nền nếu ảnh nền nhỏ hơn định dạng

## 04 Background-position

Định vị trí nền của website

# Font trong CSS

## Font-family :

Là thuộc tính định dạng font chữ cho website  
Giá trị là tên các font chữ.

## Font-weight :

Đặt độ dày hoặc độ đậm của font chữ

## Font-size :

Cỡ chữ, giá trị là các đơn vị như: px, em, ...

# Text trong css

Color : định dạng màu cho văn bản

Text-align:

- Định vị trí của 1 đoạn văn bản.
- Giá trị:
  - + Left: Căn trái
  - + Center: Căn giữa
  - + Right: Căn phải.
  - + Justify: Căn đều.

# CSS selector

```
style.css > h2
1  h2{
2      color: black;
3      font-size:10px;
4 }
```

Thẻ

CSS thảng vào các  
thẻ html

```
e.css > #black
#black{
    color: black;
    font-size:10px;
}
```

Class

CSS vào các thẻ có  
chung class,nhiều thẻ  
chung 1 class

```
# style.css > .black
1  .black{
2      color: black;
3      font-size:10px;
4 }
```

ID

Giống với class,nhưng  
chỉ nên tồn tại 1 thẻ  
chứa 1 ID duy nhất



CSS in HTML  
có 3 cách

Internal

External

Inline

# Internal

Sử dụng css ngay trong file.html

Được đặt trong cặp thẻ `<style></style>` và chứa ngay tại file.html  
cần xử lý( hay trong page cần định dạng).

VD.html × index.html 3 127.0.0.1:5500/mikanko\_card/VD.html

mikanko\_card > VD.html > html > head > style > h2

```
1  | <!DOCTYPE html>
2  | <html lang="en">
3  | <head>
4  |   <meta charset="UTF-8">
5  |   <style>
6  |     h1{
7  |       color: red;
8  |       font-size: 30px;
9  |     }
10 |     h2{
11 |       color: blue;
12 |       font-size: 45px;
13 |     }
14 |   </style>
15 |
16 </head>
17 <body>
18   <h1>CSS in html</h1>
19   <h2>chữ màu đỏ</h2>
20
21 </body>
22 </html>
23
```

**CSS in html**

**chữ màu đỏ**

# External

Cách khai báo sử dụng 1 file .css ở bên ngoài  
file .html cần định dạng.

# Mức độ ưu tiên thấp nhất

The screenshot shows a code editor interface with two tabs and a browser preview window.

**Code Editor Tab:** Contains the file `VD.html`. The code includes an external CSS link and inline CSS rules for `h1` and `h2` elements.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <link rel="stylesheet" href="style.css">
    <!-- Dẫn liên kết với file style.css --&gt;
    &lt;style&gt;
        h1{
            color:red;
            font-size: 30px;
        }
        /* h2{
            color:blue;
            font-size: 45px;
        } */
    &lt;/style&gt;
&lt;/head&gt;
&lt;body&gt;
    &lt;h1&gt;CSS in html&lt;/h1&gt;
    &lt;h2&gt;chữ màu đỏ&lt;/h2&gt;
&lt;/body&gt;
&lt;/html&gt;</pre>

Browser Preview Tab: Shows the rendered output of the HTML code. The h1 heading is displayed in red, and the h2 heading is displayed in black, indicating that the inline styles in the style block of the HTML file are being applied.



Output:



CSS in html



chữ màu đỏ

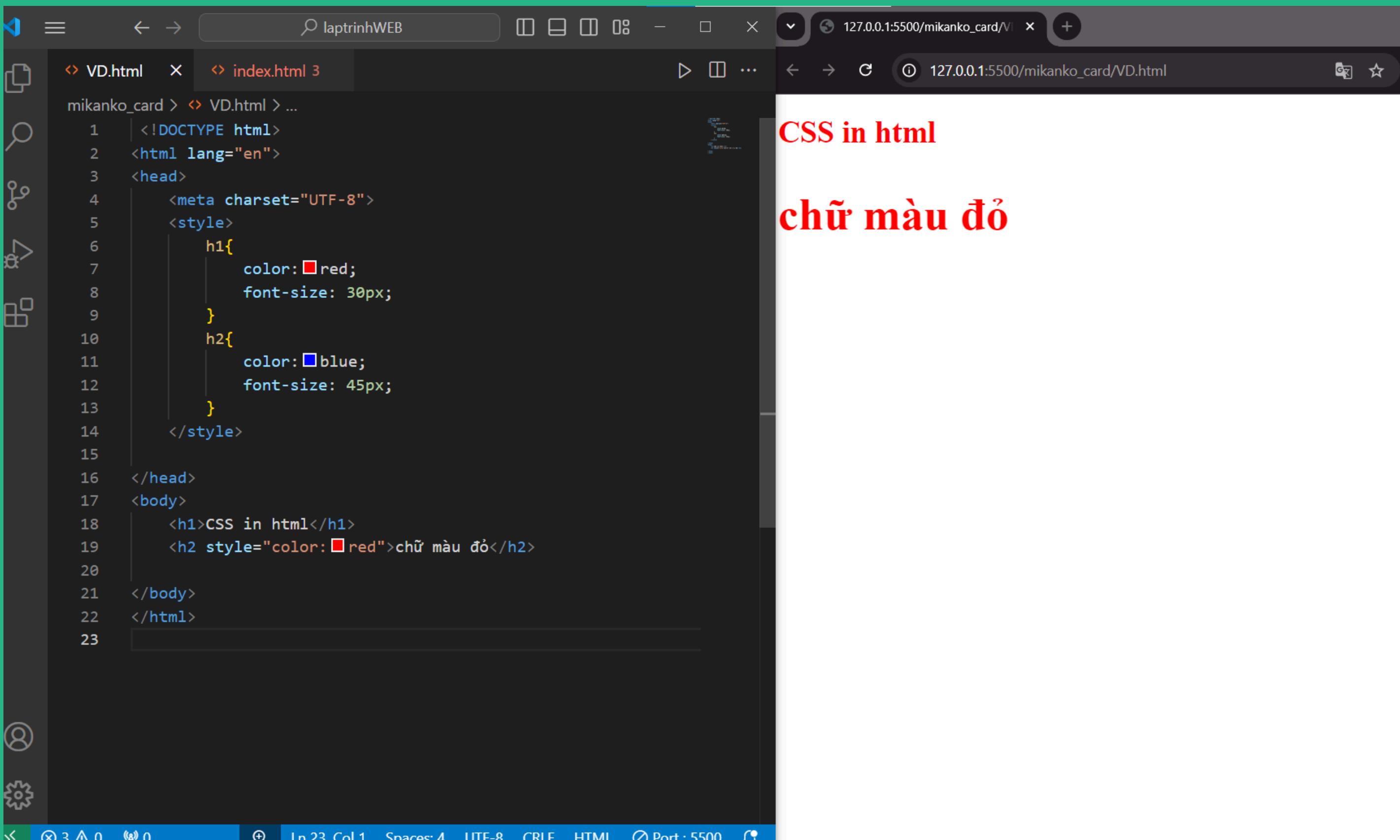

```



Inline

# khai báo ngay tại thẻ html cần định dạng

Inline có độ ưu tiên cao nhất



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <style>
        h1{
            color: red;
            font-size: 30px;
        }
        h2{
            color: blue;
            font-size: 45px;
        }
    </style>
</head>
<body>
    <h1>CSS in html</h1>
    <h2 style="color: red">chữ màu đỏ</h2>
</body>
</html>
```

CSS in html  
chữ màu đỏ

# Pseudo-class

Lớp giả

## **:hover**

Kích hoạt khi di chuyển chuột vào phần tử

## **:active**

kích hoạt khi bấm chuột vào phần tử

## **:focus**

Dùng cho các phần tử dạng nút bấm ,khi bấm chuột sẻ kích hoạt

## **::before và after**

Được sử dụng để chèn nội dung trước (::before) hoặc sau (::after) nội dung của phần tử mà không cần phải thay đổi trực tiếp trong HTML.

Phải kết hợp với thuộc tính content để tạo hoặc thêm nội dung



# Một chút về responsive

**Fluid Layouts:** Sử dụng đơn vị tỷ lệ phần trăm thay vì đơn vị cố định như pixel để các phần tử trên trang có thể mở rộng hoặc thu nhỏ dựa trên kích thước màn hình.

**Media Queries:** Là một tính năng của CSS cho phép bạn áp dụng các kiểu khác nhau cho các kích thước màn hình khác nhau.

**Flexible Images and Media:** Đảm bảo rằng hình ảnh và các phương tiện khác có thể thay đổi kích thước để phù hợp với kích thước của phần tử chứa chúng.

**Responsive Typography:** Sử dụng các đơn vị như em hoặc rem thay vì pixel để kích thước chữ có thể điều chỉnh theo kích thước của màn hình hoặc thậm chí theo cài đặt của người dùng.

**Mobile-First Design:** Thiết kế trang web với giả định rằng người dùng sẽ truy cập từ thiết bị di động trước tiên, sau đó mở rộng thiết kế cho các thiết bị có kích thước lớn hơn.

**Viewport Meta Tag:** Sử dụng thẻ meta trong HTML để kiểm soát cách trang web được hiển thị trên thiết bị di động.

```
@media (max-width: 768px) {  
    /* Các quy tắc CSS áp dụng cho màn hình nhỏ hơn 768px */  
}
```

```
img {  
    max-width: 100%;  
    height: auto;  
}
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```



# Responsive với display:flex

## 1. flex-direction:

- Xác định hướng của các phần tử con trong flex container.
- Các giá trị có thể là:
  - row (mặc định): Các phần tử con sắp xếp theo hàng ngang.
  - column: Các phần tử con sắp xếp theo cột dọc.
  - row-reverse: Các phần tử con sắp xếp theo hàng ngang nhưng theo chiều ngược lại.
  - column-reverse: Các phần tử con sắp xếp theo cột dọc nhưng theo chiều ngược lại.

## 2. flex-wrap:

- Quyết định xem các phần tử con có được phép xuống dòng khi không còn đủ không gian trên một dòng hay không.
- Các giá trị có thể là:
  - nowrap (mặc định): Không xuống dòng.
  - wrap: Xuống dòng nếu cần.
  - wrap-reverse: Xuống dòng theo chiều ngược lại.



# Thực hành