



Display property and Normal Flow





Topics Covered

- Normal Flow
- Display Property



Normal Flow

Default arrangement of elements on a page, prior to any modifications made to their layout.

When elements are arranged in Flow Layout,

- Inline elements appear in the same direction
- Block elements, on the other hand, appear consecutively.



Example - Code

HTML

```
<!DOCTYPE html>
<html lang="en-us">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport"
content="width=device-width" />
    <title>Unordered Lists</title>
  </head>
  <body>
    <h3> Normal Layout </h3>
    <p>
      Elements are placed in the order in which they
appear in the HTML
      document, and their positions are determined
by their default positioning
      properties.
    </p>
    <ol>
      <li> Item 1 </li>
      <li> Item 2 </li>
      <li> Item 3 </li>
    </ol>
  </body>
</html>
```

CSS

```
/*style.css*/

h3,p,li {
  background-color: brown;
  color: white;
  border: 5px solid yellow;
  padding: 10px;
}
```




Example - Output

Output

Normal Layout

Elements are placed in the order in which they appear in the HTML document, and their positions are determined by their default positioning properties.

Item 1

Item 2

Item 3



Display Property

Altering Default Layout

There are two kinds of display types,

- **Outer** : inline, block, inline-block, none
- **Inner** : flow, flex, grid, etc.h

In this lesson, we will study about **Inline and Block**.



Example HTML to understand display

HTML

```
<!DOCTYPE html>
<html lang="en-us">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width" />
    <title>Inline-Block Elements</title>
  </head>
  <body>
    <div>
      <div class="item1 item">Item 1</div>
      <div class="item2 item">Item 2</div>
      <div class="item3 item">Item 3</div>
    </div>
  </body>
</html>
```



display: block

Creates a block level element, generating line breaks before and after the element in Normal Flow.

CSS

```
.item {  
  display: block;  
  background-color: brown;  
  color: white;  
  border: 5px solid yellow;  
  padding: 10px;  
}
```

Output

Item 1

Item 2

Item 3

In above example, all items are **block** elements that's why, each item is onto a new line and each element takes the whole width available.



display:inline

Creates an inline level element that does not generate line breaks, and starts on the same line in Normal Flow.

CSS

```
.item {  
  display: inline;  
  background-color: brown;  
  color: white;  
  border: 5px solid yellow;  
  padding: 10px;  
}
```

Output



In the above example, we are making **div** elements explicitly inline that's why all elements are aligned on the same line taking only required space.



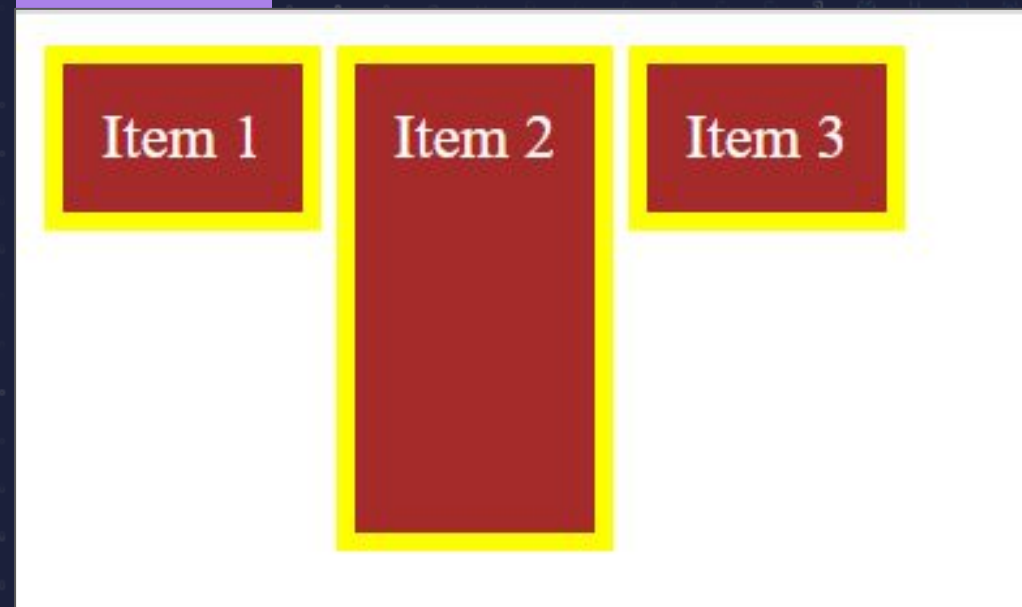
display: inline-block

Creates an element which behave like an **inline** element while retaining some features of a **block**-level element.

CSS

```
.item {  
  display: inline-block;  
  background-color: brown;  
  color: white;  
  border: 5px solid yellow;  
  padding: 10px;  
}  
  
.item2 {  
  height: 100px;  
}
```

Output



In this example, we have changed **item2** height. If it was a pure inline element, then we will not be able to set the height of the element. Since the element is **inline-block**, that's why we are able to use **block** level element properties.



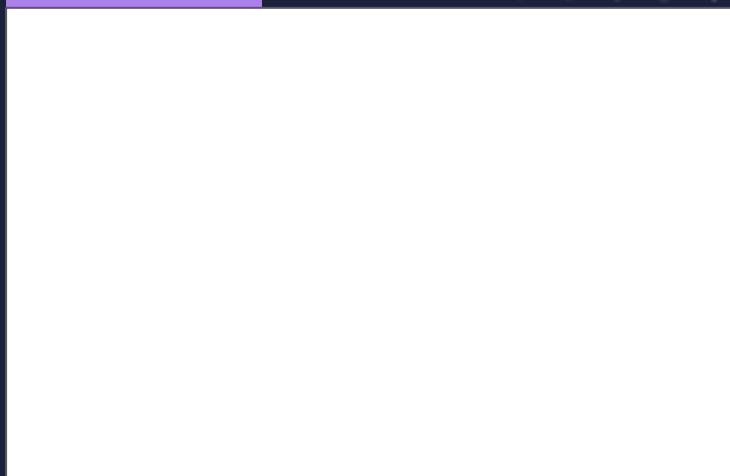
display: none

The display inline-block property in CSS is used to make an element invisible, and the element will not take any space in the document.

CSS

```
.item {  
  display: none;  
  background-color: brown;  
  color: white;  
  border: 5px solid yellow;  
  padding: 10px;  
}
```

Output





▶ THANK YOU ◀