


CSS
Cascading Style Sheets

Class # 3

CSS Properties(Float)

In CSS, the float property is used to position an element along the left or right side of its containing block, allowing other content to flow around it.

Usage:

- **float: left;:** Floats the element to the left side of its containing block, allowing content to flow around it on the right side.
- **float: right;:** Floats the element to the right side of its containing block, allowing content to flow around it on the left side.
- **float: none; (default):** Disables floating for the element, causing it to be positioned within the normal document flow.

CSS Properties(Float)

Examples:

```
img {
  float: left; /* Floats the image to the left side of its containing block */
}

div {
  float: right; /* Floats the div to the right side of its containing block */
}
```

When an element is floated, it is taken out of the normal flow of the document, and subsequent elements will wrap around it. This can be useful for creating multi-column layouts or positioning elements within a larger container.

CSS Properties(Clear)

In CSS, the clear property is used to control the behavior of an element in relation to floated elements that appear before it. It determines if an element should be pushed down (cleared) to start on a new line after floated elements.

Usage:

- **clear: left;:** The element will be pushed down to start on a new line after any left-floated elements.
- **clear: right;:** The element will be pushed down to start on a new line after any right-floated elements.
- **clear: both;:** The element will be pushed down to start on a new line after any left or right-floated elements.
- **clear: none; (default):** No clearing is performed, and the element will appear alongside any floated elements.

CSS Properties(Clear)

Examples:

```
div {
  clear: left; /* Clears any left-floated elements that appear before the div */
}

p {
  clear: both; /* Clears both left and right-floated elements that appear before the paragraph */
}
```

The clear property is typically used on block-level elements to ensure proper layout and prevent content from wrapping around floated elements when necessary.

CSS Properties(Position)

In CSS, the position property is used to specify the positioning behavior of an element within its containing element or the document.

Usage:

- **static (default):** Elements are positioned according to the normal flow of the document. This is the default value, and elements with position: static; are not affected by the other positioning properties (top, right, bottom, left, z-index).
- **relative:** Elements are positioned relative to their normal position in the document flow. When using position: relative;, you can use the top, right, bottom, and left properties to adjust the position of the element relative to its normal position.

CSS Properties(Position)

Usage:

- **absolute:** Elements are positioned relative to the nearest positioned ancestor (parent or ancestor element with position set to anything other than static). If no positioned ancestor is found, the element is positioned relative to the initial containing block (usually the viewport). Absolute positioned elements are removed from the normal flow of the document.
- **fixed:** Elements are positioned relative to the initial containing block (usually the viewport), and they remain fixed even when the page is scrolled. Fixed positioned elements are also removed from the normal flow of the document.
- **sticky:** Elements are positioned based on the user's scroll position. They are positioned relative to their containing block until a specified scroll threshold is reached, at which point they behave like position: fixed;.

CSS Properties(Position)

Examples:

```
div {
  position: relative; /* Positions the div relative to its normal position */
}
h1 {
  position: absolute; /* Positions the h1 element relative to its positioned ancestor */
  top: 50px; /* Moves the h1 50 pixels from the top */
  left: 20px; /* Moves the h1 20 pixels from the left */
}
footer {
  position: fixed; /* Positions the footer element relative to the viewport */
  bottom: 0; /* Places the footer at the bottom of the viewport */
  width: 100%; /* Sets the width of the footer to 100% of the viewport width */
}
```

CSS Properties

In CSS, the properties **top, bottom, left, and right** are used to position elements precisely when the position property is set to relative, absolute, or fixed.

Usage:

- **top**: Specifies the distance between the top edge of an element and the top edge of its containing element or the viewport. It is used to move an element down from its normal position.

- **bottom:** Specifies the distance between the bottom edge of an element and the bottom edge of its containing element or the viewport. It is used to move an element up from its normal position.

CSS Properties

Usage:

- **left:** Specifies the distance between the left edge of an element and the left edge of its containing element or the viewport. It is used to move an element to the right from its normal position.
- **right:** Specifies the distance between the right edge of an element and the right edge of its containing element or the viewport. It is used to move an element to the left from its normal position.

CSS Properties

Examples:

```
div {
  position: relative;
  top: 20px; /* Moves the div 20 pixels down from its normal position */
  left: 10px; /* Moves the div 10 pixels to the right from its normal position */
}

h1 {
  position: absolute;
  top: 50%; /* Positions the h1 element vertically centered relative to its containing element */
  left: 0; /* Positions the h1 element flush against the left edge of its containing element */
}
```