**CSS (2) - Sự kế thừa, Xếp lớp**

1.1       Sự kế thừa (Inheritance)

Trong CSS cũng có tình huống kiểu như vậy, những gì chúng ta **định dạng cho phần-tử-cha**, thì mặc định, chúng **cũng được định dạng cho các phần-tử-con**. Tổng quát hơn, khi chúng ta **định dạng CSS cho một phần tử,** thì một số **định dạng cũng áp dụng luôn cho các phần tử chứa trong nó**.

Ví dụ,

[CSS]

        p {

            font-size: 41px;

            color: red;

        }

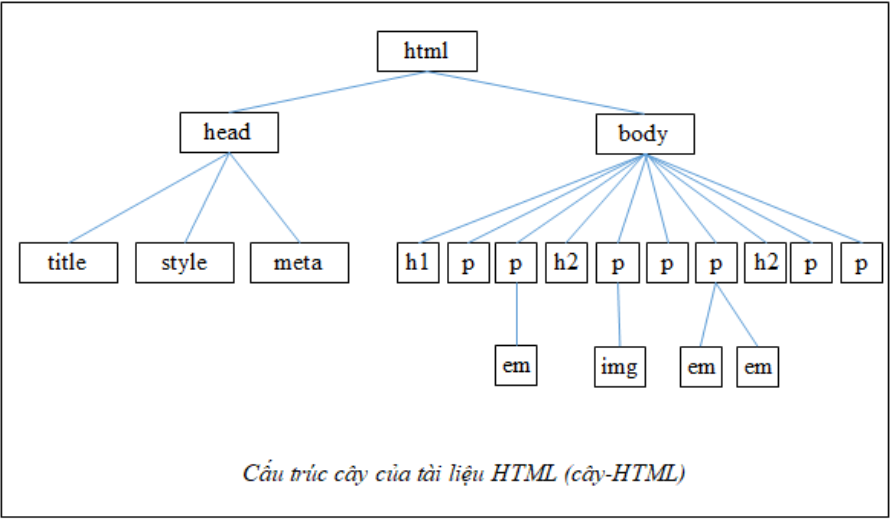
[HTML]

<p>Cuối tuần đi đá banh, giúp thêm <em>năng lượng</em> để làm việc</p>

Như bạn thấy, chúng ta chỉ định dạng cho phần tử *p* có cỡ chữ 41px và màu đỏ, chứ không định dạng cho phần tử *em*. Tuy nhiên, do *em* nằm trong *p*nên nội dung của *em* cũng có cỡ chữ 41px và màu đỏ.

hần tử *em* đã kế thừa thuộc tính định dạng font từ phần tử *p*.

***Cấu trúc của tài liệu***

 Như đã biết, tài liệu HTML đã ngầm chứa trong nó một cấu trúc phân cấp (hierarchy).

***Mối quan hệ giữa các nút trong cây-HTML***

***Có thuộc tính được kế thừa, có thuộc tính không***

Khi thiết lập định dạng font cho phần tử *p*, nó sẽ tác động lên mọi phần tử *p* của tài liệu HTML và các phần tử văn bản kiểu *inline* (*em*) nằm trong *p*. Có nghĩa là, phần tử**em đã kế thừa thuộc tính định dạng fon**t từ phần tử *p*.

Tuy nhiên, phần tử *img* cũng nằm trong *p* nhưng **sẽ không kế thừa** thuộc tính định dạng font**, vì hình ảnh thì không có thuộc tính font.**

## **Khi thiết lập thuộc tính cho một phần tử, nó sẽ ghi đè (override) lên thuộc tính được kế thừa tương ứng.** Ví dụ, với phần tử *em* là con của phần tử *p*, nếu thiết lập phông chữ cho *p* là *sans-serif* và thiết lập phông chữ cho *em* là *serif*  thì thuộc tính được áp dụng cho *em* sẽ là *serif*.

1.2       Kĩ thuật xếp lớp (Cascading)

Kĩ thuật CSS cho phép viết mã định dạng ở 3 nơi khác nhau  vậy sẽ có tình trạng nhiều định dạng cùng áp lên một phần tử. Hoặc do tính kế thừa trong CSS, cũng sẽ dẫn tới tình trạng tương tự

Để giải quyết tình huống: một phần tử HTML có nhiều định dạng cùng lúc, CSS sẽ sử dụng kĩ thuật xếp lớp (cascading).

Kĩ thuật xếp lớp là cách sắp xếp thứ tự các định dạng CSS (chỉ-dẫn-CSS) lên một phần tử HTML. **Mỗi chỉ-dẫn-CSS được đánh trọng số ưu tiên (weight)** hay độ ưu tiên, chỉ-dẫn-CSS nào có **độ ưu tiên** **thấp** **sẽ định dạng lên phần tử HTML trước**; chỉ-dẫn-CSS có **độ ưu tiên cao** hơn sẽ **định dạng đè lên** (overwrite) chỉ-dẫn-CSS **có độ ưu tiên thấp hơn**

Độ ưu tiên được xác định dựa trên 3 yếu tố:

– Chỉ-dẫn-CSS từ đâu đến (priority of style source)

– Tính rõ ràng, chính xác của chỉ-dẫn-CSS (specificity)

– Thứ tự thực thi chỉ-dẫn-CSS (rule order)

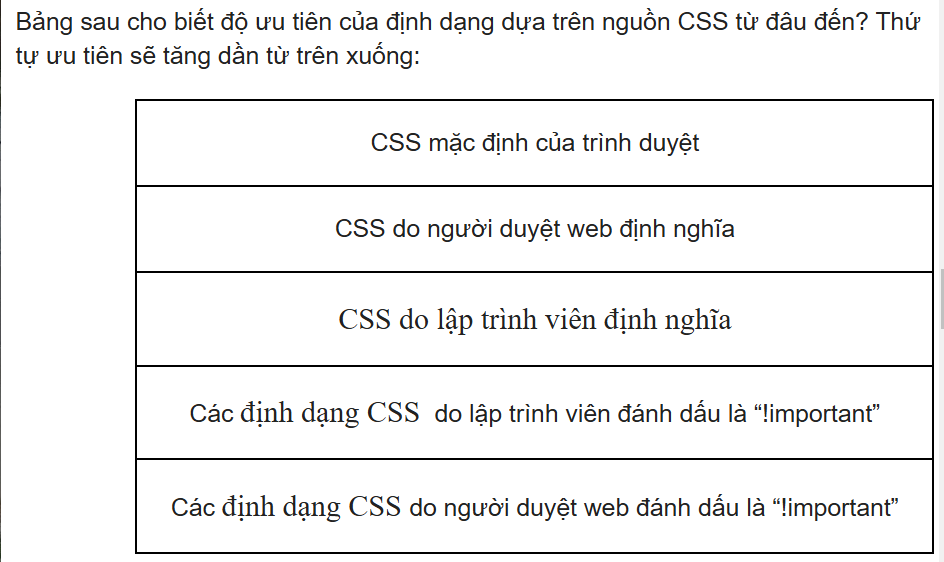
**Chỉ-dẫn-CSS từ đâu đến**

*CSS của trình duyệt*

Nếu chúng ta không viết mã CSS để định dạng cho tài liệu HTML, thì trình duyệt sẽ lấy định dạng mặc định của nó để hiển thị trang web. W3C gọi kiểu định dạng này là user agent style sheet. Mỗi trình duyệt (ví dụ Chrome, Firefox) sẽ có bộ định dạng mặc định khác nhau.

*CSS do người dùng tạo ra*

Mã này sẽ có độ ưu tiên cao hơn mã CSS của người dùng và của trình duyệt.

****

**Tính rõ ràng, chính xác của chỉ-dẫn-CSS**

 <style>

        h1 {

            color: red;

        }

        #tieu-de {

            color: yellow;

        }

    </style>

</head>

<body>

    <h1 *id*="tieu-de">Hi bac Teo!</h1>

</body>

Trong tình huống này, CSS quy định: khi hai (hay nhiều) chỉ dẫn trong một tài liệu CSS xung đột với nhau, thì sẽ dựa vào loại của bộ chọn (type of selector) để xác định độ ưu tiên.

Bộ chọn nào càng rõ ràng, chính xác, và cụ thể (specificity) thì có độ ưu tiên cao hơn.

Như ở ví dụ đang xét, bộ chọn *h1* sẽ áp dụng cho tất cả mọi phần tử *h1*, như vậy ít có tính "cụ thể" bằng bộ chọn dựa trên định danh (#tieu-de). Kết quả là bộ chọn dựa trên định danh sẽ có độ ưu tiên cao hơn, và dòng chữ “Hi bac Teo!” sẽ có màu vàng.

**Thứ tự thực thi chỉ-dẫn-CSS (rule order)**

<style>

        h1 { color: red; }

        h1 { color: yellow; }

        h1 { color: green; }

    </style>

</head>

<body>

    <h1 *id*="tieu-de">Hi bac Teo!</h1>

</body>

Khi đó, CSS đưa ra quy tắc *“cái nào gần đối tượng (HTML) cần định dạng hơn sẽ có độ ưu tiên cao hơn (cái cuối cùng sẽ có độ ưu tiên cao nhất)”*. Vậy, *h1* sẽ có màu xanh

Nếu để ý sẽ thấy, khi thực thi mã nguồn, trình duyệt sẽ thực hiện từ trên xuống dưới, từ trái sang phải. Mã CSS luôn xuất hiện trước phần tử HTML. Nên mã CSS càng gần phần tử HTML hơn càng được thực thi sau, và mã CSS cuối cùng sẽ định dạng đè lên các mã CSS trước đó.