Introduction to CSS3: Part.2

Introduction to Internet and Web

부산대학교 정보·의생명 공학대학
정보컴퓨터공학부

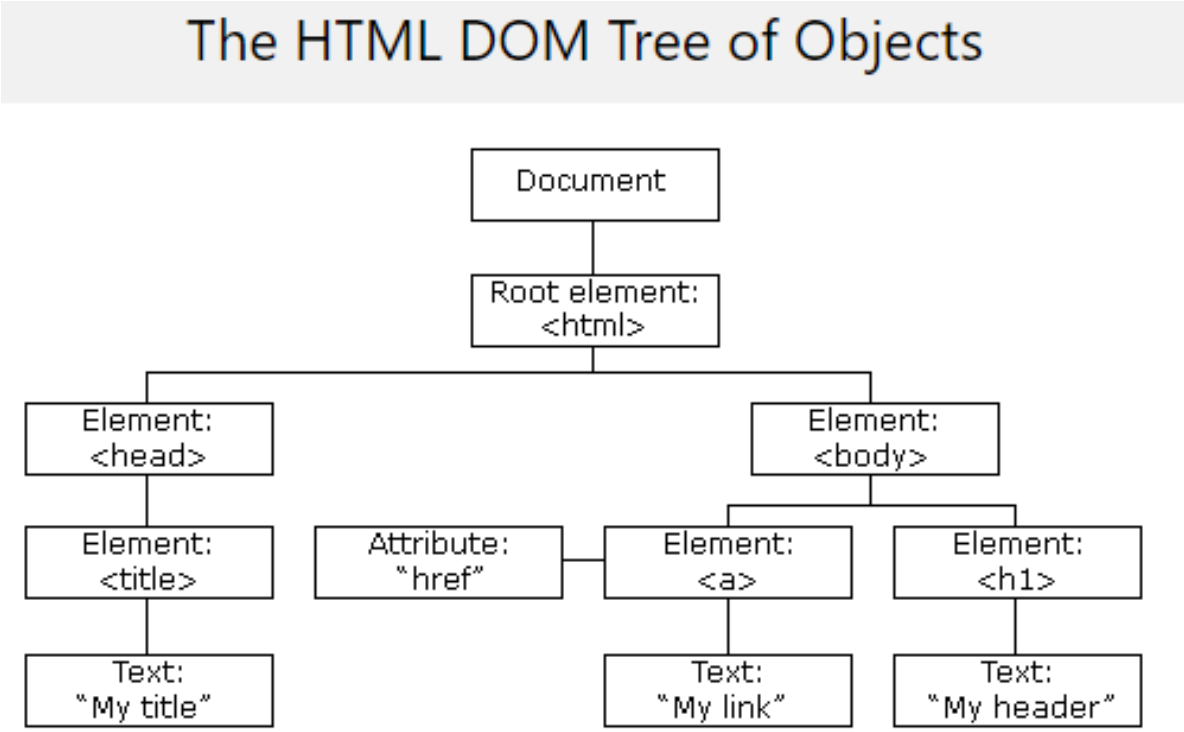
Table of Contents

HTML DOM

HTML DOM

❖ DOM (Document Object Model)

- When a web page is loaded, the browser creates a Document Object Model of the page
- The HTML model is constructed as a tree of Objects

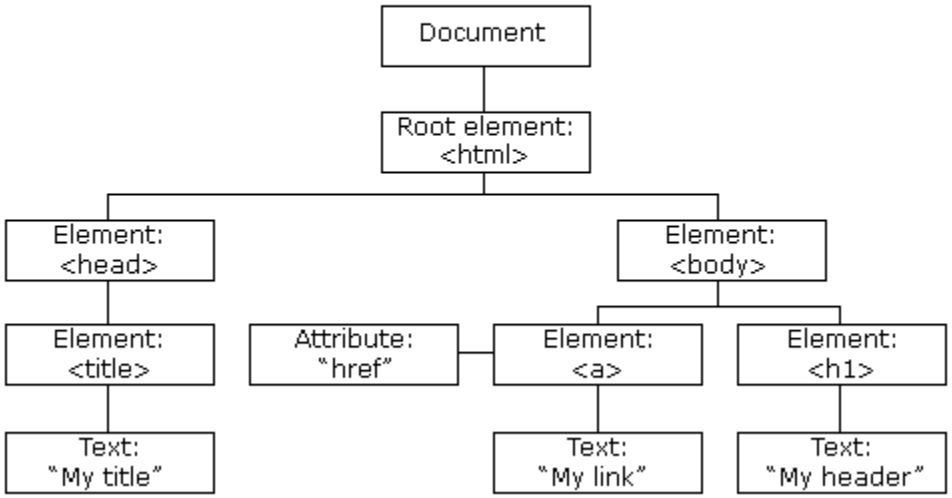


DOM Nodes

❖ **According to the W3C HTML DOM standard, everything in an HTML document is a node.**

- The entire document is a **document node**
- Every HTML element is an **element node**
- The text inside HTML elements are **text nodes**
- Every HTML attribute is an **attribute node**
- All comments are **comment nodes**

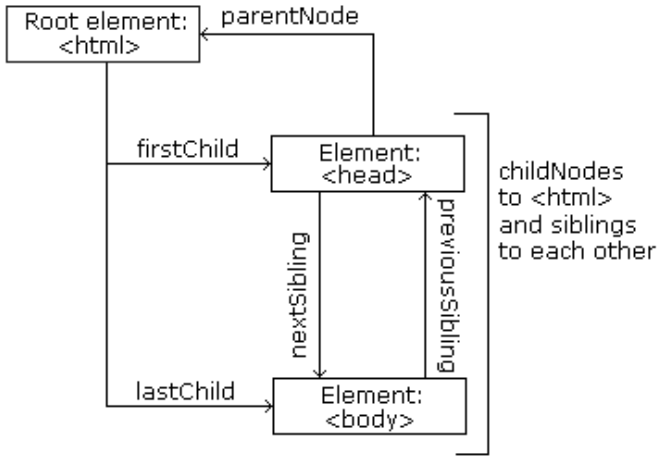
The HTML DOM Tree of Objects



Node Relationships

- **The nodes in the node tree have a hierarchical relationship to each other.**
- **The terms parent, child, and sibling are used to describe the relationships.**
  - In a node tree, the top node is called the root (or root node)
  - Every node has exactly one parent, except the root (which has no parent)
  - A node can have a number of children
  - Siblings (brothers or sisters) are nodes with the same parent

```
<html>

  <head>
    <title>DOM Tutorial</title>
  </head>

  <body>
    <h1>DOM Lesson one</h1>
    <p>Hello world!</p>
  </body>

</html>
```



CSS COMBINATORS

CSS Combinators

- A combinator is something that explains the relationship between the selectors.

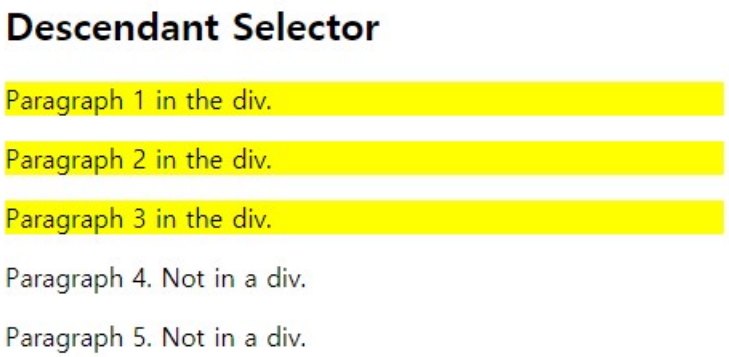
- There are four different combinators in CSS:
  - descendant selector (space)
  - child selector (>)
  - adjacent sibling selector (+)
  - general sibling selector (~)

Example	Example description
div p	Selects all <p> elements inside elements
div > p	Selects all <p> elements where the parent is a element
div + p	Selects the first <p> element that are placed immediately after elements
p ~ ul	Selects every element that are preceded by a <p> element

Descendant Selector

- The descendant selector matches all elements that are descendants of a specified element.

Descendant Selector Example

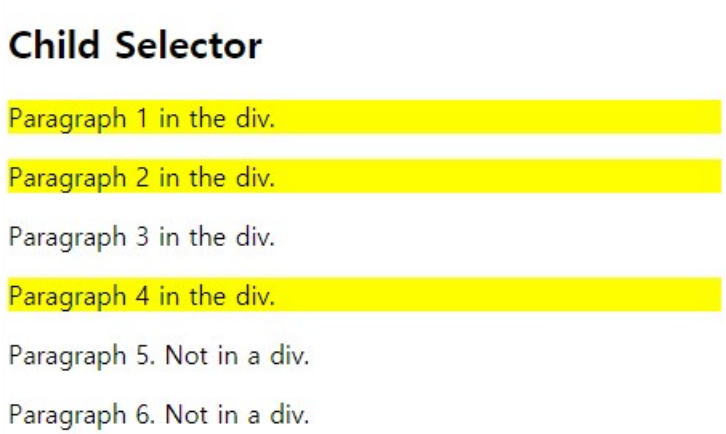



```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  div p {
6          background-color:
7          yellow; }
8  </style>
9  </head>
10 <body>
11 <div>
12 <p>Paragraph 1 in the div.</p>
13 <p>Paragraph 2 in the div. </p>
14 <section><p>Paragraph 3 in the
15 div. </p>
16 </section>
17 </div>
18 <p>Paragraph 4. Not in a div.</p>
19 <p>Paragraph 5. Not in a div.</p>
20 </body>
21 </html>
22
23
```

Child Selector

- The child selector selects all elements that are the children of a specified element.

Child Selector Example

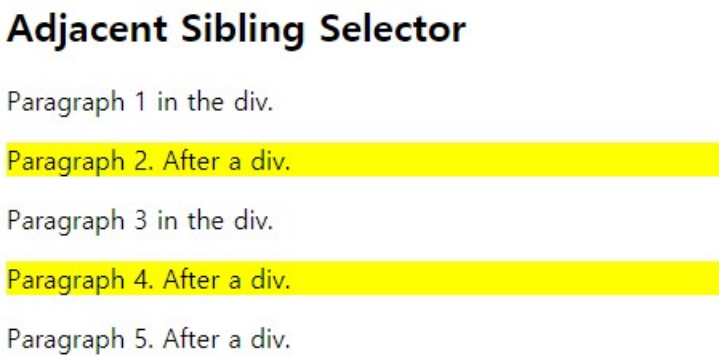



```
<!DOCTYPE html>
<html>
<head>
<style>
div > p {
        background-color:
        yellow; }
</style>
</head>
<body>
<h2>Child Selector</h2>
<div>
<p>Paragraph 1 in the div.</p>
<p>Paragraph 2 in the div.</p>
<section><p>Paragraph 3 in the
div.</p> </section>
<!-- not Child but Descendant -->
<p>Paragraph 4 in the div.</p>
</div>
<p>Paragraph 5. Not in a div.</p>
<p>Paragraph 6. Not in a div.</p>
</body>
</html>
```

Adjacent Sibling Selector

- **The adjacent sibling selector selects all elements that are the adjacent siblings of a specified element.**
  - Sibling elements must have the same parent element, and "adjacent" means "immediately following".

Adjacent Sibling Selector Example

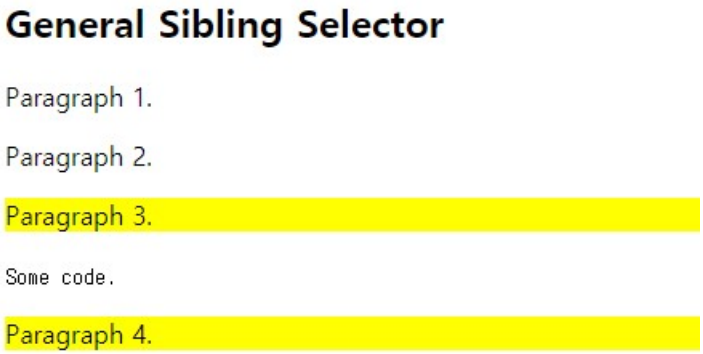



```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  div + p {
6        background-color: yellow; }
7  </style>
8  </head>
9  <body>
10 <h2>Adjacent Sibling Selector
11 </h2>
12 <div>
13 <p>Paragraph 1 in the div.</p>
14 </div>
15 <p>Paragraph 2. After a div.</p>
16 <div>
17 <p>Paragraph 3 in the div.</p>
18 </div>
19 <p>Paragraph 4. After a div.</p>
20 <p>Paragraph 5. After a div.</p>
21 </body>
22 </html>
23
```

General Sibling Selector

- The general sibling selector selects all elements that are siblings of a specified element.

General Sibling Selector Example




```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  div ~ p {
6          background-color:
7          yellow; }
8  </style>
9  </head>
10 <body>
11 <h2>General Sibling Selector</h2>
12 <p>Paragraph 1.</p>
13 <div>
14 <p>Paragraph 2.</p>
15 </div>
16 <p>Paragraph 3.</p>
17 <code>Some code.</code>
18 <p>Paragraph 4.</p>
19 </body>
20 </html>
21
22
23
```

CSS PSEUDO-CLASSES

What are Pseudo-classes?

❖ **A pseudo-class is used to define a special state of an element.**

❖ **For example, it can be used to:**

- Style an element when a user mouse over it
- Style visited and unvisited links differently
- Style an element when it gets focus

❖ **The syntax of pseudo-classes:**

```
selector:pseudo-class {
  property:value;
}
```

What are Pseudo-classes?

❖ All CSS Pseudo Classes

Selector	Example	Example description
:active	a:active	Selects the active link
:checked	input:checked	Selects every checked <input> element
:disabled	input:disabled	Selects every disabled <input> element
:empty	p:empty	Selects every <p> element that has no children
:enabled	input:enabled	Selects every enabled <input> element
:first-child	p:first-child	Selects every <p> elements that is the first child of its parent
:first-of-type	p:first-of-type	Selects every <p> element that is the first <p> element of its parent
:focus	input:focus	Selects the <input> element that has focus
:hover	a:hover	Selects links on mouse over
:in-range	input:in-range	Selects <input> elements with a value within a specified range

What are Pseudo-classes?

❖ All CSS Pseudo Classes

:last-child	p:last-child	Selects every <p> elements that is the last child of its parent
:last-of-type	p:last-of-type	Selects every <p> element that is the last <p> element of its parent
:link	a:link	Selects all unvisited links
:not(selector)	:not(p)	Selects every element that is not a <p> element
:nth-child(n)	p:nth-child(2)	Selects every <p> element that is the second child of its parent
:nth-last-child(n)	p:nth-last-child(2)	Selects every <p> element that is the second child of its parent, counting from the last child
:nth-last-of-type(n)	p:nth-last-of-type(2)	Selects every <p> element that is the second <p> element of its parent, counting from the las child
:nth-of-type(n)	p:nth-of-type(2)	Selects every <p> element that is the second <p> element of its parent
:only-of-type	p:only-of-type	Selects every <p> element that is the only <p> element of its parent

What are Pseudo-classes?

❖ All CSS Pseudo Classes

:only-child	p:only-child	Selects every <p> element that is the only child of its parent
:optional	input:optional	Selects <input> elements with no "required" attribute
:out-of-range	input:out-of-range	Selects <input> elements with a value outside a specified range
:read-only	input:read-only	Selects <input> elements with a "readonly" attribute specified
:read-write	input:read-write	Selects <input> elements with no "readonly" attribute
:required	input:required	Selects <input> elements with a "required" attribute specified
:root	root	Selects the document's root element
:target	#news:target	Selects the current active #news element (clicked on a URL containing that anchor name)
:valid	input:valid	Selects all <input> elements with a valid value
:visited	a:visited	Selects all visited links

Anchor Pseudo-classes

❖ Links can be displayed in different ways:

Anchor Pseudo-classes Example


```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  /* unvisited link */
6  a:link { color: red; }
7  /* visited link */
8  a:visited { color: green; }
9  /* mouse over link */
10 a:hover { color: hotpink; }
11 /* selected link */
12 a:active { color: blue; }
13 </style>
14 </head>
15 <body>
16 skip
17 </body>
18 </html>
19
20
21
22
23
```

Hover Pseudo-class

- The hover selector is used to select elements when you mouse over them.

Hover Example

Mouse over the div element below to change its background color:


```
<!DOCTYPE html>
<html>
<head>
<style>
div { background-color: green;
  color: white;
  padding: 25px;
  text-align: center;
}
div:hover {background-color: blue;
}
</style>
</head>
<body>
<p>Mouse over the div element
below to change its background
color:</p>

<div>Mouse Over Me</div>
</body>
</html>
```

Hover Example

Mouse over the div element below to change its background color:

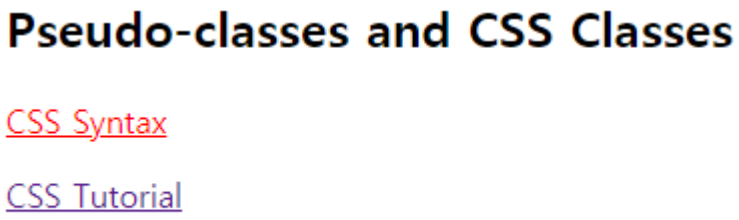

```
<!DOCTYPE html>
<html>
<head>
<style>
div { background-color: green;
  color: white;
  padding: 25px;
  text-align: center;
}
div:hover {background-color: blue;
}
</style>
</head>
<body>
<p>Mouse over the div element
below to change its background
color:</p>

<div>Mouse Over Me</div>
</body>
</html>
```

Pseudo-classes and CSS Classes

❖ Pseudo-classes can be combined with CSS classes:

- When you hover over the link in the example, it will change color:

Pseudo-classes and CSS Classes Example




```
<!DOCTYPE html>
<html>
<head>
<style>
a.highlight:hover {
        color: red; }
</style>
</head>
<body>
<h2>Pseudo-classes and CSS
Classes</h2>
<p>
<a class="highlight"
   href="css_syntax.asp">
   CSS Syntax</a>
</p>
<p><a href="default.asp">
   CSS Tutorial</a></p>
</body>
</html>
```

CSS - The :first-child Pseudo-class

- The :first-child pseudo-class matches a specified element that is the first child of another element.

CSS - The :first-child Pseudo-class Example

❖ Match the first <p> element

- In the following example, the selector matches any <p> element that is the first child of any element:

This is some text.

This is some text.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  p:first-child { color: blue; }
6  </style>
7  </head>
8  <body>
9  <p>This is some text. </p>
10 <p>This is some text. </p>
11 </body>
12 </html>
13
14
15
16
17
18
19
20
21
22
23
```

CSS - The :first-child Pseudo-class Example

- **Match the first <i> element in all <p> elements**
  - In the following example, the selector matches the first <i> element in all <p> elements:

I am a *strong* person. I am a *strong* person.

I am a *strong* person. I am a *strong* person.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  p i:first-child { color: blue; }
6  </style>
7  </head>
8  <body>
9  <p>I am a <i>strong</i> person. I
10 am a <i>strong</i> person.</p>
11 <p>I am a <i>strong</i> person. I
12 am a <i>strong</i> person.</p>
13 </body>
14 </html>
15
16
17
18
19
20
21
22
23
```

CSS - The :first-child Pseudo-class Example

- **Match all <i> elements in all first child <p> elements**
  - In the following example, the selector matches all <i> elements in <p> elements that are the first child of another element:

I am a *strong* person. I am a *strong* person.

I am a *strong* person. I am a *strong* person.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  p:first-child i { color: blue; }
6  </style>
7  </head>
8  <body>
9  <p>I am a <i>strong</i> person. I
10 am a <i>strong</i> person.</p>
11 <p>I am a <i>strong</i> person. I
12 am a <i>strong</i> person.</p>
13 </body>
14 </html>
15
16
17
18
19
20
21
22
23
```

CSS PSEUDO-ELEMENTS

What are Pseudo-elements?

❖ **A CSS pseudo-element is used to style specified parts of an element.**

❖ **For example, it can be used to:**

- Style the first letter, or line, of an element
- Insert content before, or after, the content of an element

❖ **The syntax of pseudo-elements:**

```
selector::pseudo-element {
  property: value;
}
```

What are Pseudo-elements?

❖ All CSS Pseudo Elements

Selector	Example	Example description
::after	p::after	Insert something after the content of each <p> element
::before	p::before	Insert something before the content of each <p> element
::first-letter	p::first-letter	Selects the first letter of each <p> element
::first-line	p::first-line	Selects the first line of each <p> element
::marker	::marker	Selects the markers of list items
::selection	p::selection	Selects the portion of an element that is selected by a user

The ::first-letter Pseudo-element

❖ **The ::first-letter pseudo-element is used to add a special style to the first letter of a text.**

❖ **The following properties apply to the ::first-letter pseudo- element:**

- font properties
- color properties
- background properties
- margin properties
- padding properties
- border properties
- text-decoration
- vertical-align (only if "float" is "none")
- text-transform
- line-height
- float
- clear

The ::first-letter Pseudo-element Example

You can use the ::first-letter pseudo-element to add a special effect to the first character of a text!

```
<!DOCTYPE html>
<html>
<head>
<style>
p::first-letter {
  color: red;
  font-size: xx-large;
}
</style>
</head>
<body>
~skip~
</body>
</html>
```

The ::first-line Pseudo-element

❖ **The ::first-line pseudo-element is used to add a special style to the first line of a text.**

❖ **The following properties apply to the ::first-line pseudo-element:**

- font properties
- color properties
- background properties
- word-spacing
- letter-spacing
- text-decoration
- vertical-align
- text-transform
- line-height
- clear

The ::first-line Pseudo-element Example

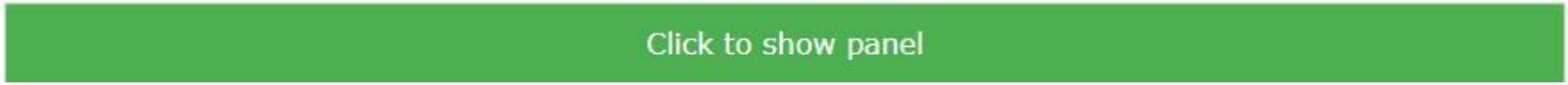
You can use the ::first-line pseudo-element to add a special effect to the first line of a text. Some more text. And even more, and more, and more, and more, and more, and more, and more, and more, and more, and more, and more, and more.

```
<!DOCTYPE html>
<html>
<head>
<style>
p::first-line {
  color: red;
  font-variant: small-caps;
}
</style>
</head>
<body>
~skip~
</body>
</html>
```

CSS DISPLAY

The display property

- The display property is the most important CSS property for controlling layout.
- The display property specifies if/how an element is displayed.
- Every HTML element has a default display value depending on what type of element it is. The default display value for most elements is block or inline.



The display property

- The display property is the most important CSS property for controlling layout.
- The display property specifies if/how an element is displayed.
- Every HTML element has a default display value depending on what type of element it is. The default display value for most elements is block or inline.

Block-level Elements / Inline Elements

❖ Block-level Elements

- A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can).
- The element is a block-level element.
- Examples of block-level elements:
  - , <h1> - <h6>, <p>, <form>, <header>, <footer>, <section>

❖ Inline Elements

- An inline element does not start on a new line and only takes up as much width as necessary.
- This is an inline element inside a paragraph.
- Examples of inline elements:
  - , <a>, <img>

Display: none;

- display: none; is commonly used with JavaScript to hide and show elements without deleting and recreating them. Take a look at our last example on this page if you want to know how this can be achieved.
- The <script> element uses display: none; as default.

Override The Default Display Value

- As mentioned, every element has a default display value. However, you can override this.
- Changing an inline element to a block element, or vice versa, can be useful for making the page look a specific way, and still follow the web standards.
- Note: Setting the display property of an element only changes how the element is displayed, NOT what kind of element it is. So, an inline element with display: block; is not allowed to have other block elements inside it.

Override The Default Display Value Example

A B C

A display property with a value of "block" results in a line break between the two elements.

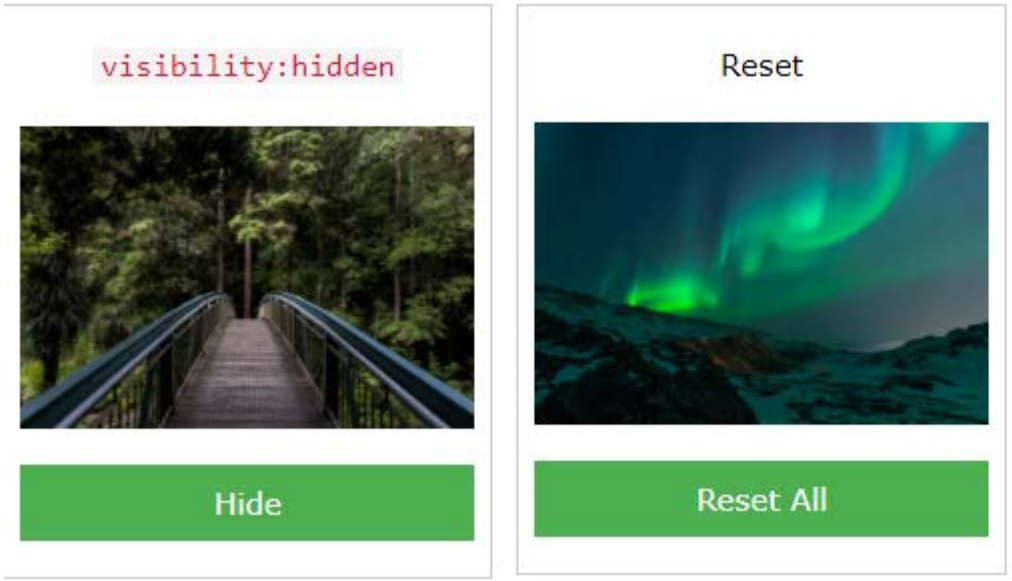
```
1  ~skip~
2  <head>
3  <style>
4  li { display: inline; }
5  span { display: block;}
6  </style>
7  </head>
8  <body>
9  ~Skip~
10 <ul>
11  <li><a href="/html/default.asp"
12        target="_blank">A</a></li>
13  <li><a href="/css/default.asp"
14        target="_blank">B</a> </li>
15  <li><a href="/js/default.asp"
16        target="_blank">C</a></li>
17 </ul>
18 <span>A display property with a
19 value of "block" results in</span>
20 <span>a line break between the two
21 elements. </span>
22 </body>
23 </html>
```

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

❖ Hiding an element can be done by setting the display property to none. The element will be hidden, and the page will be displayed as if the element is not there:

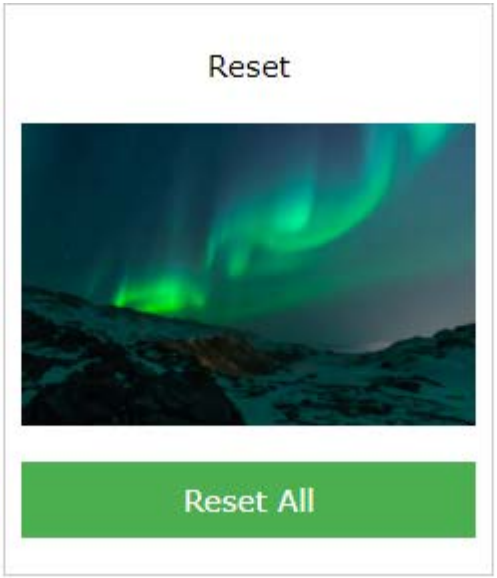
Hide an Element - display:none or visibility:hidden?

❖ Hiding an element can be done by setting the display property to none. The element will be hidden, and the page will be displayed as if the element is not there:



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

- visibility:hidden; also hides an element.
- However, the element will still take up the same space as before. The element will be hidden, but still affect the layout:



Hide an Element Example


```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  h1.hidden { visibility: hidden; }
6  </style>
7  </head>
8  <body>
9  <h1>This is a visible
10 heading</h1>
11 <h1 class="hidden">This is a
12 hidden heading</h1>
13 <p>Notice that the hidden heading
14 still takes up space.</p>
15 </body>
16 </html>
17
18
19
20
21
22
23
```

Simple Tooltip Hover

❖ Hover over a element to show a <p> element (like a tooltip):

Simple Tooltip Hover Example

Hover over me to show the p element

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
        display: none;
        background-color: yellow;
        padding: 20px; }
div:hover p {
        display: block; }
</style>
</head>
<body>
<div>
Hover over me to show the p
element
<p>Tada! Here I am!</p>
</div>
</body>
</html>
```

Simple Tooltip Hover Example

Hover over me to show the p element

Tada! Here I am!

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
        display: none;
        background-color: yellow;
        padding: 20px; }
div:hover p {
        display: block; }
</style>
</head>
<body>
<div>
Hover over me to show the p
element
<p>Tada! Here I am!</p>
</div>
</body>
</html>
```

CSS FLOAT AND CLEAR

The float Property

- **The float property is used for positioning and formatting content e.g. let an image float left to the text in a container**

- **The float property can have one of the following values:**
  - left- The element floats to the left of its container
  - right- The element floats to the right of its container
  - none - The element does not float (will be displayed just where it occurs in the text). This is default
  - inherit - The element inherits the float value of its parent
- **In its simplest use, the float property can be used to wrap text around images.**

Float Example

Float

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor.

```
<!DOCTYPE html>
<html>
<head>
<style>
img{
        float: right;
}
</style>
</head>
<body>
SKIP
</body>
</html>
```

Float Example

Float


Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  img{
6          float: left;
7  }
8  </style>
9  </head>
10 <body>
11 SKIP
12 </body>
13 </html>
14
15
16
17
18
19
20
21
22
23
```

Float Example

Float


Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor.

```
<!DOCTYPE html>
<html>
<head>
<style>
img{
        float: none;
}
</style>
</head>
<body>
SKIP
</body>
</html>
```

The clear Property

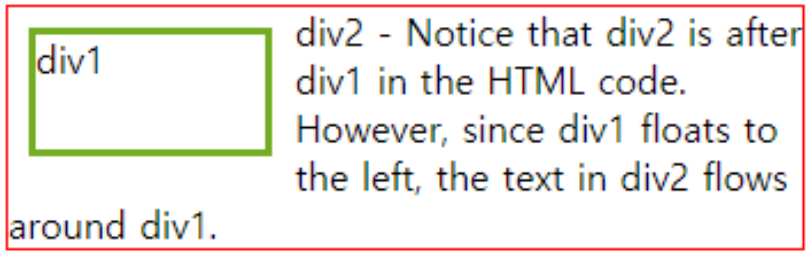
- **The clear property specifies what elements can float beside the cleared element and on which side.**

- **The clear property can have one of the following values:**
  - none - Allows floating elements on both sides. This is default
  - left - No floating elements allowed on the left side
  - right- No floating elements allowed on the right side
  - both - No floating elements allowed on either the left or the right side
  - inherit - The element inherits the clear value of its parent

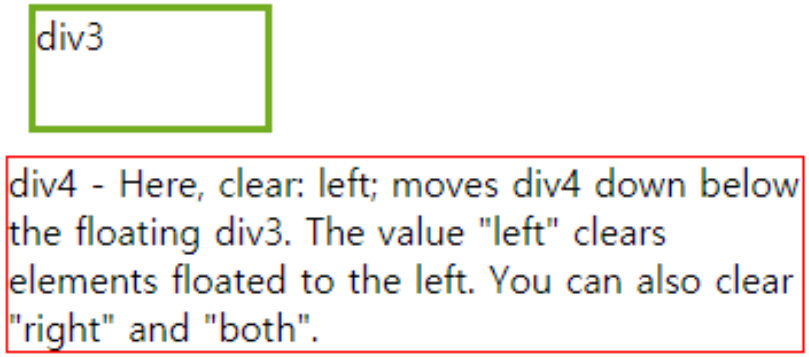
- **The most common way to use the clear property is after you have used a float property on an element.**

Clear Example

Without clear



With clear




```
1  <!DOCTYPE html>
2  ~skip~
3  <style>
4  .div1{   float: left;
5           width: 100px;height:50px;
6           margin: 10px;
7           border: 3px solid
8           #73AD21;}
9  .div2{   border: 1px solid red;}
10 .div3{    ~skip~ /*same as div1*/}
11 .div4{   border: 1px solid red;
12          clear: left;   }
13 ~skip~
14 <body>
15 ~skip~
16 <div class="div1"> div1 </div>
17 <div class="div2"> div2 ~skip~
18 </div>
19 <div class="div3"> div3 </div>
20 <div class="div4"> div4~skip~
21 </div>
22 ~skip~
23
```

CSS ICONS

How To Add Icons

- The simplest way to add an icon to your HTML page, is with an icon library, such as Font Awesome.

- Add the name of the specified icon class to any inline HTML element (like <i> or).

- All the icons in the icon libraries below, are scalable vectors that can be customized with CSS (size, color, shadow, etc.)

Icons

- To use the Font Awesome icons, go to www.fontawesome.com , sign in, and get a code to add in the <head> section of your HTML page:
  - <script src="https://kit.fontawesome.com/yourcode.js"></script>

- To use the Bootstrap glyphicons, add the following line inside the <head> section of your HTML page:
  - <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

- To use the Google icons, add the following line inside the <head> section of your HTML page:
  - <link rel="stylesheet" href="https://fonts.googleapis.com/icon?family=Material+Icons">

Icon Example

Some Google icons:

Styled Google icons (size and color):

❖ **Other Icons**

- https://fonts.google.com/icons

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <link rel="stylesheet"
5  href="https://fonts.googleapis.co
6  m/icon?family=Material+Icons">
7  </head>
8  <body>
9  <i class="material-
10 icons">cloud</i>
11 <i class="material-
12 icons">favorite</i>
13 <i class="material-
14 icons">attachment</i>
15 <i class="material-
16 icons">computer</i>
17 <i class="material-
18 icons">traffic</i>
19 ~skip~
20 <i class="material-icons"
21 style="font-size:24px;">cloud</i>
22 </body>
23 </html>
```

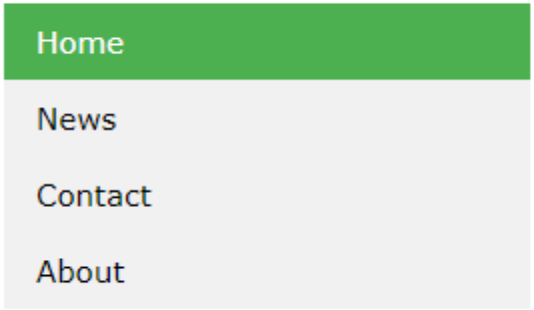
CSS NAVIGATION BAR

How To Add Navigation Bar

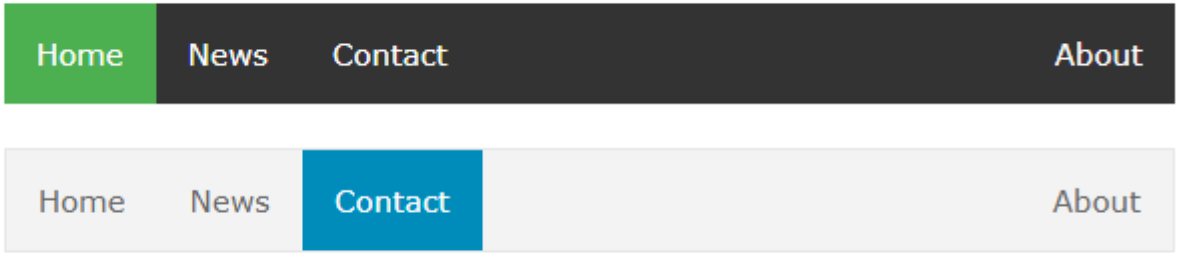
- With CSS you can transform boring HTML menus into good-looking navigation bars.

- Navigation Bar = List of Links

Vertical



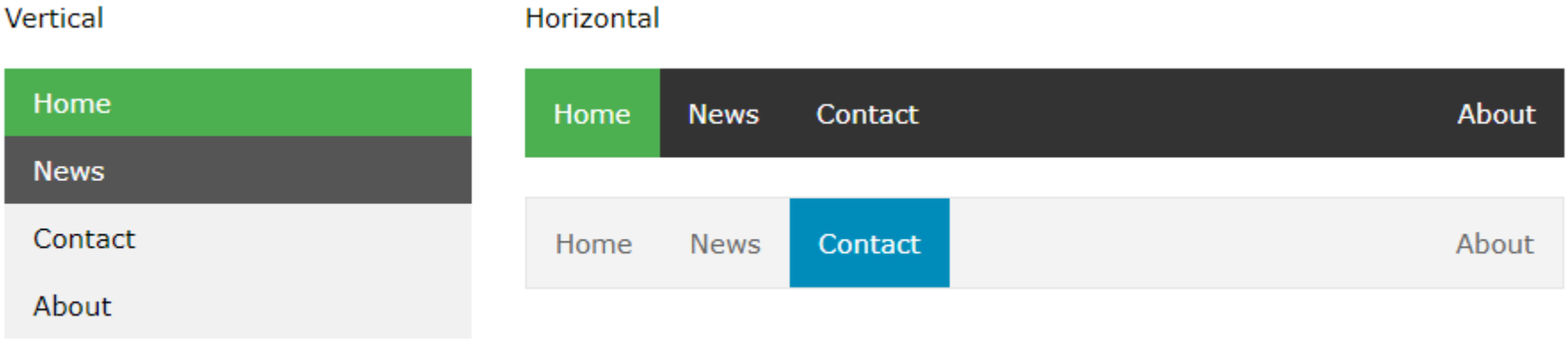
Horizontal



How To Add Navigation Bar

- With CSS you can transform boring HTML menus into good-looking navigation bars.

- Navigation Bar = List of Links



CSS 2D/3D TRANSFORMATION

What are 2d transformation

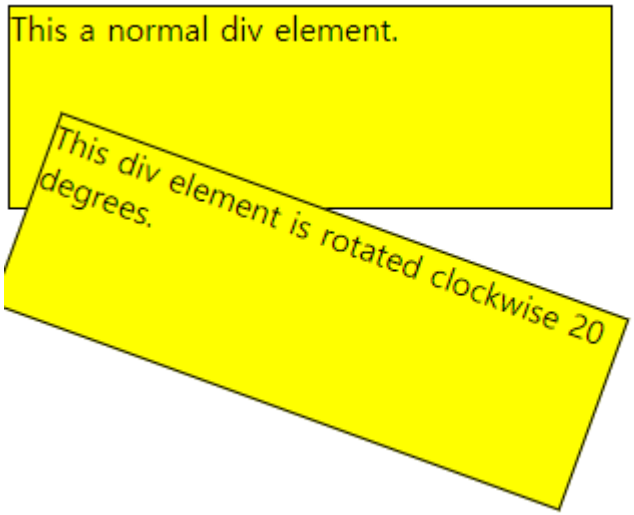
- CSS 2D transforms allow you to move, rotate, scale, and skew elements.

- The numbers in the table specify the first browser version that fully supports the property.

Property					
transform	36.0	10.0	16.0	9.0	23.0

- With the CSS transform property you can use the following 2D transformation methods.
  - `translate()`
  - `rotate()`
  - `scaleX()`
  - `scaleY()`
  - `scale()`
  - `skewX()`
  - `skewY()`
  - `skew()`
  - `matrix()`

What are 2d transformation Example




```
<!DOCTYPE html>
<html>
<head>
<style>
div {
     width: 300px; height: 100px;
     background-color: yellow;
     border: 1px solid black;}
div#myDiv {
     -ms-transform: rotate(20deg);
     transform: rotate(20deg);
< /style>
</head>
<body>
~skip~
</body>
</html>
```

What are 3d transformation

- CSS 3D transforms allow you to move, rotate, scale, and skew elements.

- The numbers in the table specify the first browser version that fully supports the property.

Property					
transform	36.0	10.0	16.0	9.0	23.0

- With the CSS transform property you can use the following 3D transformation methods.

  - `rotateX()`
  - `rotateY()`
  - `rotateZ()`

What are 3d transformation

- CSS 3D transforms allow you to move, rotate, scale, and skew elements.

- The numbers in the table specify the first browser version that fully supports the property.

Property					
transform	36.0	10.0	16.0	9.0	23.0

- With the CSS transform property you can use the following 3D transformation methods.

  - `rotateX()`
  - `rotateY()`
  - `rotateZ()`

Example in the Textbook

❖ CSS pseudo-class 사용

- link
- visited
- hover

❖ CSS Combinator 사용

- descendant / child

❖ 추가로 사용해야 할 CSS 속성

- width, height
- float
- clear
- display
- text-decoration

Example in the Textbook

❖ CSS pseudo-class 사용

- link
- visited
- hover

❖ CSS Combinator 사용

- descendant / child

❖ 추가로 사용해야 할 CSS 속성

- width, height
- float
- clear
- display
- text-decoration

Assignments

❖ CSS pseudo-class 사용

- link
- visited
- hover

❖ CSS Combinator 사용

- descendant / child

❖ Icon 사용

- Google Icons

❖ 추가로 사용해야 할 CSS 속성

- width, height
- float
- clear
- display
- text-decoration

Assignments

❖ CSS pseudo-class 사용

- link
- visited
- hover

❖ CSS Combinator 사용

- descendant / child

❖ Icon 사용

- Google Icons

❖ 추가로 사용해야 할 CSS 속성

- width, height
- float
- clear
- display
- text-decoration