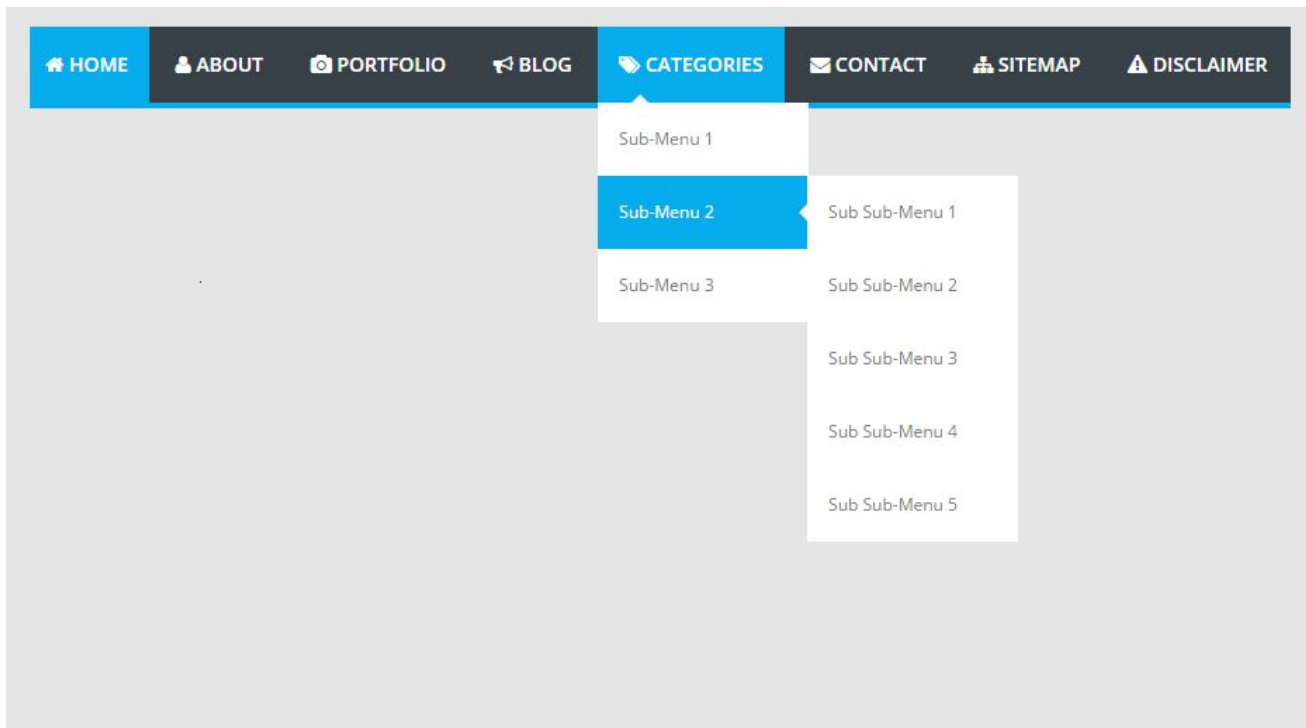




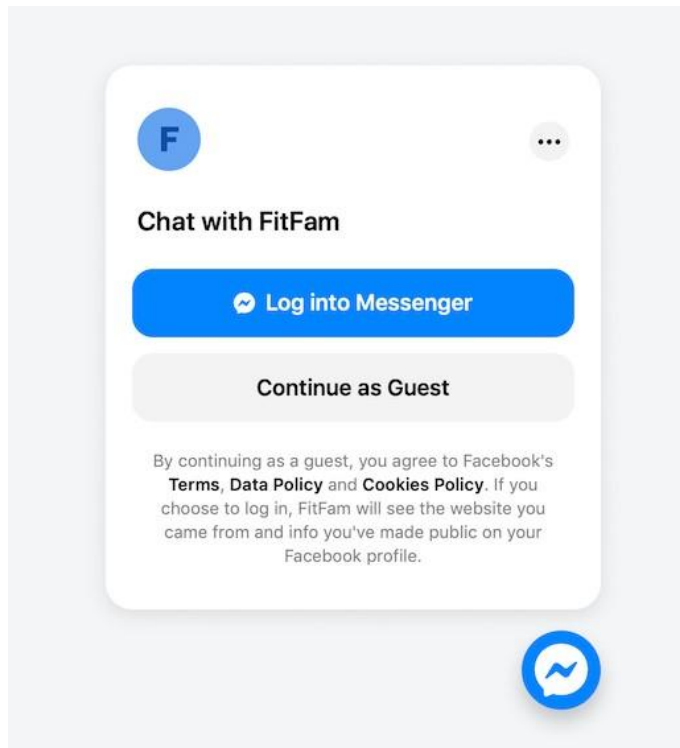
# CSS Position

Ba Nguyễn

# CSS Position



# CSS Position



# CSS Position

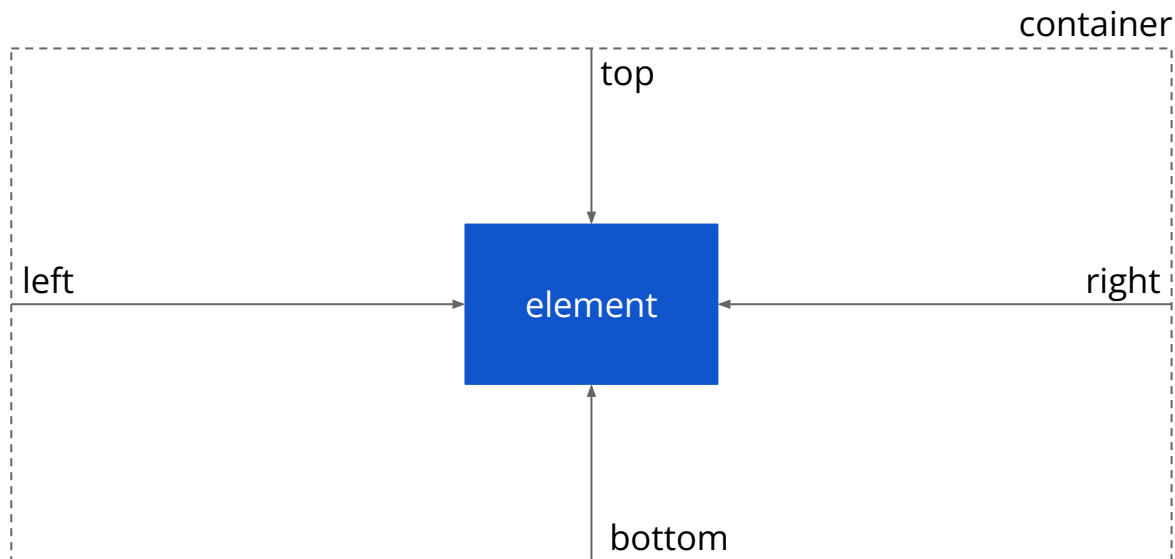
Thuộc tính **position** chỉ định cách một phần tử HTML được định vị trên trình duyệt. Nó đi kèm với các thuộc tính **top**, **right**, **bottom**, **left** và **z-index** để chỉ định vị trí hiển thị của phần tử.

Các giá trị có thể đặt cho thuộc tính **position**:

```
.element {  
    position: static;  
    position: relative;  
    position: absolute;  
    position: fixed;  
    position: sticky;  
}
```

# CSS Position

Các thuộc tính `top`, `right`, `bottom`, `left` đặt vị trí hiển thị của phần tử theo hướng tương ứng.



# z-index

Các phần tử được định vị với position sẽ tạo một ngăn xếp (*stack*), các phần tử xếp chồng lên nhau.

Thuộc tính **z-index** được sử dụng để chỉ định vị trí hiển thị của một phần tử trong ngăn xếp đó (*stacking order*), các phần tử có giá trị **z-index** cao được hiển thị bên trên các phần tử khác.

💡 Các phần tử có chung phần tử cha tạo thành một nhóm ngăn xếp (*stacking context*), thuộc tính **z-index** chỉ áp dụng với các phần tử trong cùng nhóm đó.



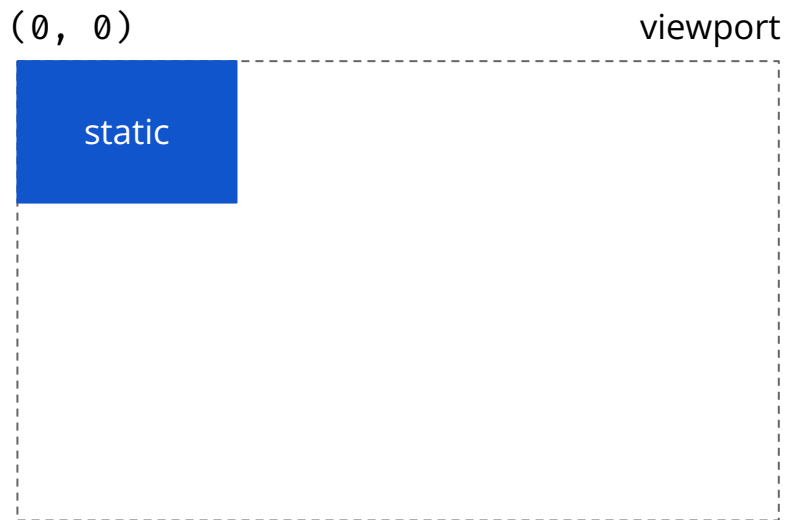
# Static

Mọi phần tử đều có vị trí hiển thị mặc định trên trang theo *luồng* thông thường, mặc định thuộc tính `position` có giá trị là `static`. Với các phần tử có `position: static`, các thuộc tính `top`, `right`, `bottom`, `left` và `z-index` không có tác dụng.

```
.static {  
    position: static;  
    /* top, right, bottom, left, z-index */  
}
```

# Static

Mọi phần tử đều có vị trí hiển thị mặc định trên trang theo *luồng* thông thường, mặc định thuộc tính `position` có giá trị là `static`. Với các phần tử có `position: static`, các thuộc tính `top`, `right`, `bottom`, `left` và `z-index` không có tác dụng.





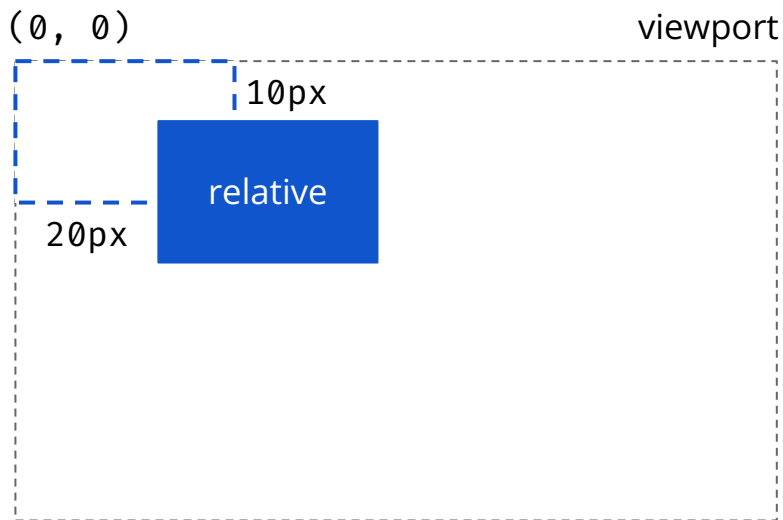
# Relative

Với **position: relative**, vị trí ban đầu (*gốc*) của phần tử vẫn giữ nguyên trong *luồng* hiển thị, tuy nhiên, có các thuộc tính **top**, **right**, **bottom**, **left** và **z-index** có thể được áp dụng. Vị trí hiển thị của phần tử sẽ dịch chuyển so với vị trí ban đầu tùy theo giá trị được đặt.

```
.relative {  
    position: relative;  
    top: 10px;  
    left: 20px;  
}
```

# Relative

Với **position: relative**, vị trí ban đầu (gốc) của phần tử vẫn giữ nguyên trong *luồng* hiển thị, tuy nhiên, có các thuộc tính **top**, **right**, **bottom**, **left** và **z-index** có thể được áp dụng. Vị trí hiển thị của phần tử sẽ dịch chuyển so với vị trí ban đầu tùy theo giá trị được đặt.



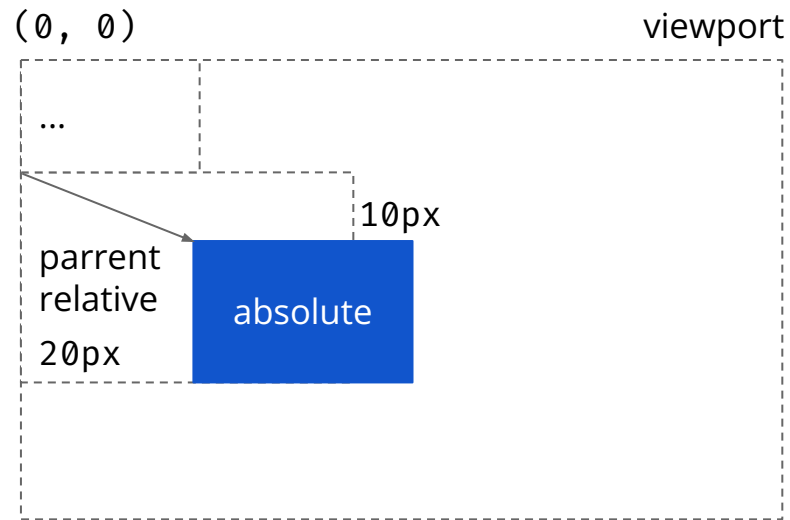
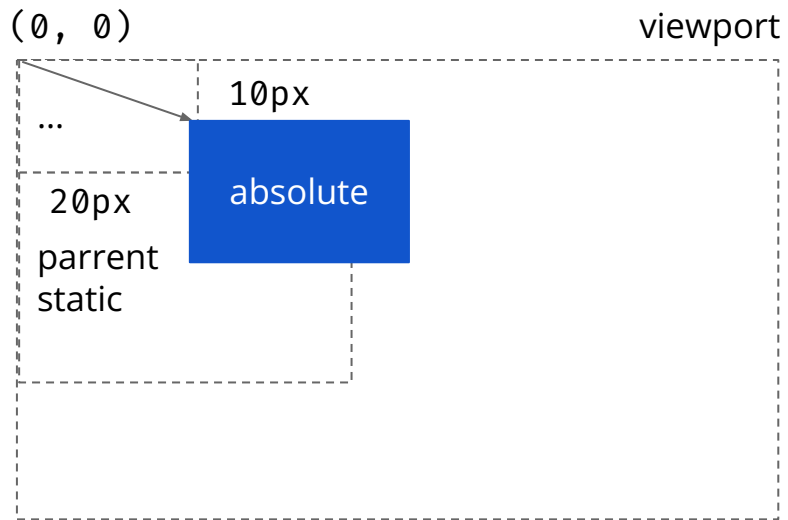
# Absolute

Với `position: absolute`, vị trí và kích thước của phần tử trong *luồng* hiển thị bị xóa bỏ.

Các thuộc tính `top`, `right`, `bottom`, `left` sẽ định vị vị trí của phần tử theo *phần tử cha gần nhất* có `position` là một trong các giá trị: `relative`, `absolute`, `fixed`, `sticky` hoặc theo khung nhìn của trình duyệt nếu không có phần tử cha nào có `position` hợp lệ.

```
.absolute {  
    position: absolute;  
    top: 10px;  
    left: 20px;  
}
```

# Absolute



# Fixed

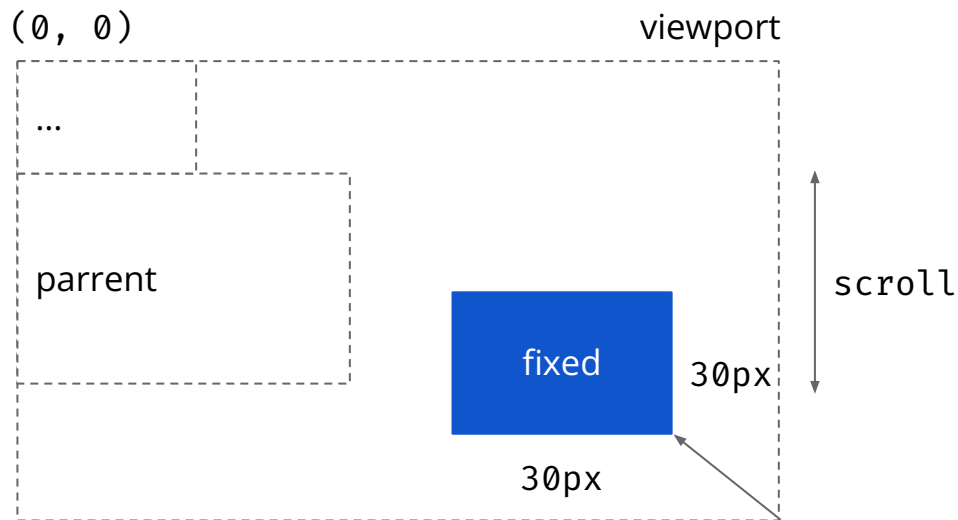


Với **position: fixed**, vị trí và kích thước của phần tử trong *luồng* hiển thị bị xóa bỏ, tương tự **absolute**. Tuy nhiên, khác với **absolute**, các thuộc tính **top**, **right**, **bottom**, **left** đặt vị trí của phần tử cố định theo khung nhìn của trình trình duyệt, và không bị ảnh hưởng khi cuộn trang.

```
.fixed {  
    position: fixed;  
    bottom: 30px;  
    right: 30px;  
}
```

# Fixed

Với **position: fixed**, vị trí và kích thước của phần tử trong *luồng* hiển thị bị xóa bỏ, tương tự **absolute**. Tuy nhiên, khác với **absolute**, các thuộc tính **top**, **right**, **bottom**, **left** đặt vị trí của phần tử cố định theo khung nhìn của trình trình duyệt, và không bị ảnh hưởng khi cuộn trang.

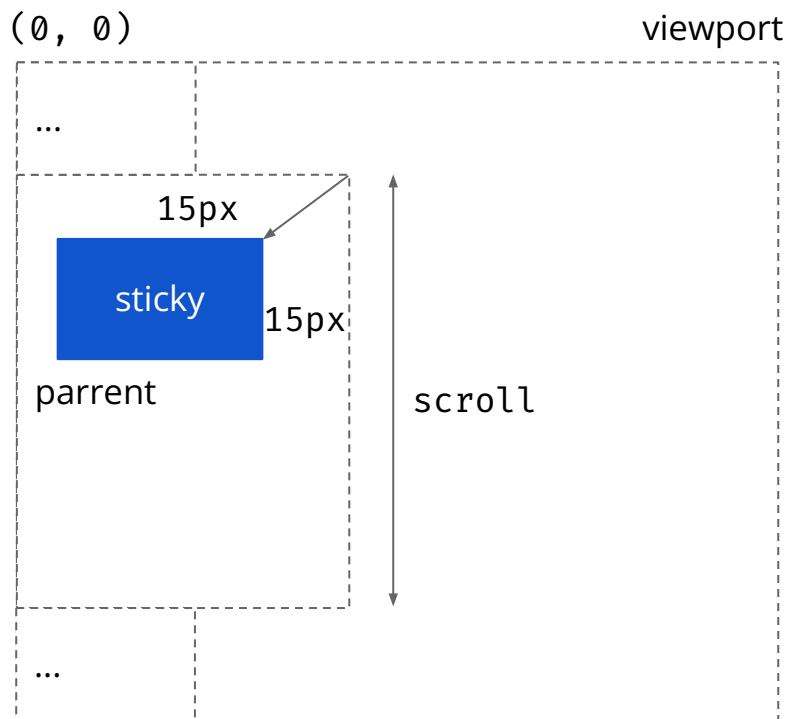


# Sticky

Với `position: sticky`, vị trí và kích thước của phần tử được đặt bình thường trong luồng hiển thị, chỉ khi vị trí cuộn của khung nhìn trong *ngưỡng chỉ định* (đính với phần tử cha), nó sẽ có vị trí cố định `fixed` được đặt bởi các thuộc tính `top`, `right`, `bottom`, `left`

```
.sticky {  
    position: sticky;  
    top: 15px;  
    right: 15px;  
}
```

# Sticky





# Note

---

💡 `position: absolute` hay `fixed` có thể bị ảnh hưởng bởi các phần tử cha có thuộc tính `transform`, `perspective` hoặc `filter` có giá trị khác `none`

💡 `position: sticky` chỉ hoạt động khi không có phần tử cha (hoặc cao hơn) có thuộc tính `overflow` có giá trị là: `hidden`, `scroll`, `auto` hoặc `overlay`

# Exercise



Tạo một trang web với các yêu cầu sau:

- Có menu đa cấp, sử dụng `:hover`, `transition` tạo hiệu ứng ẩn hiện khi menu khi di chuột qua
- Menu luôn hiển thị phía trên cùng của trình duyệt (sử dụng `position: fixed` hoặc `sticky`)
- Có nút back to top (sử dụng `position: fixed`) ở góc dưới bên phải màn hình, khi click thì cuộn về đầu trang (`a href="#"`)
- Nội dung/chủ đề trên trang web tùy ý
- Nội dung trang căn giữa màn hình
- Trang sử dụng bố cục 2 hoặc 3 cột
- Style đẹp