

Bài 5: ID, CLASS TRONG HTML VÀ THUỘC TÍNH FLOAT TRONG CSS

Xem bài học trên website để ủng hộ Kteam: [ID, class trong HTML và thuộc tính float trong CSS](#)

Mọi vấn đề về lỗi website làm ảnh hưởng đến bạn hoặc thắc mắc, mong muốn khóa học mới, nhằm hỗ trợ cải thiện Website. Các bạn vui lòng phản hồi đến Fanpage [How Kteam](#) nhé!

Dẫn nhập

Ở bài trước thì chúng ta đã cùng nói về [MODEL BOX TRONG CSS](#) cũng như các thuộc tính liên quan của nó đến một phần tử HTML.

Trong bài học ngày hôm nay chúng ta sẽ tìm hiểu về **ID và class trong HTML và thuộc tính float trong CSS**.

Nội dung

Để đọc hiểu bài này, bạn nên có kiến thức về:

- [TỔNG QUAN HTML CƠ BẢN](#)
- [TỔNG QUAN CSS CƠ BẢN](#)
- [CẤU TRÚC MODEL BOX TRONG CSS](#)

Trong bài này, Kteam sẽ cùng bạn tìm hiểu về:

- ID và Class trong HTML
- Các cú pháp nâng cao hiệu suất code
- Thuộc tính Float trong CSS
- Clearfix class

Các thao tác được thực hành trên project của bài trước, bạn có thể tải xuống project ở cuối bài [MODEL BOX TRONG CSS](#)

Lưu ý: Bài viết nhằm tóm tắt nội dung thực hiện project & giúp bạn nắm bắt các phần được hướng dẫn một cách đơn giản nhất. Tuy nhiên, hãy theo dõi video để học hỏi các kiến thức và kỹ thuật chi tiết nhất từ tác giả.

ID và Class trong HTML

Giả sử, với project Blog mà chúng ta đang làm trong bài trước, mình cần thêm các thẻ `<h1>` khác vào trong blog như sau:

```
<h1>Hello Kters</h1>  
<h1>Front End Course</h1>
```

Tuy nhiên, do mình đã chỉ định thẻ `<h1>HOWKTEAM BLOG</h1>` trước đó có các thuộc tính riêng tại file **style.css** như **chữ có kích thước 900px, màu xanh lá (green); nền màu aqu; có viền chấm 2px (dotted) màu đỏ (red)**,...Điều này dẫn đến việc các thẻ `<h1>` được thêm vào trong file **index.html** sau này có định dạng thuộc tính tương tự thẻ `<h1>` ban đầu mà mình không mong muốn.



Chính vì vậy **Class** & **Id** sinh ra để giải quyết các trường hợp này, giúp xác định cụ thể phần tử mà bạn muốn định dạng. Tuy nhiên, ở khóa [LẬP TRÌNH FRONT END CƠ BẢN](#) có thể bạn chưa thấy được nhiều khác biệt của 2 thuộc tính này, mà sẽ tìm hiểu rõ hơn trong các khóa CSS NÂNG CAO.

Thuộc tính ID

Thuộc tính id dùng để đặt tên cho **một phần tử** trong HTML, mỗi thuộc tính id chỉ chỉ định cho **một phần tử duy nhất** trong file HTML hiện hành.

Giá trị id được sử dụng trong CSS hoặc JavaScript để chỉ định cụ thể định dạng hoặc thực hiện tác vụ nhất định cho phần tử đó.

Gán thuộc tính id trong HTML:

<phần tử HTML id= "id_name " >

Sử dụng giá trị id trong CSS:

```
#id_name{  
    thuộc tính: giá trị;  
}
```

Trong đó:

- Thuộc tính **id** có thể sử dụng trên bất kỳ phần tử HTML nào.
- **id_name** (giá trị id) chứa các ký tự chữ cái, số, gạch (_), nối (-); có phân biệt chữ hoa và chữ thường. Trong giá trị id phải chứa ít nhất một ký tự và không có khoảng trắng. Không nên bắt đầu tên id bằng số.

Ví dụ:

Tại file **index.html**

```
<h1 id="heading1">HOWKTEAM BLOG</h1>
```

Tại file **style.css** thay **h1** bằng giá trị id **heading1**

```
#heading1{  
    font-family: 'Lato', sans-serif;  
    font-weight: 900;  
    font-size: 50px;  
}  
  
#heading1{  
    color: green;  
    /* Sử dụng tên trong của màu sắc */  
  
    color: #0000ff;  
    /* Sử dụng mã màu hexan */  
  
    color: rgb(0, 255, 0);  
    /* Sử dụng phối màu rgb() red green blue */  
  
    color: rgba(0, 255, 0, 1);  
    /* Tương tự rgb() và thêm 1 thông số là độ mờ */  
}
```

```
#heading1{
  border: 4px solid red;
  border-style: dotted;
  background-color: aqua;
}
```

Sau khi gán id trang blog thay đổi như hình:

- Ngoại trừ thẻ <h1> có chứa thuộc tính id **heading1** có định dạng riêng, các thẻ <h1> khác đều có định dạng mặc định.

HOWKTEAM BLOG

Hello Kters

Front end course

HOWKTEAM

Hello Kter

HOWKTEAM HOWKTEAM HOWKTEAM Lorem, ipsum dolor sit amet consectetur adipisicing elit. Rem debitis eos dolorem itaque tempore libero ipsa quibusdam fuga, vel, molestiae maxime harum consequatur architecto repellat dolores cupiditate iure nisi fugiat? Lorem ipsum, dolor sit amet consectetur adipisicing elit. Nulla, culpa.

HOWKTEAM 2

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Voluptas, incididunt dolores perferendis sapiente, quas voluptatem cumque ipsa accusantium ab possimus iusto quo porro esse eius nemo explicabo quasi rerum! Cumque.



Ứng dụng Id trong việc đánh dấu

Thuộc tính id cung cấp cho bạn một ứng dụng khá thú vị, thường hay nhìn thấy ở trang LandingPage hoặc một số web có nội dung khá dài.

Cụ thể, bạn có thể dùng id như một vị trí, sau đó thêm liên kết đi đến vị trí đó

Tạo vị trí đánh dấu theo cách gán id thông thường

<phần tử HTML id= "id_name " >

Tạo đường dẫn đến vị trí đánh dấu (trong cùng trang HTML)

 nội_dung_gắn_liên_kết

Tạo đường dẫn đến vị trí đánh dấu (từ trang khác)

```
<a href=" <tên_trang_html>#id_name"> nội_dung_gắn_liên_kết  
</a>
```

Ứng dụng này sẽ được nói rõ hơn trong quá trình xây dựng website Landing Page của khóa học này.

Thuộc tính class

Thuộc tính class dùng để đặt tên cho các phần tử trong html, mỗi thuộc tính class có thể chỉ định cho nhiều phần tử HTML (cùng hoặc khác loại) để có cùng định dạng.

Giá trị class được sử dụng trong CSS hoặc JavaScript để chỉ định đồng loạt các định dạng chung hoặc thực hiện chung tác vụ cho các phần tử HTML có cùng thuộc tính class.

Gán thuộc tính class trong HTML:

```
<phần tử HTML class= "class_name " >
```

Sử dụng giá trị class trong CSS:

```
.class_name{  
    thuộc tính: giá trị;  
}
```

Trong đó:

- Thuộc tính **class** có thể sử dụng trên bất kỳ phần tử HTML nào.
- **class_name** (giá trị class) chứa các ký tự chữ cái, số, gạch (_), nối (-); có phân biệt chữ hoa và chữ thường. Trong giá trị class phải chứa ít nhất một ký tự và không có khoảng trắng. Không nên bắt đầu tên class bằng số.

Ví dụ:

Tại file **index.html** thay vì dùng **id heading1**, mình đổi thành **class heading**.
Và thêm thẻ **p** cũng có cùng class **heading** như thẻ **h1**

```
<h1 class="heading">HOWKTEAM BLOG</h1>
<h1 class="heading">Hello Kters</h1>
<p class="heading">
  Lorem, ipsum dolor sit amet consectetur adipisicing elit. Voluptas, incidunt dolores
  perferendis sapiente,
  quas voluptatem cumque ipsa accusantium ab possimus iusto quo porro esse eius
  nemo explicabo quasi rerum! Cumque.
</p>
```

Tại file **style.css** thay giá trị id **heading1** bằng giá trị class **heading**

```
.heading1{
  font-family: 'Lato', sans-serif;
  font-weight: 900;
  font-size: 50px;
}

.heading1{
  color: green;
  /* Sử dụng tên trong của màu sắc */

  color: #0000ff;
  /* Sử dụng mã màu hexan */

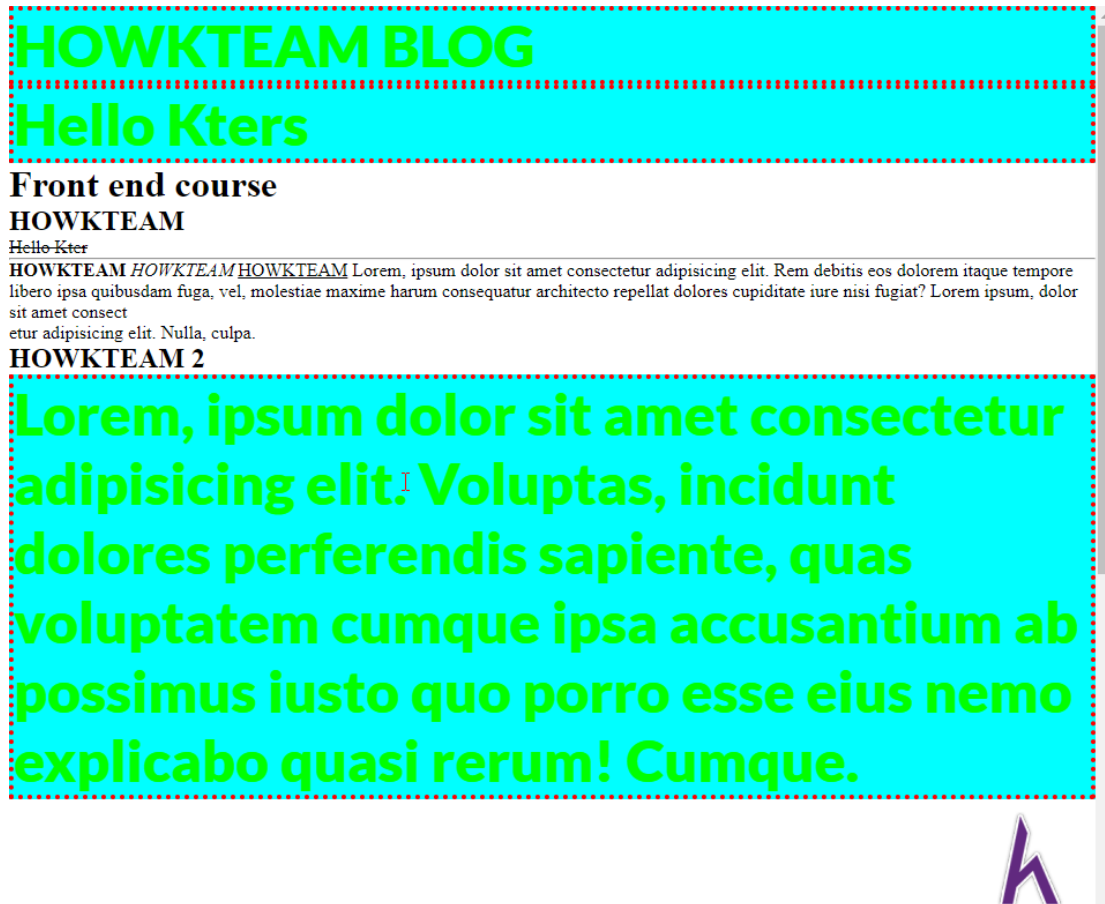
  color: rgb(0, 255, 0);
  /* Sử dụng phối màu rgb() red green blue */

  color: rgba(0, 255, 0, 1);
  /* Tương tự rgb() và thêm 1 thông số là độ mờ */
}

.heading1{
  border: 4px solid red;
  border-style: dotted;
  background-color: aqua;
}
```

Sau khi gán class trang blog thay đổi như hình:

- Các thẻ có chứa thuộc tính class **heading** (như h1, p) có cùng kiểu định dạng, các thẻ khác đều có định dạng mặc định.



Sử dụng nhiều class cho một phần tử HTML

Các phần tử HTML có thể có một hoặc nhiều thuộc tính class, các thuộc tính được tách biệt nhau bằng dấu cách.

<phần tử HTML class= "class_name_1 class_name_2 " >

Ví dụ:

Index.html

```
<h1 class="heading center">HOWKTEAM BLOG</h1>
<h1 class="heading">Hello Kters</h1>
<p class="heading">
```


Lorem, ipsum dolor sit amet consectetur adipisicing elit. Voluptas, incidunt dolores
perferendis sapiente,
quas voluptatem cumque ipsa accusantium ab possimus iusto quo porro esse eius
nemo explicabo quasi rerum! Cumque.
</p>

Style.css

```
.center{
  text-align: center;
}
.heading{
  font-family: 'Lato', sans-serif;
  font-weight: 900;
  font-size: 50px;
}

.heading{
  color: green;
  /* Sử dụng tên trong của màu sắc */

  color: #0000ff;
  /* Sử dụng mã màu hexan */

  color: rgb(0, 255, 0);
  /* Sử dụng phối màu rgb() red green blue */

  color: rgba(0, 255, 0, 1);
  /* Tương tự rgb() và thêm 1 thông số là độ mờ */
}

.heading{
  border: 4px solid red;
  border-style: dotted;
  background-color: aqua;
}
```

Từ kiến thức ở các phần trên, bạn hoàn toàn có thể chạy thử và phân tích kết quả của ví dụ này!

Các cú pháp nâng cao hiệu suất code

Trong quá trình code, việc gõ đủ định dạng của các thẻ khiến bạn khá mất thời gian dù đã được hỗ trợ nhắc lệnh từ Visual Studio Code. Để tối đa hiệu suất code, Kteam gợi ý bạn sử dụng một số cú pháp sau

- Gõ nhanh các cặp thẻ div chứa id

#id_name

Ví dụ: gõ **#heading1** cho kết quả

```
<div id="heading1"></div>
```

- Gõ nhanh các cặp thẻ div chứa class

.class_name

Ví dụ: gõ **.container** cho kết quả

```
<div class="container"></div>
```

- Gõ nhanh các phần tử khác trong HTML với id

phần_tử_HTML#id_name

Ví dụ: gõ **h1#heading1** cho kết quả

```
<h1 id="heading1"></h1>
```

- Gõ nhanh các phần tử khác trong HTML với class

phần_tử_HTML.id_name

Ví dụ: gõ **h1.container** cho kết quả

```
<h1 class="container"></h1>
```

Cú pháp reform trang HTML

Ctrl + A sau đó **Ctrl + K** cuối cùng là **Ctrl + F**

Giúp canh chỉnh các thẻ code đã sử dụng trong file HTML giúp bạn dễ dàng quan sát file HTML của bạn.

Chuyển comment thành code

Bạn có thể chuyển code thành comment và ngược lại bằng cách tô đen đoạn code cần chuyển rồi dùng tổ hợp phím:

Ctrl + /

Ta có thể chỉnh sửa một chút tại file **index.html** giúp bạn ứng dụng các cú pháp nhanh trên:

```
div class="container">
  <div class="content">
    <h1 class="heading" id="heading1">HOWKTEAM BLOG</h1>
    <h2>HOWKTEAM</h2>

    <!--heading tiêu đề của các phần trong website-->
    <p>
      <strike>Hello Kter</strike>
      <!--tao chu có gạch ngang-->
      <br>
      <!--break có tác dụng giúp xuống dòng-->
      <hr>
      <b>HOWKTEAM</b>
```

```

<!--bold tao chu in dam-->
<em>HOWKTEAM</em> <!-- Emphasize tao chu in nghieng-->
<u>HOWKTEAM</u>
<!--underline tao chu co gach chan-->
Lorem, ipsum dolor sit amet consectetur adipisicing elit.
Rem debitis eos dolorem itaque tempore libero ipsa quibusdam fuga, vel,
molestiae maxime harum consequatur architecto repellat dolores cupiditate
iure nisi fugiat?
  Lorem ipsum, dolor sit amet consectetur <br>et adipisicing elit. Nulla, culpa.
</p>

<h2>HOWKTEAM 2</h2>
<p>Lorem ipsum dolor sit amet consectetur,
  adipisicing elit. Voluptatem autem reprehenderit asperiores
  distinctio similique beatae adipisci officiis, rerum vitae quaerat veniam
  porro repellendus odit aspernatur a est magni odio. Non!</p>
<!-- Noi ban chua van ban-->

  <!-- src source alt alternative -->
  <a href="https://www.howkteam.vn/">Link dan den HOWKTEAM</a>

<form>
  <input type="text" placeholder="dien ten cac ban vao day">
  <input type="submit" value="gui tin nhan">
  <textarea cols="100" rows="10" placeholder="hello anh em"></textarea>
</form>
</div>
<div class="avatar">
  
  <h4>MY NAME IS ABC</h4>
</div>
<div class="info">
  <div>
    SĐT: 09-999-999-999
  </div>
  <div>
    Địa chỉ: số 1, quận 2
  </div>
  <div>
    Email: blogtest@gmail.com
  </div>

```

```
</div>
</div>
</div>
```

Thuộc tính float trong CSS

Nói một cách đơn giản **thuộc tính Float** xác định rằng một phần tử HTML nên được đặt dọc theo bên trái hay bên phải của vùng chứa của nó, đồng thời nó cho phép các phần tử văn bản và phần tử nội tuyến khác quần quanh nó.

Phần tử được xóa khỏi luồng thông thường của trang web, mặc dù vẫn còn một phần của luồng (ngược lại với vị trí hiển thị tuyệt đối)

Cú pháp:

```
phần_tử_HTML{
    float: left (hoặc right );
    /* Mặc định float: none */
}
```

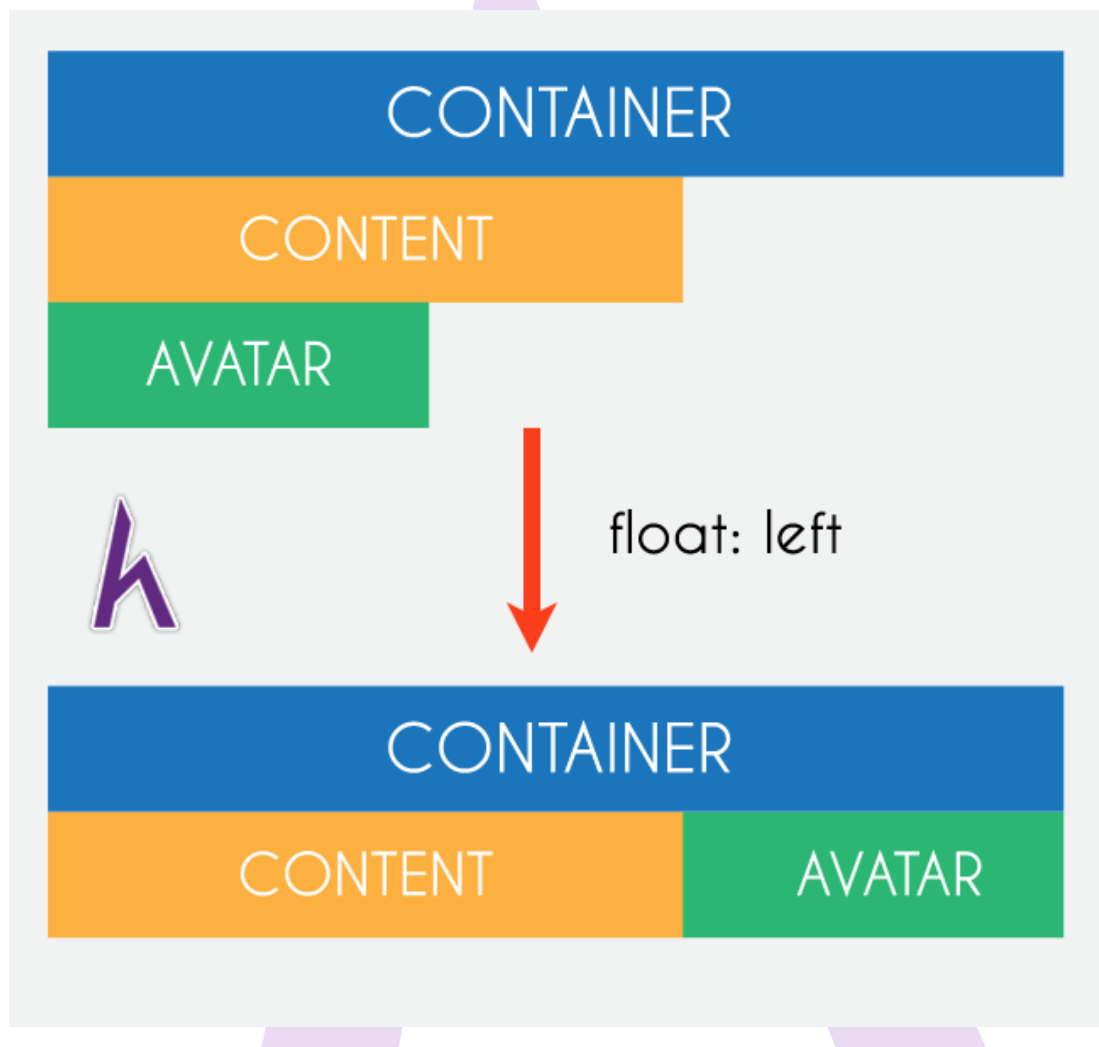
Ví dụ: Sử dụng thuộc tính **float: left** (trong trường hợp này content chiếm 75% và avatar chiếm 25% chiều rộng container)

style.css

```
.container{
    width: 1140px;
    margin: 0 auto;
}
.content{
    background-color: palegoldenrod;
    width: 75%;
    float: left;
}
.avatar{
```

```
width: 25%;  
background-color: greenyellow;  
float: left;  
}  
  
.info{  
  background-color: aquamarine;  
}
```

Kết quả sau khi sử dụng thuộc tính float như sau:



Clearfix Class

Clearfix class trong HTML

Clearfix class nói ngắn gọn là 1 kỹ thuật giúp cho element không chứa thuộc tính float, không phải chịu ảnh hưởng của các phần tử chứa thuộc tính float gần nó.

Lưu ý: Ở phần này các bạn không nhất thiết phải hiểu quá sâu, chỉ cần biết cách áp dụng là được

Trong ví dụ trên, nếu để ý bạn sẽ thấy nội dung phần **Info** cũng trôi theo phần **avatar** dù không gán thuộc tính **float: left**.



Để giải quyết vấn đề này, chúng ta sẽ sử dụng **Clearfix Class**:

index.html

```
<div class="avatar">
  
  <h4>MY NAME IS ABC</h4>
</div>
<div class="clearfix"> </div>
```

Thuộc tính CSS trong clearfix

Có nhiều cách tạo thuộc tính CSS cho class clearfix, dưới đây là mình giới thiệu cho các bạn 1 trong số đó.

```
.clearfix::after{
  content: "";
  display: table;
  clear: both;
}
```

Sau khi áp dụng clearfix class, bạn có thể thấy phần **info** không còn bị ảnh hưởng bởi thuộc tính float của **avatar**



Bạn có thể tham khảo chi tiết hơn về clearfix class tại:

https://www.w3schools.com/css/css_float.asp

Project tham khảo

Nếu việc thực hành theo hướng dẫn không diễn ra suôn sẻ như mong muốn. Bạn cũng có thể tải xuống **PROJECT THAM KHẢO** ở link bên dưới!



Kết luận

Ở bài này chúng ta đã cùng nhau tìm hiểu về cách sử dụng ID và Class trong HTML và cách sử dụng thuộc tính float trong CSS.

Ở bài sau chúng ta sẽ tìm hiểu [THUỘC TÍNH POSITION TRONG CSS](#) và hoàn thiện trang Blog của chúng ta.

Cảm ơn các bạn đã theo dõi bài viết. Hãy để lại bình luận và góp ý của mình để giúp phát triển bài viết tốt hơn. Đừng quên: **"Luyện tập - Thử thách - Không ngại khó"**.