

CSS EFFECT

Mục tiêu

- Sử dụng được CSS Combinators
- Sử dụng được CSS Pseudo-Classes
- Sử dụng được CSS Pseudo-Elements
- Sử dụng được CSS Opacity/Transparency
- Sử dụng được CSS Transitions
- Sử dụng được CSS Animations

CSS Combinators

- CSS Combinators giải thích mối quan hệ giữa các bộ chọn
- Có bốn tổ hợp khác nhau trong CSS:
 - bộ chọn con, cháu (dấu cách)
 - bộ chọn con (>)
 - bộ chọn anh/chị/em kế cận (+)
 - bộ chọn anh/chị/em chung (~)

CSS Combinators – Ví dụ

```
div p { background-color: yellow; }
```

```
div > p { background-color: yellow; }
```

```
div + p { background-color: yellow; }
```

```
div ~ p { background-color: yellow; }
```

CSS Pseudo - Classes

- Một lớp giả được sử dụng để xác định trạng thái đặc biệt của một phần tử.
- Ví dụ, nó có thể được sử dụng để:
 - Tạo kiểu cho một phần tử khi người dùng di chuột qua nó
 - Tạo kiểu cho các liên kết được truy cập và không được truy cập khác nhau
 - Tạo kiểu cho một phần tử khi nó được focus

CSS Pseudo-Class – Cú pháp

```
selector:pseudo-class  
{  
    property: value;  
}
```

CSS Pseudo-Elements

- Phần tử giả CSS được sử dụng để tạo kiểu cho các phần cụ thể của phần tử.
- Ví dụ, nó có thể được sử dụng để:
 - Định kiểu chữ cái đầu tiên hoặc dòng, của một phần tử
 - Chèn nội dung trước hoặc sau nội dung của một phần tử

CSS Pseudo-Elements – Cú pháp

```
selector::pseudo-element  
{  
    property: value;  
}
```


CSS Opacity

- Thuộc tính opacity chỉ định độ mờ đục / trong suốt của một phần tử.
- Thuộc tính opacity có thể có giá trị từ 0.0 – 1.0. Giá trị càng thấp, càng trong suốt.

CSS Opacity – Ví dụ



opacity 0.2

```
img { opacity: 0.2; }
```



opacity 0.5

```
img { opacity: 0.5; }
```



opacity 1
(default)

```
img { opacity: 1; }
```

CSS Transitions

- CSS Transitions cho phép bạn thay đổi các giá trị thuộc tính một cách mượt mà, trong một khoảng thời gian nhất định.
- Trong bài này, bạn sẽ tìm hiểu về các thuộc tính sau:
 - transition
 - transition-delay
 - transition-duration
 - transition-property
 - transition-timing-function

CSS Transition – Ví dụ

```
div {  
    width: 100px;  
    height: 100px;  
    background: red;  
    transition: width 2s, height 4s;  
}  
div:hover { width: 300px; height: 300px; }
```

CSS Animations

- CSS cho phép tạo hoạt ảnh cho các phần tử HTML mà không cần sử dụng JavaScript hoặc Flash!
- Trong bài này, bạn sẽ tìm hiểu về các thuộc tính sau:
 - @keyframes
 - animation-name
 - animation-duration
 - animation-delay
 - animation-iteration-count
 - animation-direction
 - animation-timing-function
 - animation-fill-mode
 - animation

CSS Animations – Ví dụ

```
div {  
    width: 100px;  
    height: 100px;  
    background-color: red;  
    animation-name: example;  
    animation-duration: 4s; }  
  
@keyframes example {  
    from {background-color: red;}  
    to {background-color: yellow;}  
}
```

[Thực hành] Text in Transparent Box

[Thực hành] Tạo xung hướng chuyển đổi vuông và tròn

[Thực hành] Tạo xung hướng tròn