**Đặt biến trong CSS:**

:root {

--text-color: red;

}

* Dùng để đổi giá trị trùng nhiều màu. Khi nào muốn đổi chỉ cần đổi lại màu của biến.

VD:

h1 {

color: var(--text-color);

}

**Box-sizing:** Dùng để cố định khung khi muốn thêm padding, boder.

box-sizing: border-box;

**Background-image**

background-image: linear-gradient(0, #333, #ccc);

0: góc xoay (180deg)

Màu từ #333 sang #ccc tùy vào góc xoay

background-image: linear-gradient(0, rgba(255, 255, 0, 0.5), rgba(0, 0, 255, 0.5), url(image);

Cho nhìn thấy hình ảnh với màu nền trong suốt.

Rgba: red, green, blue, anpha.

Anpha: độ trong suốt của màu.

**CSS pseudo-elements (Phần tử giả)**

<style>

.box::**before**{ //**after**

content: “Có thể viết chữ vào!!”;

display: block;

}

</style>

<body>

<div class= “box”> </div>

</body>

**display**: block (thuộc tính để cho nó là dạng khối).

Mặc định thẻ Div là 1 khối còn before thì không có nên phải thêm thuộc tính display: block vào.

<style>

h1::**selection**{

background-color: red;

}

</style>

<body>

<h1> Xin Chào!!! </h1>

</body>

selection: khi bôi chữ sẽ hiển ra màu đỏ.

**Positison:** Thiết lập vị trí

**CSS Position: Relative**

Là vị trí tương đối, sẽ lấy chính mình làm gốc tọa độ và không phụ thuộc vào đối tượng nào khác ở bên ngoài.

Có 4 thuộc tính: **top**, **right**, **bottom** và **left** (Đẩy đối tượng theo hướng tùy ý).

**CSS Position: Absolute(vị trí tuyệt đối)**

Bị phụ thuộc vào **thẻ cha gần nhất** có thuộc tính **Position (Positon nào cũng được)** để lấy thẻ cha đó làm gốc tọa độ.

Khi có 2 đối tượng 1 đối tượng con và 1 đối tượng cha. Nếu muốn đối tượng con di chuyển xung quanh đối tượng cha thì sử dụng.

Có 4 thuộc tính: **top**, **right**, **bottom** và **left** (Đẩy đối tượng theo hướng tùy ý – có thể âm).

* *Làm phủ mờ mờ trang web (lớp con phủ toàn bộ lớp cha)*

{

background-color: rgba(255, 255, 255, 0.5);

position: absolute;

top: 0;

bottom: 0;

right: 0;

left: 0

}

**CSS Position: Fixed**

Tạo ra phần cố định khi kéo sẽ đi theo.

**CSS Plexbox sample**

