# CSS Display

- description : CSS Display Property, Block-level Elements, Inline Elements
- author : 오션
- email : shlim@repia.com
- lastupdate : 2021-03-15

## Source of the article

- CSS Display

# The display Property

`display`속성은 레이아웃을 조절하는 가장 중요한 CSS 속성입니다.
`display`속성은 한 요소의 표시 여부와 표시 방법을 지정합니다.
모든 HTML 요소들은 타입에 따른 기본 display 값을 가지고 있습니다. 대부분의 요소들의 기본 display 값은 `block` 또는 `inline`입니다.

# Block-level Elements

블록 레벨 요소는 항상 새로운 줄에서 시작하고 사용 가능한 화면의 전체 너비를 차지합니다.

블록 레벨 값을 가지는 요소들

- 
- <h1> ~ <h6>
- <p>
- <form>
- <header>
- <footer>
- <section>

# Inline Elements

인라인 요소는 새로운 줄에서 시작하지 않고 필요한 만큼만 차지합니다.
인라인 값을 가지는 요소들

- 
- <a>
- <img>

# Display: none;

`display: none`속성은 일반적으로 자바스크립트와 함께 사용되어 요소들을 삭제하고 다시 만들지 않고 숨기고 표시합니다.
`<script>`요소는 기본 값으로 `display: none;`을 가집니다.

# Override The Default Display Value (기본 디스플레이 값 무시하기)

언급한 것처럼, 모든 요소에는 기본 디스플레이 값이 있습니다. 그러나 이것을 무시할 수 있습니다.
인라인 요소를 블록 요소로 또는 그 반대로 변경하는 것이, 해당 웹 페이지를 특정 방식으로 보이게 하고 웹 표준을 준수하는 데에 유용할 수 있습니다.
일반적인 예제는 가로 메뉴에 대해 인라인 <li> 요소들을 만드는 것입니다

### 예제

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Display</title>
    <style>
        li {
            display: inline;    /* <li>는 블록요소이나 이 속성이 인라인요소로 변경해준다.*/
    }               /* 인라인 요소로 변경되어 한 라인에 링크 3개가 나타난다 */
```

```
    </style>
</head>

<body>

    <p>Display a list of links as a horizontal menu:</p>

    <ul>
        <li><a href="https://www.w3schools.com/html/default.asp"
target="_blank">HTML</a></li>
        <li><a href="https://www.w3schools.com/css/default.asp"
target="_blank">CSS</a></li>
        <li><a href="https://www.w3schools.com/js/default.asp"
target="_blank">Javascript</a></li>
    </ul>

</body>

</html>
```

Note: 한 요소의 디스플레이 속성을 설정하면 요소의 종류가 아니라 **해당 요소의 표시 방법**만 변경됩니다.
따라서 display: block속성을 가진 인라인 요소는 내부에 다른 블럭 요소를 가질 수 없습니다.
아래의 예제는 블럭 요소로서의 요소들을 표시합니다.

## 예제

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>inline element changed as block element</title>
    <style>
        span {
            display: block;  /* 인라인 요소 span을 블럭 요소로 변경하여 두 줄로 표시된
다.*/
        }
    </style>
</head>

<body>

    <span>A display property with a value of "block" results
in</span><span>a line break between the two elements</span>
```

```
</body>

</html>
```

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Display links as block elements</title>
    <style>
        a {
            display: block;
       /* 인라인요소 <a>를 블록 속성으로 변경하여 3줄로 표시 */
        }
    </style>
</head>

<body>

    <p>Display links s block elements</p>
    <a href="https://www.w3schools.com/html/default.asp">HTML</a>
    <a href="https://www.w3schools.com/css/default.asp">CSS</a>
    <a href="https://www.w3schools.com/js/default.asp">Javascript</a>

</body>

</html>
```

# Hide an Element -display:none or visibility:hidden?

display속성을 none으로 설정하여 한 요소를 숨길 수 있습니다. 해당 요소는 숨겨질 것이고, 해당 웹 페이지는 마치 해당 요소가 그곳에 없는 것처럼 표시될 것입니다.

### 예제

```
<!DOCTYPE html>
```

```
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>visible heading</title>
    <style>
        h1.hidden {
            display: none;   /* hidden 클래스의 h1이 사라지고, 공간도 사라진다. */
        }
    </style>
</head>

<body>

    <h1>This is a visible heading</h1>
    <h1 class="hidden">This is a hidden heading</h1>
    <p>Notice tht the h1 element with display: none; does not take up any
space.</p>

</body>

</html>
```

visibility:hidden;도 요소를 숨깁니다.

하지만, 해당 요소는 여전히 예전처럼 동일한 공간을 차지할 것입니다. 해당 요소는 숨겨지지만 여전히 레이아웃에 영향을 줍니다.

## 예제

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Visibility: hidden;</title>
    <style>
        h1.hidden {
            visibility: hidden;
        }
    </style>
</head>

<body>
```

```
    <h1>This is a visible heading</h1>
    <h1 class="hidden">This is a hidden heading</h1>
    <p>Notice that the hidden heading still takes up space.</p>
</body>

</html>
```

## More Examples

```
<!DOCTYPE html>
<html>
<head>
<style>
.imgbox {
  float: left;
  text-align: center;
  width: 120px;
  border: 1px solid gray;
  margin: 4px;
  padding: 6px;
}

button {
  width: 100%;
}
</style>
</head>
<body>

<h3>Difference between display:none and visiblity: hidden</h3>
<p><strong>visibility:hidden</strong> hides the element, but it still takes
up space in the layout.</p>
<p><strong>display:none</strong> removes the element from the document. It
does not take up any space.</p>

<div class="imgbox" id="imgbox1">Box 1<br>
  <img src="img_5terre.jpg" alt="Italy" style="width:100%">
  <button onclick="removeElement()">Remove</button>
</div>

<div class="imgbox" id="imgbox2">Box 2<br>
  <img src="img_lights.jpg" alt="Lights" style="width:100%">
  <button onclick="changeVisibility()">Hide</button>
</div>

<div class="imgbox">Box 3<br>
  <img src="img_forest.jpg" alt="Forest" style="width:100%">
```

```
  <button onclick="resetElement()">Reset All</button>
</div>

<script>
function removeElement() {
  document.getElementById("imgbox1").style.display = "none";
}

function changeVisibility() {
  document.getElementById("imgbox2").style.visibility = "hidden";
}

function resetElement() {
  document.getElementById("imgbox1").style.display = "block";
  document.getElementById("imgbox2").style.visibility = "visible";
}
</script>

</body>
</html>
```

```
<!DOCTYPE html>
<html>

<head>
    <style>
        #panel,
        .flip {
            font-size: 16px;
            padding: 10px;
            text-align: center;
            background-color: #4CAF50;
            color: white;
            border: solid 1px #a6d8a8;
            margin: auto;
        }

        #panel {
            display: none;
        }
    </style>
</head>

<body>

    <p class="flip" onclick="myFunction()">Click to show panel</p>

    <div id="panel">
        <p>This panel contains a div element, which is hidden by default
```

```
(display: none).</p>
        <p>It is styled with CSS and we use JavaScript to show it (display:
block).</p>
        <p>How it works: Notice that the p element with class="flip" has an
onclick attribute attached to it. When the
            user clicks on the p element, a function called myFunction() is
executed, which changes the style of the div
            with id="panel" from display:none (hidden) to display:block
(visible).</p>
        <p>You will learn more about JavaScript in our JavaScript
Tutorial.</p>
    </div>

    <script>
        function myFunction() {
            document.getElementById("panel").style.display = "block";
        }
    </script>

</body>

</html>
```

## CSS Display/Visibility Properties

| Property | Description |
|---|---|
| display | 요소가 표시되는 방법을 지정합니다. |
| visibility | 요소의 표시 여부를 지정합니다. |

오션,, CSS, Display,

From:
http://125.132.25.164/dokuwiki/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
**http://125.132.25.164/dokuwiki/doku.php?id=wiki:css:css_note:css_display&rev=1615781726**

Last update: **2022/03/10 19:52**