

## 01. HTML5

HTML(Hyper Text Markup Language)란 Web Page를 만들기위해 사용되는 프로그래밍 언어이다. 그렇다면 Web이란 일종의 문서 전송을 위한 서비스로 HTML로 만든 문서들이 얹혀 있는 것을 본 따서 Web 이라고 이야기 한다. 구조를 보자면 HTML로 -> Web 문서가 만들어지고 -> Web 서비스를 통해서 사용자들에게 문서가 전달된다.

### ☑ WWW

Web 이라 하면 WWW(World Wide Web)를 뜻하는데 인터넷 상에서 이루어지는 서비스이다. 구체적으로 Web 서비스라고 하면 HTTP를 통해서 HTML로 만든 문서를 전송할 수 있도록 구축된 서비스를 말한다. Web Server, Web Document, Web Page 등등 Web 이라는 말은 많이 들어 봤을 것이다.

인터넷 상에는 Web 서비스 뿐만 아니라 ftp, smtp 등 다양한 전송 프로토콜을 사용하는 서비스들이 존재한다. Web이란 인터넷 상에서 이루어지는 이런 서비스들 중 하나인 것이다.

서비스들마다 데이터 전송을 위한 고유한 약속을 정해 놓고 데이터를 전송한다. 이것이 바로 HTTP나 FTP라는 용어로 사용된다. Web 서비스에서는 HTTP(Hyper Text Transfer protocol)라는 전송 규약을 정해 놓고 문서를 전달한다. Web Browser에서 http로 시작되는 주소 HTTP://www.~.~.~ 는 HTTP 전송 규약을 통해서 문서를 전달한다는 뜻이 된다. 왜 Web Page 주소가 HTTP로 시작되는지 이해할 수 있다. 간혹 ftp로 시작되는 주소도 있ek. ftp를 사용하는 서비스이다. 보통 ftp는 Web 브라우저보다는 ftp 전용 프로그램을 사용한다.

### ☑ HTML

HTTP를 통해서 전달되는 문서는 HTML이라는 프로그래밍 언어로 이루어져 있다. 사실 애초에 HTML은 문서를 위해 포맷을 잡아주는 형태라서 프로그래머 사이에서도 C언어와 같은 프로그래밍 언어와는 좀 다른 시각을 둔 것이 사실이다.

그러나 최근에는 복잡해져가는 Web 환경으로 인해 Web Page를 개발하려면 HTML 뿐만 아니라 CSS, JavaScript나 그 밖에 다른 프로그래밍 언어에 대한 지식도 수반되는게 일반적이므로 Web 개발 역시 프로그래머가 선택할 수 있는 하나의 확고한 영역이 되었다.

HTML 하나만 놓고 보면 그렇게 크게 어렵지 않다. 거기에다 결과를 바로 바로 Web Browser로 확인할 수 있으므로 학습하기에도 재미있다.

### ☑ HTML 역사

현재 HTML5는 탄생 배경에는 흥미로운 역사가 있다. 20여년전, 1990년대만 하더라도 인터넷을 사용하는 경우가 그리 흔하지는 않았다. 통신회선의 데이터 전송속도도 느렸고 인터넷 페이지도 요새에 비하면 그리 많지 않았다. 인터넷 보다는 터미널을 이용한 하이텔, 나우누리, 천리안 같은 통신을 주로 했었

다.

이 당시만 하더라도 인터넷 브라우저 중에 NetScape 것이 있었다. 물론 윈도우 익스플로러도 있었지만 그 당시만 하더라도 네스케이프의 성능과 인지도가 높았다.



그러나 마이크로 소프트사가 2000년도 들어와서 MS의 Internet Explorer가 거의 독점하다시피 한다. 그 배경에는 ActiveX라는 것이 있었다. 예전에는 관공서나 인터넷 뱅킹, 공인인증서 등을 사용하려면 ActiveX라는 것을 설치했다. 인터넷 익스플로러에만 있는 것이다. 이것은 네스케이프나 타사 브라우저와 호환되지 않는 형태를 띄므로 점점 사용자는 Internet Explorer 만을 사용할 수 밖에 없게 되었다. 물론 이를 해결하고자 다른 브라우저 개발사들이 표준을 제안하지만 W3C에서 거절한다. W3C(World Wide Web Consortium)은 국제 Web 표준화 기구이다.

이러한 문제에도 불구하고 최근에는 이 표준이 다시 등장했는데 이것이 바로 HTML5 이다. 예전에 MS를 제외한 Web 브라우저 개발사들이 제안한 표준이다. 이것이 2014년의 일이다.

이런 Web 표준을 통해서 다양한 브라우저의 선택적인 사용이 가능해 진다. MS의 Internet Explorer의 독점의 시대는 끝난 것이다.

## ☑ HTML5 + CSS3

앞으로 HTML5를 배우면서 CSS3에 대한 내용도 살펴보게 될 것이다. 이 둘은 실과 바늘처럼 같이 생각하는 것이 좋다. CSS3(Cascading Style Sheets)는 일종의 장식을 어떻게 할지 기술해주는 언어이다. HTML5로 집을 지으면 CSS3로 인테리어를 하는 것이다.

## ☑ 개발을 위한 환경 설정

프로그래밍이 처음이시라면 개발 환경을 설정하는 것에 대해서 익숙하지 않을 것이다. 같은 종류의 개발을 하더라도 개발환경은 개발자들마다 회사마다 다를 수가 있다. Web 개발의 경우는 텍스트 에디터와 Web Browser를 사용할 수 있는 환경이면 가능하다. 텍스트 에디터는 윈도우 기본 메모장도 가능한데 이 강의에서는 Visual Studio Code를 사용한다.

## ☑ HTML5 구조

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
    <title>JS Bin</title>
  </head>
  <body>
  </body>
</html>
```

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
    <title>JS Bin</title>
  </head>
  <body>
    <h1>Heading1</h1>
    <h2>Heading2</h2>
    <button>Click Me!</button>
  </body>
</html>
```

## ☑ W3C

<https://www.w3.org/>

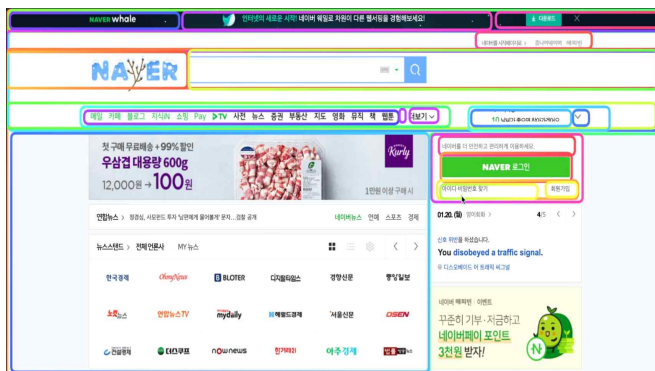
<https://validator.w3.org/>

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element>

## ☑ Document and website structure

[https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction\\_to\\_HTML/Document\\_and\\_website\\_structure](https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Document_and_website_structure)

## ☑ Web Site의 Box Model Sample





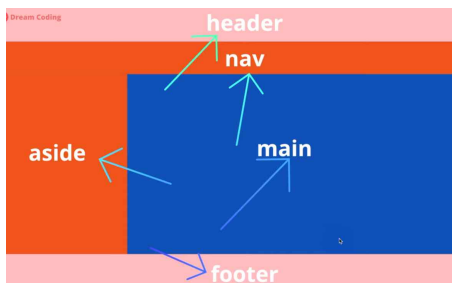
## ☑ tag들의 종류

☞ 모든 tag들은 box가 되거나 item이 되는 tag들이다. 따라서 사용자가 볼수 있는 item들이 있는가 하면 이 item들을 잘 정리(sectioning)할 수 있도록 도와주는 보이지 않는 box가 있다.

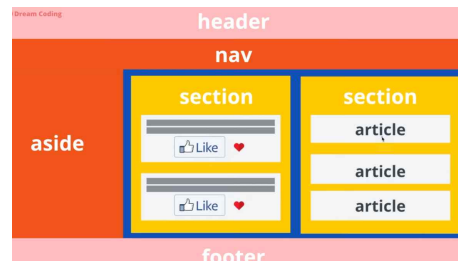
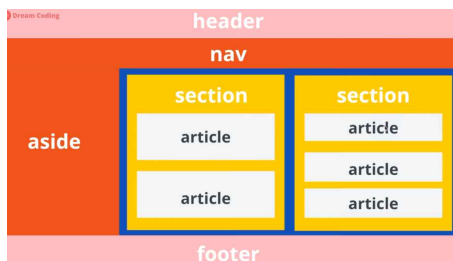


☞ 보이지 않는 box들은 CSS를 통하여 styling이 가능하다. 그러나 기본적으로 html에서는 사용자에게 보여지지 않는다.

 BOX	 ITEM
header	a
footer	video
nav	button
aside	audio
main	input
	map
	label
	canvas
	table
	img



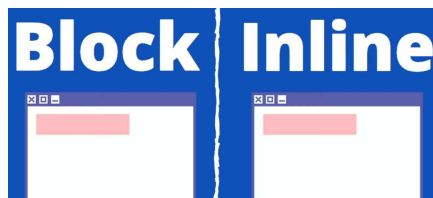
☞ section 안에 article이 있고, article 안에는 text가 2줄이 있고, Like와 hart라는 icon이 있다. 이렇게 반복된 것을 재사용 가능하게 묶어준 것을 article 이라고 한다. 이와 같이 Web Site를 크게 나누어 볼 수가 있다.



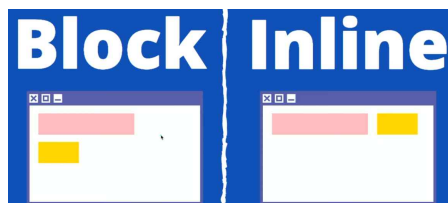
☞ Box는 header, footer, nav, aside, main 식으로 session이 나누어지는 것을 가리키는 tag들을 말한다. 또한 section 안에서 article이 여러 개가 들어 갈 수가 있고, article은 조금 반복되는

재사용이 가능한 것을 묶어놓은 것이다.

- ☞ div는 아무 곳에서나 사용할 수 있는 tag 이다. 묶어서 styling을 할 필요가 있을 때, 즉 text나 button을 묶어서 border를 주거나 background-color를 주거나 묶고 싶을 때 div를 이용해서 묶을 수가 있다.
- ☞ item은 a, button, input, lable, img, video, audio, map, canvas, table 등 사용자들에게 보여지는 tag 들이다.
- ☞ tag들은 Box가 되는 것과 사용자에게 보여지는 item들이 되는 tag 가 있다. 그리고 사용자에게 보여지는 item 들도 2가지로 나누어서 생각할 수가 있으면 추후 CSS를 사용할 때 편리하게 사용할 수가 있다. 바로 Block과 Inline으로 나누어 질 수가 있다.
- ☞ 여기 Web Site에서 pink color item이 이미 Browser에 들어가 있는 것을 볼 수가 있다.



- ☞ 여기에 yellow item을 배치하고자 하면, 아무런 CSS 없이 pink item 옆에 yellow를 정리하게 되면 똑 같은 사이즈의 yellow color tag를 바로 pink color 옆에 작성하였지만 yellow element가 block level 인지 inline level 인지에 따라서 배치되는 구조가 조금 달라지는 것을 볼 수가 있다.

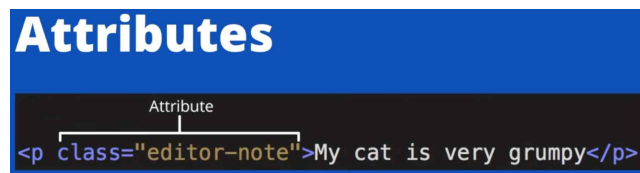


- ☞ block level인 경우 충분한 공간이 있음에도 불구하고 yellow tag는 rm 다음 라인으로 들어가는 것을 볼 수가 있다. 즉, block level의 element는 한 줄에 하나가 차지하는 것을 말한다.
- ☞ inline level에서는 충분한 공간이 있었기 때문에 yellow color는 바로 pink color 옆에 배치가 된다. 즉, inline level tag는 공간이 충분하면 그 옆의 다른 tag 옆에 배치가 가능하다.

- ☞ tag는 <> 를 이용하여 시작하고 </>을 이용해서 끝난다. tag 안에 있는 것은 content 이다. 이 <> tag에서 시작하여 끝나는 </> tag 하나를 element라고 한다. 이것을 node 라고 이야기 하기도 한다.
- ☞ 향후 JavaScript에서 DOM Tree를 학습할 때 이 element가 어떻게 DOM Tree에 입력되고 구성이 이루어지는지에 대하여 학습하도록 한다.



- ☞ tag 안에는 attribute 라는 속성이 있다. attribute에 class는 CSS에서는 이 class를 통하여 원하는 class들을 어떠한 식으로 꾸밀지를 정하게 된다.
- ☞ 같은 page에 있는 다양한 button이라도 이 button이 어떤 class를 가지느냐에 따라서 borderless 인 button을 만들 것인지 invisible button을 만들 것인지 그리고 click이 가능하지 않게 할 것인지를 이 class를 통하여 정의할 수가 있다.



지금부터 사용자에게 보여지는 box와 사용자에게 보이지 않는 item에 대하여 실습하도록 한다.

<header></header>, <footer></footer>, <section></section>, <div></div>, <span></span> 등 Box에 해당하는 tag들은 안에 content가 없는 이상 절대 Browser에서 보이지지가 않는다.

엄밀히 말하면 <h1></h1> 도 content가 없으면 사용자에게 보이지 않아서 box 라고 정의할 수있지만, <h1></h1> tag는 안의 content를 decoration 해주기 때문에 box가 아니라고 볼 수가 있다.

<button></button>은 안에 내용이 없어도 자동적으로 Browser에서 rendering하여 사용자에게 보여준다.

<a> tag는 속성값에는 href="https://www.googl.com" 을 통하여 link 하고 싶은 주소를 입력하면 된다. 새로운 창에서 띄워보기 위해서는 target=\_blank 를 통하여 원하는 page를 open 할 수 있다.

<a href="https://www.google.com target=\_blank>Click</a>

\_self는 현재 창에서 열고(기본), \_blank는 일반적으로 새 탭이지만 사용자는 대신 새 창을 열도록 Browser를 구성할 수 있다. \_parent는 현재의 parent browsing context. parent가 없으면 \_self로 동작한다. \_top은 topmost browsing context(현재 context의 ancestor(조상)인 "highest" context). ancestor가 없으면 \_self로 동작한다.

다음으로 text와 관련된 tag들을 살펴보면서 block과 inline level이 어떻게 다른지 학습하도록 한다.

<p></p>은 paragraph의 약자이다. 문장을 정의할 때 사용하는 tag 이다. 만약 2개의 문장이 있으면, 문장 문장을 연결할 때 문단을 작성하게 된다. 만약 문장을 bold 체로 꾸밀 경우에는 다음과 같이 소요하면 된다.

<p> Semyung University. <b>School of SmartIT</b></p>

<b></b> tag는 inline level 이기 때문에 다음 라인으로 넘어가지가 않는다.

<p> Semyung University. <span>School of SmartIT</span></p>

만약 같은 문장을 <span></span>을 이용하여 꾸민 다음에 CSS를 통하여 이 <span> tag 안의 content를 원하는 color로 지정을 할 수가 있다. <span> tag를 이용하면 다음 라인으로 넘어가지 않고 똑같이 한 문장 안에 있게 된다. 이유는 <span> tag는 inline level 이기 때문이다.

<p> Semyung University. <div>School of SmartIT</div></p>

반대로 <span> tag 대신에 <div> tag를 사용하면 바로 다음 라인으로 넘어가게 된다. 이것은 바로 <div> tag는 block level 이기 때문이다.

따라서 inline과 block level을 어떤 tag를 사용하느냐에 따라서 자동으로 줄바꿈이 일어나는지 일어나지 않는지 달라지게 된다. 그리고 기본 적인 행동들은 CSS를 통하여 수정이 가능하다.

다음으로는 list에 대하여 학습하도록 한다.

list에는 order 가 있는 <ol> tag가 있고, order가 없는 <ul> tag가 있으며, 이 list 안에 들어가는 list item들이 있는 <li> tag가 있다.

```
ol>li*3 => tab
```

자동으로 <ol> tag들의 list item들은 Browser에서 1, 2, 3 으로 지정해서 화면에 나타난다. 속성값으로는 reversed, start, type 등이 있다. Roman numerals 를 표시해 보면 다음과 같다.

```
<ol type="i">
  <li>1</li>
  <li>1</li>
  <li>1</li>
</ol>
```

이제부터 tag를 배울 때는 MDN Site를 보면서 실습하도록 한다.

```
<ol type="i reversed">
```

이 경우에는 boolean 속성값에 해당하는 것이다. boolean 속성값은 값을 넣지 않아도 toggle 형식으로 사용할 수가 있다.

```
<ul type="i">
  <li>HTML5</li>
  <li>CSS3</li>
  <li>javascript</li>
</ul>
```

<ul> tag는 이렇게 숫자가 아니라 dot 형식으로 나타나는 것을 볼 수가 있다. 물론 CSS를 통해서 이러한 글머리표를 나타나지 않게 하거나 다른 방식으로 표시할 수가 있다.

이렇게 list item 들은 숫자나 dot의 형식으로 이용하지 않아도 item 들을 정렬할 때 가장 많이 사용하는 tag 이다.

다음에 학습할 tag는 <input> tag 이다. 이 tag는 Browser에서 많이 사용되는 tag 이다. 사용자에게 입력을 받을 수 있기 때문이다. 우리가 원하는 데이터를 사용자에게 요구해서 사용자에게 받게 된다.

<input> tag 만 사용되기 보다는 <table>을 같이 사용하여 사용자에게 정확하게 어떠한 정보를 원하는지를 명확하게 나타내주기 때문이다.



```
<label for="input_name">Name: </label>
<input id="input_name" type="text" >
```

<label>이나 <input>은 둘다 inline element라는 것을 알 수가 있다. 둘 다 한 줄에 표기되기 때문이다.

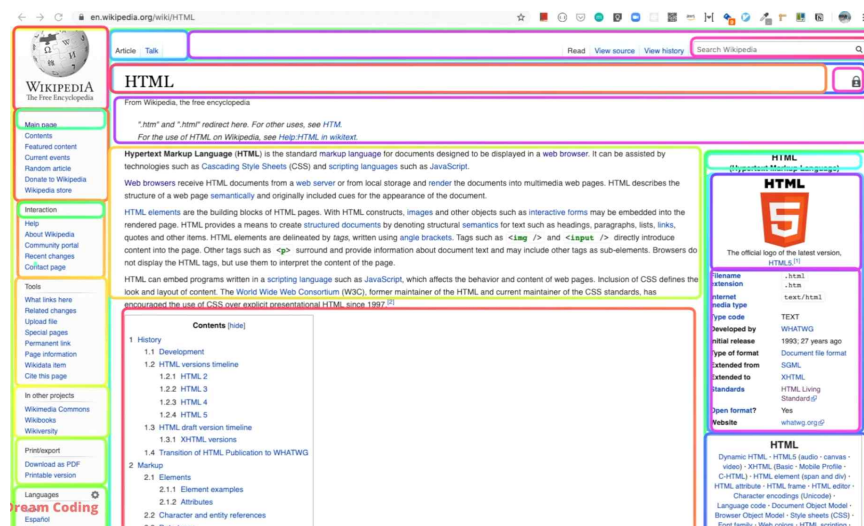
<input>은 한 page 안에 많은 input이 있을 수 있기 때문에 id를 통하여 고유한 식별자를 주게 된다. 만약 id="input\_name" 으로 입력하게 되면 input\_name을 위한 label이라고 label에서 설정할 수가 있다. <input> 속성의 type에는 많은 type들이 존재한다.

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input>

```
<input id="input_name" type="password">
```

type을 password로 지정하면 사용자에게 보이지 않는 text input을 받을 수 있다.

Wikipedia Web Site를 Box Model로 나누어 보도록 한다. 어떻게 분류하느냐에 따라 성능의 차이가 나타날 수 있겠지만 사실 정해진 정답은 없다.



이렇게 Box Model로 나누어 놓은 결과를 Wikipia에서 어떻게 구현하였는지 확인하기 위해서는 Chrome의 Delveopment Tool을 사용하면 된다. Ctrl+Shift+i 를 클릭하면 볼 수가 있는데, [Elements] tab에 Element inspector를 이용하여 mouse를 올려보면 Box Model이 어떤식으로 구현되어 있는지 확인을 할 수가 있다. 초록색은 padding 이고, 주황색은 margin을 나타낸다.

## 02. HTML tag와 attribute

### ☑ HTML : attribute와 property 의 차이

HTML에서 말하는 attributes와 properties의 차이점은 jQuery 1.6에 소개되는 .prop() 함수에 대해 기존 .attr() 과 무엇이 다른지 많은 질문이 제기되었다.

#### ❖ attribute는 무엇인가?

attributes는 HTML element(요소)의 추가적인 정보를 전달하고, 이름="값" 이렇게 쌍으로 온다. 예를 들어 <div class="my-class"></div> 를 보면 div 태그가 class 라는 값이 'my-class'인 attribute를 가지고 있다.

#### ❖ property는 무엇인가?

property는 attribute에 대한 HTML DOM tree 안에서의 표현이다. 그래서 위 예시에서 attribute는 값이 'my-class'이며 이름이 'className'인 property를 가진다.

Our DIV node

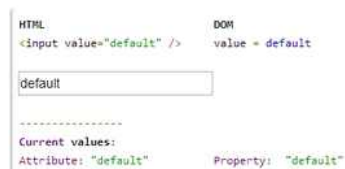
```
| - nodename = "DIV"
| - className = "my-class"
| - style
|   | - . . .
| - . . .
```

#### ❖ attribute 와 property의 차이는 무엇인가?

attributes는 HTML text 문서에 있는 것이고, properties는 HTML DOM tree에 있는 것이다. 이것은 attribute는 변하지 않고 초기 (default)값을 전달한다는 것은 의미한다. 반면에 HTML properties는 변할 수 있다.

예를 들어 사용자가 체크박스를 체크했거나 input 박스에 텍스트를 넣었거나 JavaScript로 값을 변경하면 property의 값은 변한다.

시각적으로 설명하면 :



사용자가 그의 이름 "Joe"를 input 박스에 입력했다고 가정하자. 그러면 attribute와 property 값이 아래와 같이 된다.



보시다시피, property의 값만 변경되었다. 왜냐하면 property는 DOM 안에 존재하고 동적이기 때문이다. 그러나 attribute는 HTML 문서 안에 존재하고 결코 변하지 않는다.

## ☑ HTML의 구성요소 tag, element(요소), attribute(속성), arguments

HTML을 공부하면서 아래와 같은 다소 생소한 용어가 몇개 등장한다. 단지 이 부분을 이렇게 부른다는 정도의 의미이지만, 설명을 하기 위해서는 이런 용어를 사용할 수 밖에 없다.

### ❖ elements

HTML에서 시작 tag와 종료 tag로 이루어진 모든 명령어들을 의미한다.

### ❖ tag

elements의 일부로 시작 tag와 종료 tag 두 종류가 있다. 시작 tag는 element를 시작하며, 종료 tag는 element를 끝내는 기능을 가지고 있다. 일부 tag 중에는 종료 tag가 없는 것도 있다.

element와 tag의 개념이 뚜렷이 구별되지 않고 혼용되는 경우가 많은데 HTML을 배우는다는 굳이 element 라는 말을 쓰지 않아도 되지만 CSS나 JavaScript를 배울 때는 element 라는 말이 아주 중요한 용어가 된다.

⟨p>하나의 문단내용이 여기에 들어 온다.</p>

위의 예제에서 element는 ⟨p> 하나의 문단내용이 여기에들어 온다.</p> 전체를 의미한다. 그러나 tag는 ⟨p>와 </p>만을 의미한다.

### ❖ attribute

element의 시작 tag 안에서 사용되는 것으로 좀 더 구체화된 명령어 체계를 의미한다.

⟨p align="center">태그에 속성을 부여하면 태그의 성격을 좀더 구체화시킨다.</p>

⟨p> tag 안에 align 이라는 attributes을 추가적으로 사용하였다.

단순히 ⟨p>는 문단을 의미하지만 ⟨p>안에 정렬을 의미하는 align 이라는 attribute를 사용하면, 중앙정렬을 하는 문단을 의미한다.

## ❖ arguments

attribute 과 관련된 value(값) 이다.

위의 예제에서 align 이라는 attribute에 center라는 argument를 지정했다.

보통은 tag, attribute, argument 이와 같이 우리말로 굳이 바꾸어서 사용하지 않고 원어로 그대로 사용하는 경우가 많기 때문에 가능하면 원어로 학습하도록 한다.

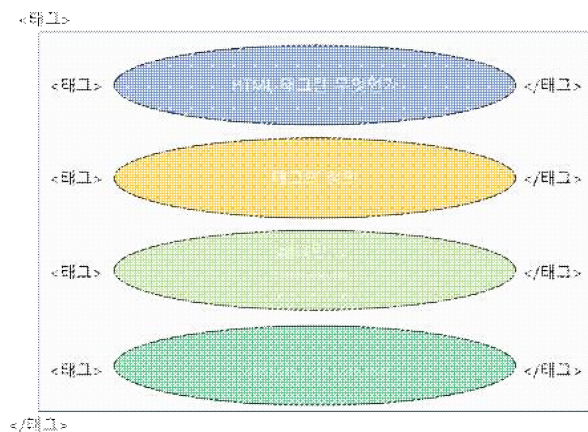
## ☑ HTML tag란 무엇인가?

그냥 무작정 tag를 공부하기 보단 그 전에 왜 tag라는 것이 존재하는가에 대해서 학습하도록 한다.

HTML(Hyper Text Markup Language)은 Web 문서를 작성하는 언어이다. HTML은 프로그래밍 언어의 범주에 들어있지만 범용적인 프로그래밍 언어와 달리 문서를 작성하는데 특화된 언어이다. 그래서 범용적인 프로그래밍 언어와는 구조적으로 많은 차이가 있다.

우선 HTML은 구조가 단순하다. 문서를 만드는 언어이므로 가장 중요한 것은 문서의 내용이다. HTML은 여기에 보조적인 역할을 하는 것이다. 예를들어 특정 텍스트에 링크가 걸려있는 것은 Web 문서에서 흔하게 볼 수 있다. 바로 HTML tag를 사용해서 이런 기능을 추가하는 것이다.

tag를 붙여 놓아서 이 부분이 문장인지 그림인지, HTML 문서인지, 표인지 리스트인지 구별할 수도 있고 tag에 attribute 라는 것을 추가해서 attribute을 설정할 수 있다.



이렇게 HTML 문서는 문서를 만드는데 특화시키기 위해서 대부분 tag로 이루어져 있다. tag가 없다고 문서의 내용이 변하는 것이 아니기 때문에 tag는 일종의 문서 양식을 만드는 역할을 하는 것이라 볼 수 있다.

## ☑ Web 문서를 만들 때 HTML는 꼭 필요한가?

일반적인 문서를 만들때 목적에 부합되는 양식에 따라서 문서를 작성한다. 이력서 양식을 생각해보면 쉽게 이해할 수 있다.

HTML 역시 정해진 양식이 있는데 사실 단순히 텍스트의 내용만 html 파일로 저장해 놓아도 Browser가 읽어서 문서를 표시해줄 수 있다.

다음은 위키백과에서 HTML에 대한 내용을 소개하는 페이지의 일부이다. 큰 제목으로 HTML이라고 표시되어 있고 구분선이 그어져 있다. 그 다음 내용이 들어가 있고 오른쪽에는 박스와 그 안에 내용들이 있는데 이 부분은 지금 생각하지 않는다.

## HTML

위키백과, 우리 모두의 백과사전.  
(Html에서 넘어옴)

**HTML**은 하이퍼텍스트 마크업 언어(HyperText Markup Language, 문화어: 초본문표식달기언어, 하이퍼본문표식달기언어)라는 의미의 **웹 페이지**를 위한 지배적인 마크업 언어다. HTML은 제목, 단락, 목록 등과 같은 본문을 위한 구조적 의미를 나타내는 것뿐만 아니라 링크, 인용과 그 밖의 항목으로 구조적 문서를 만들 수 있는 방법을 제공한다. 그리고 이미지와 객체를 내장하고 대화형 양식을 생성하는 데 사용될 수 있다. HTML은 웹 페이지 콘텐츠 안의 *꺾쇠 괄호*에 둘러싸인 "태그"로 되어있는 **HTML 요소** 형태로 작

HTML (HyperText Markup Language)	
개발자	W3C와 WHATWG
최근 버전	HTML5 <a href="#">↗</a>
최근 버전 출시일	2014년 10월 28일
미리보기 버전	HTML 5.1 <a href="#">↗</a> (초안)
미리보기 버전 출시일	2015년 10월 8일
주요 구현체	TEXT/HTML
영향을 받은 언어	SGML
영향을 준 언어	XHTML
웹사이트	<a href="http://www.w3.org/">http://www.w3.org/</a> <a href="#">↗</a>

이번에는 위 내용을 복사해서 html 확장자를 파일로 저장해서 Web 브라우저로 열어본다.

HTML 위키백과, 우리 모두의 백과사전. (Html에서 넘어옴) HTML은 하이퍼텍스트 마크업 언어(HyperText Markup Language, 문화어: 초본문표식달기언어, 하이퍼본문표식달기언어)라는 의미의 **웹 페이지**를 위한 지배적인 마크업 언어다. HTML은 제목, 단락, 목록 등과 같은 본문을 위한 구조적 의미를 나타내는 것뿐만 아니라 링크, 인용과 그 밖의 항목으로 구조적 문서를 만들 수 있는 방법을 제공한다. 그리고 이미지와 객체를 내장하고 대화형 양식을 생성하는 데 사용될 수 있다. HTML은 웹 페이지 콘텐츠 안의 *꺾쇠 괄호*에 둘러싸인 "태그"로 되어있는 **HTML 요소** 형태로 작성한다. HTML은 웹 브라우저와 같은 HTML 처리 장치의 행동에 영향을 주는 자바스크립트와 본문과 그 밖의 항목의 외관과 배치를 정의하는 CSS 같은 스크립트를 포함하거나 불러올 수 있다. HTML과 CSS 표준의 공동 책임자인 W3C는 명확하고 표상적인 마크업을 위하여 CSS의 사용을 권장한다.[1]

(한글이 혹시 깨져보일 수도 있다. 이럴 경우 tag로 문자셋(charset을 지정해 주어야 한다.)

위 문서는 무엇이 제목인지 어디에서 줄바꿈이 일어나는지 알 수가 없다. 위 두 문서의 차이는 HTML tag의 사용유무 이다. Browser는 HTML 문서를 해석할 때 HTML tag를 가지고 해석한다. 줄 바꿈 역시 tag로 표시하고 글자의 크기나 문자열에 연결된 링크를 만들려 해도 역시 HTML tag가 필요한 것이다. 이미지를 불러오거나, 표를 만들고 리스트를 만드는 작업 역시 tag가 필요하다. 위 문서처럼 HTML tag 없이 내용만 있는 문서는 글자와 공백(한칸 이상 띄어 쓸 수 없음)으로만 표현되게 된다.

## ☑ HTML 문서의 기본 구조

위 내용에 tag를 붙여서 HTML 기본 구조를 만들어 본다.

```
1 <!DOCTYPE HTML>
2 <HTML>
3   <HEAD>
4     <meta charset="UTF-8">
5     <title> HTML </title>
6   </HEAD>
7   <BODY>
8     <H1>HTML</H1>
9     <HR/>
10    <p>
11      <a href="http://thrillfighter.tistory.com" target="_blank">HTML</a>은
12      하이퍼본문표식달기언어라는 의미의 웹 페이지를 위한 지배적인 마크업 언어다. HTML은
13      단락, 목록 등과 같은 본문을 위한 구조적 의미를 나타내는 것뿐만 아니라 링크, 인용
14      밖의 항목으로 구조적 문서를 만들 수 있는 방법을 제공한다. 그리고 이미지와 객체를
15      대화형 양식을 생성하는 데 사용될 수 있다.
16    </p>
17    <p>
18      HTML은 웹 페이지 콘텐츠 안의 꺾쇠 괄호에 둘러싸인 "태그"로 되어있는 HTML 요소 형
19      태로 작성한다. HTML은 웹 브라우저와 같은 HTML 처리 장치의 행동에 영향을 주는 자바스크립트와 본문과 그
20      밖의 항목의 외관과 배치를 정의하는 CSS 같은 스크립트를 포함하거나 불러올 수 있다. HTML
21      표준의 공동 책임자인 W3C는 명확하고 표상적인 마크업을 위하여 CSS의 사용을 권장한다.[1]
22    </p>
23  </BODY>
24 </HTML>
```



tag는 <tag> 의 형식으로 표현한다. 시작 tag와 끝나는 tag가 쌍으로 존재하는 데 끝나는 tag는 tag이름 앞에 /(슬래시)를 붙인다.

### <tag> 내용 </tag>

- ☞ <HTML>....</HTML>은 HTML 문서라는 것을 표시하는 tag이다. 따라서 HTML 문서의 내용은 HTML tag로 둘러싸여 있다.
- ☞ HTML tag 내부를 보면 <HEAD>라는 tag와 <BODY>라는 tag가 있다.
- ☞ HEAD tag에는 문서의 정보를 알려주는 메타 정보들이 주를 이루고,



☞ BODY tag는 문서의 내용을 기술한다.

이 형식이 절대적이라기 보다는 일종의 약속이다. <HEAD> 내부에 <meta charset="UTF-8">이라는 것이 보이는데 이것은 meta라는 tag에 속성이 추가된 것이다.



☞ meta라하면 데이터를 설명해주는 정보를 의미하는데 meta tag는 문서의 정보를 기술해 준다. 예를 들어 meta tag의 charset 속성은 문서의 인코딩이 어떻게 되어있는지를 알려주는 속성이다. 보통 Web 상에서 문서가 깨져보이는 경우 charset이 제대로 설정되어 있지 않은 경우이다. 한글이 깨져 보인다면 위와 같이 설정해 준다.

☞ meta 뿐만아니라 다른 tag들도 tag에 attribute가 추가될 수 있다. attribute에 대한 것은 나중에 설명한다.

☞ title tag는 문서의 제목을 알려준다. Web 문서에 표시되는 내용은 아니고 Web 문서에 대한 정보를 보여주므로 HEAD tag 내에 넣어주었다. 그러나 BODY tag에 넣어주어도 문서를 표시하는데는 지장이 없다. 그러나 양식대로 문서를 작성할 것을 권장한다.

## <title> 제목 </title>

☞ 앞서 그림에서처럼 Web 문서의 제목이 Browser저의 제목 표시줄에 표시되는데 이것이 title tag가 하는 일이다.

☞ <BODY> tag 내부에는 문서의 실질적인 내용을 기술한다.

☞ H1 tag는 headline으로 머릿말을 뜻한다. H1부터 H6까지의 tag가 존재하는데 H1이 제일 큰 글씨로 표시되고 H6가 가장 작은 글씨로 표시된다.

☞ 그 밑에 <HR/>이라는 표시는 내용 구분선이다. 일종의 나홀로 tag인데 이런 tag는 시작과 끝이 없이 단독으로 사용된다.

☞ 참고로 띄어쓰기를 위한 <br/>tag가 있다. 이 tag는 나홀로 tag이다. 즉 시작과 끝이 없이 단독으로 쓰는 tag 이다.

☞ &nbsp;는 띄어쓰기를 나타낸다.

☞ <p> tag는 한 문단을 의미한다. 위 그림처럼 문단과 문단간에는 구분될 정도의 띄어쓰기가 되어있다. <p> tag를 사용하여 구분하는 것과 <br/> tag를 사용해서 문단을 구분하는 것 중에 어느 것이 옳은가에 대해서는 정답은 없다. 다만 내용을 tag로 묶어 놓게 되면 tag에 속성을 지정하여 일괄적으로 속성값을 줄 수 있다는 편리성이 있다. 문단을 일괄적으로 관리하는 측면에서 <p> tag를 사용하는 편이 좋다.

☞ a tag는 추가적으로 링크를 달아볼 수 있다. HTML이라는 문자열에 하이퍼링크를 달아서 다른 사이

트의 주소를 연결해본다.

```
<BODY>
<H1>HTML</H1>
<p>
  <a href="http://thrillfighter.tistory.com">HTML</a>은
  하이퍼본문표식달기언어)라는 의미의 웹 페이지를 위한 지배적인 마크
  단락, 목록 등과 같은 본문을 위한 구조적 의미를 나타내는 것뿐만
  밖의 항목으로 구조적 문서를 만들 수 있는 방법을 제공한다. 그리고
  대화형 양식을 생성하는 데 사용될 수 있다.
</p>
<p>
  HTML은 웹 페이지 콘텐츠 안의 꺾쇠 괄호에 둘러싸인 "태그"로 되어
  HTML은 웹 브라우저와 같은 HTML 처리 장치의 행동에 영향을 주는 것
```

**<a href="http://thrillfighter.tistory.com"> HTML </a>**

☞ <a> tag는 meta tag 처럼 속성을 추가적으로 사용해야 의미가 있다. href는 바로 링크시킬 주소를 설정하는 속성이다. 이렇게 링크를 걸어놓으면 HTML에 링크가 표시된다. 클릭하면 해당 Web Page로 이동한다.

☞ 여기에 target이라는 속성을 추가적으로 붙여본다.

**<a href="http://thrillfighter.tistory.com" target="\_blank"> HTML </a>**

☞ target속성의 속성값이 "\_blank"로 지정되면 하이퍼링크로 열리는 Web 문서가 새로운 창으로 열린다. 이렇게 속성은 여러 개를 지정할 수도 있다.

## ☑ HTML 문서에 주석 넣기

주석은 한마디로 설명이다. 책의 내용 옆에 모르거나 중요한 내용을 메모해 놓듯이 바로 주석도 마찬가지로 주석을 달아두어서 이해하기 힘든 부분이나 해당 코드가 하는 기능 또는 나중에 수정되어야 할 부분 표시 등을 표시해 놓기 위해서 주석을 사용할 수 있다.

주석은 Web Browser에 의해서 HTML 코드가 해석될 때 무시되므로 문서에 영향을 미치지 않는다.

**<!-- 주석 내용 작성 -->**

그렇다고 주석을 남발하는 것은 좋지 않다. 설명을 해 놓는 것은 좋지만 너무 많은 설명은 오히려 코드의 이해를 방해한다. 주석을 작성하는 요령은 정말 필요한 내용만 그리고 설명은 심플하면서도 명확하게 하는 것이다.



### 03. html 표 만들기

#### ☑ html 표 만들기 tag의 구조 예제

```
<table>
  <tr>
    <td> 값 </td>
    .....
  </tr>
  <tr>
    <td> 값 </td>
    .....
  </tr>
</table>
```

표는 행과 열로 구성되어 있다. HTML 문서라고 다를 것은 없다. HTML 문서에서 표를 만들기 위해서는 세 개의 tag를 사용하는데 table tag, tr tag, td tag 이다.

- ☞ <table> tag는 표의 시작과 끝을 알려주는 tag 이다. 실질적으로 <table> tag가 표를 그려주는 것은 아니다.
- ☞ tr tag는 row(행)을 그려준다. 앞에서 보여준 html 예제는 tr tag가 두 번 사용되었다. tag 시작과 끝을 포함해서 <tr> ... </tr> 을 한번으로 친다. 그리고 한 번의 tr tag는 하나의 행을 만들어 준다.
- ☞ 따라서 앞에서 예로든 html 테이블은 두 개의 행을 가질 것이다.
- ☞ table tag와 tr tag만 있는 표는 다음과 같아야 한다.(하지만 실제 이렇게 표시되지는 않는다.)

표

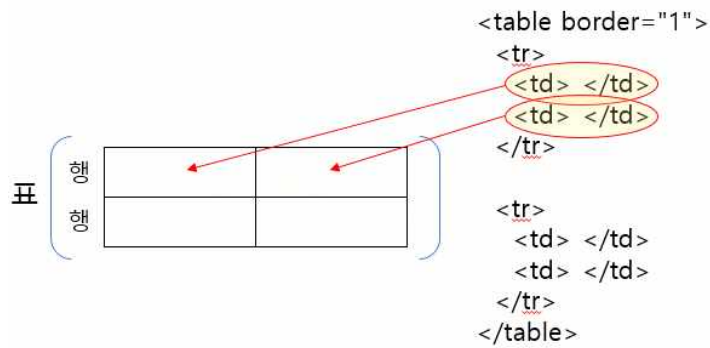
행
행


하지만 실제로 위 그림처럼 표시되지 않고 다음과 같이 네모 표시가 될 뿐이다.

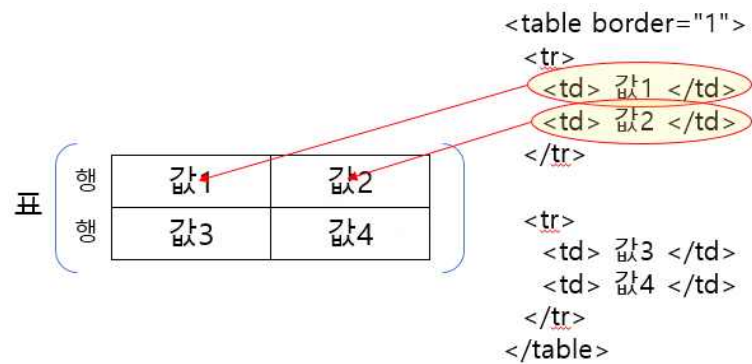
□

왜냐면 표가 제대로 정의되기 위해서는 <td> tag까지 적용되어야 하기 때문이다.

- ☞ td tag는 각각의 행을 나누어주는 역할을 한다. 따라서 <tr> tag 안에 넣어주어야 하고 <tr> tag 안에 있는 <td> tag가 개수가 해당 행의 column(열)의 개수가 된다.



실제로 Web브라우저에는  이렇게 표시된다. 빈 구멍이 네개가 난 레고블럭 같은 모양인데 이 구멍에 다음과 같이 내용을 넣을 수 있다.



이렇게 <td> tag는 각각의 행에서 열을 나누는 동시에 <td> 값 </td> 의 형태로 값을 넣는 기능까지 한다.

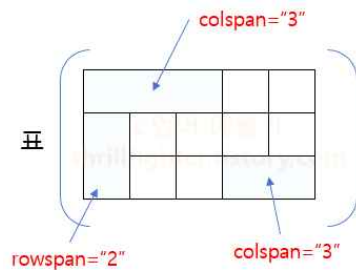
❖ 참고 : table tag에 border 속성을 넣어서 테이블의 외곽선의 굵기를 정할 수 있다.

```
<table border="1">
....
</table>
```

border 속성이 없다면 테이블의 외곽선은 없다.

## ☑ 복잡한 표 만들기

n x m 형태의 표 말고 다음 그림과 같이 복잡한 형태의 표도 만들 수 있다.



이런 표를 만들기 위해서 colspan과 rowspan 속성에 대해서 이해해야 한다.

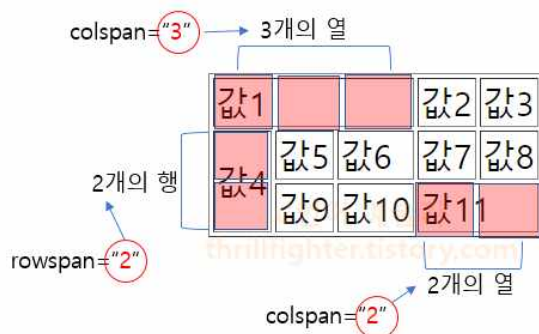
☞ span의 뜻은 한 뼘, 또는 폭, 기간 등과 같이 길이를 의미한다. 따라서 colspan과 rowspan은 열과 행에 대한 길이를 의미할거라 짐작된다.

위와 같은 형태의 표를 만드는 소스를 먼저 살펴보자.

```
<table border="1">
  <tr>
    <td colspan="3"> 값1</td>
    <td> 값2</td>
    <td> 값3</td>
  </tr>
  <tr>
    <td rowspan="2"> 값4</td>
    <td> 값5</td>
    <td> 값6</td>
    <td> 값7</td>
    <td> 값8</td>
  </tr>
  <tr>
    <td> 값9</td>
    <td> 값10</td>
    <td colspan="2"> 값11</td>
  </tr>
</table>
```

값1			값2	값3
값4	값5	값6	값7	값8
	값9	값10	값11	

<td> tag의 속성으로 colspan과 rowspan 값이 설정된다.



colspan="n"이라 하면 n개의 열을 합쳐서 쓴다는 말이다. 위에서는 처음 colspan="3"이므로 3개의 열의 크기에 하나의 값만 들어가게 된다. 이 이후 td tag로 넣은 값들이 글을 읽는 방향으로 차례대로 채워진다.

<tr> tag는 행을 나눌 뿐이다.

rowspan 속성이 와도 동일하다. 위 코드에서 rowspan="2"는 두 행을 하나로 쓴다는 말이다. 그 다음 <td> tag에 의해 값5, 값6, ... 이 차례대로 채워지고 <tr> tag에 의해서 행이 나누어진다.

앞에서 rowspan으로 이미 쓰여진 칸을 제외하고 값9부터 차례대로 채워진다. 나머지는 쉽게 이해할 수 있을 것이다. 한 가지 더 colspan과 rowspan을 동시에 쓸 수도 있다.

## 04. HTML tag 연습 리스트를 만드는 ul, ol, dl 그리고 li tag

앞서 학습한 표를 만드는 <table> tag와 더불어 오늘 학습할 리스트를 만드는 tag는 layout을 설정하는데도 사용되고 사용 빈도도 높으므로 꼭 알아 두어야 하는 기본적인 tag 이다.

사용법은 <table> tag와 비슷하다. 다음은 테이블 tag의 형식이다.

```
<table>
  <tr>
    <td> .. </td>
    ...
  </tr>
  <tr>
    <td> .. </td>
    ....
  </tr>
</table>
```

### ☑ ol tag, ul tag

이번에는 리스트 tag를 보자.

```
<ol>
  <li>아이템 1 </li>
  <li>아이템 2 </li>
</ol>
```

테이블의 시작과 끝을 알려주는 table tag 처럼 리스트도 리스트의 시작과 끝을 알려주는 ol, ul tag가 있다. ol, ul 둘 다 리스트지만 형식이 다르다.

```
<ul>
  <li>아이템 1 </li>
  <li>아이템 2 </li>
</ul>
```

☞ ol은 ordered list의 약자로 순서가 있는 리스트라는 의미다. 그래서 아이템 앞에 번호가 매겨진다.

1.   아이템 1
2.   아이템 2

☞ ul은 unordered list의 약자다. 순서는 없고 다음과 같이 표시된다.

- 아이템 1
- 아이템 1

이렇게 표시가 된다.

ol tag와 ul tag의 차이는 다음 그림을 참고하자



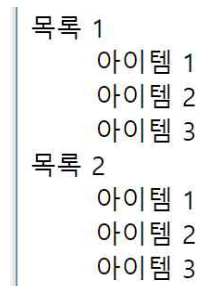
## ☑ dl 리스트 tag

<ol> tag, <ul> tag와 같은 리스트 tag 지만 형식이 좀 다르다. 이 tag는 하위 tag로 <dt> tag와 <dd> tag 두 개를 사용한다. 따라서 하나의 하위 항목을 더 가질 수 있다.

사용법은 다음과 같다.

```
<dl>
  <dt> 목록 1 </dt>
  <dd> 아이템 1 </dd>
  <dd> 아이템 2 </dd>
  <dd> 아이템 3 </dd>
  <dt> 목록 2 </dt>
  <dd> 아이템 1 </dd>
  <dd> 아이템 2 </dd>
  <dd> 아이템 3 </dd>
</dl>
```

다음은 브라우저에서 위 코드를 실행한 화면이다.



리스트 tag를 이용해서 Web Site의 layout을 잡을 수 있다. 인터넷 뉴스 목록과 썸네일을 리스트 tag로 구성할 수도 있다. 그 때는 텍스트 대신 <img> tag를 목록에 넣어주면 된다.

[ index01.html ]

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
  </head>
  <body>
    안녕하세요. <strong>세명대학교 스마트IT 학부</strong>입니다.
  </body>
</html>
```

[ index02.html ]

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
  </head>
  <body>
    안녕하세요. <a href="http://www.semyung.ac.kr/smartit.do" target="_blank">스마트IT 학부</a>입니다.
  </body>
</html>
```

[ index03.html ]

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
  </head>
  <body>
    <ol>
      <li>HTML5</li>
      <li>CSS3</li>
      <li>JavaScript</li>
    </ol>
    <ul>
      <li>SmartIT01</li>
      <li>SmartIT02</li>
      <li>SmartIT03</li>
    </ul>
  </body>
</html>
```

[ index04.html ]

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
</head>
<body>
  <script>
    password = prompt("비밀번호");
    if(password == 3333) {
      document.write("안녕하세요. SmartIT 님");
    } else {
      document.write("누구신지요?");
    }
  </script>
</body>
</html>
```