

Layout

Float

— — —

La proprietà float è stata introdotta per implementare layout che coinvolgono un'immagine fluttuante all'interno di una colonna di testo, con il testo che si avvolge a sinistra o a destra di essa.

Rimane possibile rendere mobile qualsiasi cosa, non solo le immagini.

La proprietà float indica a un elemento di "fluttuare" nella direzione specificata. All'immagine in questo esempio viene indicato di fluttuare in alto verso sinistra per poter "avvolgere" gli elementi di pari livello. Puoi indicare a un elemento di galleggiare left, right, none o inherit (eredita il valore dell'elemento genitore).

Float CSS

Box

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla luctus aliquam dolor, eu lacinia lorem placerat vulputate. Duis felis orci, pulvinar id metus ut, rutrum luctus orci. Cras porttitor imperdiet nunc, at ultricies tellus laoreet sit amet.

Sed auctor cursus massa at porta. Integer ligula ipsum, tristique sit amet orci vel, viverra egestas ligula. Curabitur vehicula tellus neque, ac ornare ex malesuada et. In vitae convallis lacus. Aliquam erat volutpat. Suspendisse ac imperdiet turpis. Aenean finibus sollicitudin eros pharetra congue. Duis ornare egestas augue ut luctus. Proin blandit quam nec lacus varius commodo et a urna. Ut id ornare felis, eget fermentum sapien.

Nam vulputate diam nec tempor bibendum. Donec luctus augue eget malesuada ultrices. Phasellus turpis est, posuere sit amet dapibus ut, facilisis sed est. Nam id risus quis ante semper consectetur eget aliquam lorem. Vivamus tristique elit dolor, sed pretium metus suscipit vel. Mauris ultricies lectus sed lobortis finibus. Vivamus eu urna eget velit cursus viverra quis vestibulum sem. Aliquam tincidunt eget purus in interdum.

https://codepen.io/cristina_iurissevich/pen/JjVdzWK

Clear

Quando utilizziamo la proprietà float e vogliamo che l'elemento successivo sia sotto (non a destra o a sinistra), dovremo utilizzare la proprietà clear che specifica cosa dovrebbe accadere con l'elemento che è accanto a un elemento flottante.

La proprietà clear può avere uno dei seguenti valori:

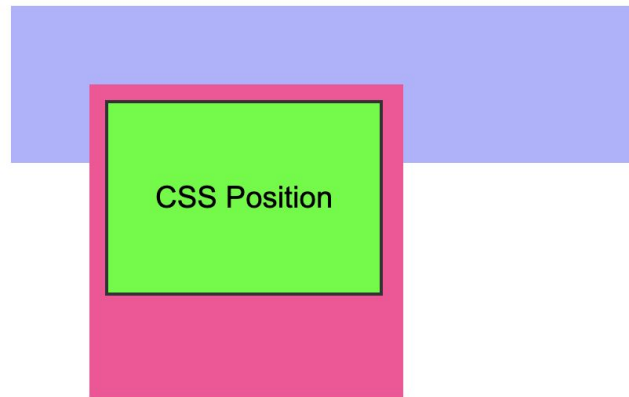
none (valore predefinito), left, right, both (spostato sotto sia gli elementi flottanti a sinistra che quelli a destra), inherit (eredita il valore di clear dal suo elemento genitore).

Il tuo elemento flottante continuerà a fluttuare, ma l'elemento cancellato apparirà sotto di esso sulla pagina web.

Posizionamento

— — —

Il posizionamento è ciò che determina dove gli elementi appaiono sullo schermo e come appaiono. Possiamo spostare gli elementi in giro e posizionarli con la proprietà CSS position.



https://codepen.io/cristina_iurissevich/pen/NWRVJPV

Può assumere i valori:

- **static** - valore predefinito per un elemento. Gli elementi con posizionamento statico sono mostrati nel normale flusso della pagina
- **relative** - può essere posizionato secondo uno spostamento impostando le proprietà offset: top, right, bottom, left (accettano un valore di lunghezza o una percentuale).
- **absolute** - rimuove l'elemento dal flusso del documento e lo mantiene alle coordinate di origine impostate con le proprietà offset relative al contenitore più vicino non static.
- **fixed** - l'elemento viene rimosso dal flusso della pagina. A differenza del posizionamento assoluto gli elementi sono sempre posizionati in base alla finestra, invece che rispetto al primo contenitore non statico.
- **sticky** - consente di posizionare l'elemento secondo il normale flusso del documento, tenendolo in quella posizione anche quando poi scorriamo la pagina.

Position: relative

Un elemento con `position: relative` è posizionato rispetto alla sua posizione normale.

L'impostazione delle proprietà `top`, `right`, `bottom` e `left` di un elemento posizionato in modo relativo farà sì che sia spostato dalla sua posizione normale.

```
h1.relative {  
  position: relative;  
  left: 20px;  
  border: 5px solid yellow;  
}
```

Position: fixed

Un elemento con `position: fixed;` è posizionato rispetto alla finestra di visualizzazione, perciò rimane presente anche se la pagina viene scrollata.

Le proprietà `top`, `right`, `bottom` e `left` vengono utilizzate per posizionare l'elemento.

```
h1 {  
  position: fixed;  
  bottom: 0;  
  left: 0;  
  width: 300px;  
  border: 5px solid yellow;  
}
```

Position: absolute

Un elemento con `position: absolute` è posizionato rispetto all'elemento posizionato più vicino. Tuttavia, se un elemento posizionato in modo assoluto non ha genitori posizionati, utilizza l'elemento `<body>` come riferimento.

Gli elementi posizionati in modo assoluto vengono rimossi dal flusso normale e possono sovrapporsi agli elementi.

```
h1 {  
  position: relative;  
  width: 400px;  
  height: 400px;  
  border: 5px solid yellow;  
}  
  
h2 {  
  position: absolute;  
  top: 80px;  
  right: 0;  
  width: 400px;  
  height: 400px;  
  border: 5px solid red;  
}
```


Position:sticky

Un elemento con `position: sticky` è posizionato in base alla posizione di scorrimento dell'utente.

Un elemento sticky alterna tra il posizionamento relativo e fisso, a seconda della posizione di scorrimento. Viene posizionato in modo relativo fino a quando si raggiunge una determinata posizione nella finestra di visualizzazione, quindi rimane fissato in quel punto (come `position:fixed`).

```
div.sticky {  
  position: sticky;  
  top: 0;  
  background-color: black;  
  border: 5px solid red;  
}
```

Display

La proprietà display di un oggetto determina se e come verrà presentato nel browser.

Può utilizzare i seguenti valori (qui sono elencati solo i principali):

- **block** - gli elementi sono impilati gli uni sugli altri e ogni elemento occupa il 100% della pagina.
- **inline** - valore predefinito di display per tutti gli elementi in CSS. Gli elementi inline non hanno alcun margine o padding applicati. Lo stesso vale per altezza e larghezza.
- **none** - fa scomparire un elemento. Esiste ancora nell'HTML, ma non è visibile nel browser.
- **flex** - crea un genitore flexbox
- **grid** - crea un genitore grid

Z-index

La proprietà z-index imposta esplicitamente un ordine dei livelli per il codice HTML in base allo spazio 3D del browser, ovvero l'asse Z. Questo è l'asse che mostra i livelli più vicini e lontani da te. L'asse verticale sul Web è l'asse Y, mentre l'asse orizzontale è l'asse X.

La proprietà z-index accetta un valore numerico, che può essere un numero positivo o negativo. Gli elementi verranno visualizzati sopra un altro elemento se hanno un valore z-index più alto.

Nel flusso normale, **se imposti un valore specifico per z-index e questo non funziona devi impostare il valore position dell'elemento su un valore diverso da static**. Questo non accade se ti trovi in un contesto flexbox o grid.

https://codepen.io/cristina_iurissevich/pen/NWmGgWY