

# 1 HTML

## 1. HTML 이해하기

### 1) HTML 소개

#### HTML이란?

HTML은 프로그래밍 언어로 웹 페이지를 만들 때 사용됩니다.  
웹 사이트들이 모두 HTML을 사용하여 만들어집니다.

#### HTML의 의미와 특징

HTML은 Hyper Text Markup Language의 줄임말입니다.

Hyper Text는 단순한 텍스트를 넘어서 웹 페이지의 특정 부분과 연결할 수 있는 기능을 가진 텍스트 즉, 링크를 의미합니다. Markup Language는 프로그래밍 언어의 한 종류로 정보를 구조적, 계층적으로 표현 가능하다는 특징이 있습니다. HTML은 파일 확장자로 .html을 쓰며, 그 파일 안에 html 코드를 작성하게 됩니다.

#### HTML의 역사

HTML은 1990년대 영국의 물리학자 팀 버너스리가 제안하여 개발되었습니다.  
초기 개발 목적은 연구소의 연구원들이 신속하게 정보와 문서를 공유하기 위해서였습니다.  
현재의 웹은 문서의 형태는 아니지만, 정보를 공유한다는 목적은 여전히 같습니다

### 2) HTML 문법 - 태그

#### 태그란?

HTML은 태그들의 집합이며, 태그는 HTML에서 가장 중요하고 기본이 되는 규칙입니다.  
태그는 '무언가를 표시하기 위한 꼬리표, 이름표'라는 의미가 있으며, HTML에서도 이와 비슷한 의미로 해석됩니다.  
우리가 다양한 태그들을 이용해 코드를 작성하면, 브라우저가 이를 인식해 내용을 표현하게 됩니다.

#### 태그를 사용하는 방법

시작 태그와 종료 태그 사이에 실제 화면에 나타나는 내용이 위치하게 됩니다.

`<h1>Hello, HTML</h1>` 태그의 가장 기본적인 예로, `<h1>`을 사용해 'Hello, HTML'을 출력하는 코드입니다.

#### 요소란?

내용을 포함한 태그 전체를 요소(Element)라고 합니다.

태그와 요소는 의미가 다르지만 많은 사람이 태그와 요소를 같은 의미로 사용하니 혼동하지 않도록 주의해야 합니다.

#### 정리

HTML에는 많은 종류의 태그들이 있습니다.

태그는 각각의 의미가 있으며 이 의미에 맞게 태그를 사용해야 합니다.

또한, 태그는 HTML뿐만 아니라 XML, SGML, XHTML 등 다양한 마크업 언어에서 모두 사용됩니다.

### 3) HTML 문법 - 속성

속성이란?

속성은 태그에 추가로 정보를 제공하거나 태그의 동작이나 표현을 제어할 수 있는 설정값을 의미합니다.

속성을 사용하는 방법

속성은 이름과 값으로 이루어져 있습니다.

시작 태그에서 태그 이름 뒤에 공백으로 구분 후 속성 이름="속성값"으로 표현합니다.

속성값은 홑따옴표(')와 쌍따옴표(")로 감싸 표현합니다.

```
<h1 id="title">Hello, HTML</h1>
```

위는 <h1>에 id 속성을 추가해 title 값을 선언한 코드입니다.

여러 속성을 사용하는 방법

의미와 용도에 따라 여러 속성이 존재하며 하나의 태그에 여러 속성을 선언할 수 있습니다.

여러 속성을 선언할 때는 공백으로 구분해서 사용합니다.

```
<h1 id="title" class="main">Hello, HTML</h1>
```

위는 <h1>에 id와 class 2개의 속성을 선언한 코드입니다. 속성의 선언 순서는 태그에 영향을 미치지 않으며 class를 id보다 먼저 선언해도 결과는 같습니다.

속성의 종류

속성은 종류에 따라 모든 태그에 사용할 수 있는 글로벌 속성과 특정 태그에서만 사용할 수 있는 속성으로 구분됩니다. 또한, 선택적으로 쓸 수 있는 속성과 특정 태그에서 필요한 필수 속성으로 구분됩니다. 위의 예시에서 쓰인 id와 class 속성은 글로벌 속성입니다.

### 4) HTML 문법 - 태그 중첩

### 5) HTML 문법 - 빈 태그

빈 태그란?

- <br>
- <img src="">
- <input type="">

위는 빈 태그의 예시입니다.

빈 태그는 내용이 없어서 종료 태그가 필요하지 않습니다.

빈 태그의 특징

빈 태그는 내용만 비어있을 뿐 속성을 통해서 화면에 나타내거나 화면에 표시되지 않더라도 다른 용도로 사용되는 태그입니다. 빈 태그의 대표적인 경우는 브라우저가 직접 화면에 내용을 그려줘야 하는 경우입니다. 이런 태그는 브라우저가 내용을 대체한다고 하여 replacement 태그, 대체되는 태그라고 합니다. 빈 태그에 대체되는 태그만 있는 것은 아니며

실제로 화면에 출력될 내용이 없어 다른 용도로 쓰이는 태그도 존재합니다. 위 예시 코드의 <br>이 바로 이 경우입니다. <br>에 대해서는 다음에 좀 더 자세하게 살펴보도록 하겠습니다.

### 6) HTML 문법 - 공백

#### HTML에서의 공백

기본적으로 HTML은 두 칸 이상의 공백을 모두 무시합니다.

```
<h1>Hello, HTML</h1>
```

```
<h1>Hello,  HTML</h1>
```

```
<h1>
```

```
Hello,  
HTML  
</h1>
```

HTML은 두 칸 이상의 공백과 개행을 모두 무시하기 때문에 위 세가지 모두 같은 텍스트가 화면에 나타나게 됩니다.

## 7) HTML 문법 - 주석

### HTML에서의 주석

주석은 화면에 노출되지 않고 메모의 목적으로만 사용하는 것을 의미합니다.

HTML 파일 내에 주석으로 표시를 해주면 브라우저는 해당 부분을 인식하여 해석하지 않습니다.

```
<!-- 여기에 작성되는 내용들은 모두 주석 처리가 됩니다. -->  
<!-- 주석은 여러 줄로 작성할 수도 있습니다.  
    <h1>Hello, HTML</h1>  
    위 <h1>태그는 브라우저가 해석하지 않습니다.  
-->
```

주석의 시작은 <!-- 로 표시하고, --> 표시로 종료합니다.

## 8) 문서의 기본 구조

```
<!DOCTYPE html>  
<html lang="ko">  
  <head>  
    <meta charset="UTF-8">  
    <title>HTML, jueun</title>  
  </head>  
  <body>  
    <h1>Hello, HTML</h1>  
  </body>  
</html>
```

### 문서 타입 정의

문서 타입 정의는 보통 DTD(doctype)라고 부릅니다.

이 문서가 어떤 버전으로 작성되었는지 브라우저에 알려주는 선언문이며 반드시 문서 내 최상단에 선언되어야 합니다.

### <html>요소

문서 타입 선언 후에는 <html>태그가 나와야 하고, 자식으로는 <head>태그와 <body>태그가 있습니다.

<html> 태그의 lang 속성은 문서가 어느 언어로 작성되었는지를 의미합니다.

<head>태그에 위치하는 태그들은 브라우저 화면에 표시되지 않습니다.

대신 문서의 기본 정보 설정이나 외부 스타일 시트 파일 및 js 파일을 연결하는 등의 역할을 합니다.

<meta> 태그의 charset 속성은 문자의 인코딩 방식을 지정합니다.

<body>태그에는 실제 브라우저 화면에 나타나는 내용이 들어가며, 앞으로 우리가 다루는 태그들 대부분이 모두 여기에 해당이 됩니다.

위 코드는 가장 기본적인 문서 구조로, 보통은 이보다 더 많은 태그가 들어가게 됩니다.

## 2. HTML 태그

### 1) HTML 태그 소개

통계 사이트: <https://www.advancedwebranking.com/html/#overview>

레퍼런스 3개: <https://developer.mozilla.org>, <http://html5doctor.com>, <https://www.w3schools.com>

### 2) 제목과 단락요소

#### 제목 태그

제목(heading) 태그는 문서 내에 제목을 표현할 때 사용하는 태그입니다.

태그 이름은 heading을 줄여서 h로 쓰며, 제목의 레벨에 따라서 <h1>~<h6>까지 있습니다.

보통 <h1>은 해당 페이지를 대표하는 큰 제목으로 주로 사용되며, 숫자가 올라갈수록 조금 더 낮은 수준의 소제목을 나타내게 됩니다.

하지만 현재 웹 페이지의 내용은 제목과 본문 식의 문서 형태보다는 주로 이미지나 그림처럼 시각적인 형태로 표현되고 있어서 제목 태그를 <h6>까지 쓰는 경우는 거의 없습니다.

#### 단락 태그

단락(paragraph) 태그는 paragraph를 줄여서 p로 씁니다.

#### 개행

따라서 개행을 위해 쓰이는 태그가 바로 <br>입니다. (linebreak 를 줄여서 br 이라고 합니다.)

<br>은 닫기 태그와 내용이 존재하지 않는 빈 태그입니다.

### 3) 텍스트를 꾸며주는 요소

#### 텍스트 표현 태그

웹 표준화가 대두하면서 웹 문서의 구조와 표현을 분리하였습니다.

그 과정에서 많은 표현용 태그들이 사라졌고, 지금은 표현용 태그가 얼마 남지 않았습니다.

그 중 <b>, <i>, <u>, <s>에 대해 살펴보도록 하겠습니다.

- <b> : bold 태그는 글자를 굵게 표현하는 태그입니다.
- <i> : italic 태그는 글자를 기울여서 표현하는 태그입니다.
- <u> : underline 태그는 글자의 밑줄을 표현하는 태그입니다.
- <s> : strike 태그는 글자의 중간선을 표현하는 태그입니다. (예전에 존재했던 strike 태그와는 다른 태그로, strike 태그는 폐기되어 더는 사용할 수 없습니다.)

[https://developer.mozilla.org/en-US/docs/Web/HTML/Element#Inline\\_text\\_semantics](https://developer.mozilla.org/en-US/docs/Web/HTML/Element#Inline_text_semantics)

### 4) 앵커 요소

<a> \_ 인라인!!!!!!!!!!!!!! 뭔가 해주고 싶을 땐 안에 span하고 span의 display: block으로 하더라~

<a>(anchor 태그)는 a태그, 앵커, 링크 등 여러 이름으로 불립니다.

<a href="http://www.naver.com/" target="\_blank">네이버</a>

#### href 속성

링크를 만들기 위해 <a>는 반드시 href(hypertext reference) 속성을 가지고 있어야 합니다.

href 속성의 값은 링크의 목적지가 되는 URL입니다.

#### target 속성

target 속성은 링크된 리소스를 어디에 표시할지를 나타내는 속성입니다.

속성값으로는 \_self, \_blank, \_parent, \_top이 있습니다.

\_self는 현재 화면에 표시한다는 의미로, target 속성이 선언되지 않으면 기본적으로 self와 같이 동작합니다.  
\_blank는 새로운 창에 표시한다는 의미로 외부 페이지가 나타나게끔 하는 속성입니다.  
\_parent와 \_top은 프레임이라는 특정 조건에서만 동작하는 속성으로,  
최근에는 프레임을 잘 쓰지 않기 때문에 따로 다루지 않고 넘어가겠습니다.

### 기타 속성

<a>에는 이 외에도 많은 속성이 있습니다.

아래 링크를 참고해 다루지 않은 나머지 속성을 직접 공부하기를 권장합니다.

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

(참고로 속성 이름 옆 html5 레이블로 표시된 것은 html5 버전이 나오면서 새로 생긴 속성이라는 뜻입니다.

Obsolete attributes로 표시된 부분은 폐기된 속성들이므로 보지 않으셔도 됩니다.)

### 내부링크

<a>를 통해 만들어진 링크가 꼭 외부 페이지로만 이동하는 것은 아닙니다.

<a>를 통해 페이지 내부의 특정 요소로 초점을 이동할 수도 있는데, 이를 내부 링크라고 합니다. 내부 링크를 사용할 때는 href 속성값에 #을 쓰고 그 뒤에 페이지 내에서 이동하고자 하는 요소의 id 속성값을 적으면 됩니다.

<a href="#some-element-id">회사 소개로 이동하기</a>

... 중략.

<h1 id="some-element-id">회사 소개</h1>

## 5) 의미가 없는 컨테이너 요소

### 의미가 없는 컨테이너 요소

태그 자체에 아무 의미가 없으며, 단순히 요소들을 묶기 위해 사용되는 태그입니다.

스타일을 주거나 서버에 보내는 데이터를 담기 위한 용도로 이런 의미 없는 요소들이 사용됩니다.

이런 의미 없는 태그의 사용빈도는 매우 높습니다.

그 이유는 html 태그들은 문서를 웹에 나타내기 위해 제작되어 기본적으로 문서에 최적화되어있는 의미를 지니는데,

현재 웹의 형태는 문서 형태에서 많이 벗어났기 때문입니다.

다행히 HTML 버전이 업그레이드되면서 웹에 알맞은 태그들이 많이 생겼습니다.

가장 대표적으로 많이 쓰이는 의미가 없는 태그는 <div>, <span>입니다.

### <div>태그와 <span>태그

div(division) 태그는 블록 레벨 태그입니다.

블록 레벨 요소는 기본적으로 한 줄을 생성해서 내용을 표현합니다.

반면, span 태그는 인라인 레벨 태그입니다.

인라인 레벨 요소들은 블록 레벨 요소의 한 줄 안에서 표현되는 요소들입니다.

<div>, <span>는 모두 아무 의미가 없으므로 실제 브라우저도 별다른 스타일을 적용하지 않습니다.

## 6) 리스트 요소

### <ul>

ul(unordered list) 태그는 순서가 없는 리스트를 표현할 때 사용합니다.

<ul>

<li>콩나물</li>

<li>파</li>

<li>국 간장</li>

</ul>

### <ol>

ol(ordered list) 태그는 순서가 있는 리스트를 표현할 때 사용합니다.

<ol>

```
<li>냄비에 국물용 멸치를 넣고 한소끔 끓여 멸치 육수를 7컵(1,400ml) 만든다.</li>
<li>콩나물을 넣고 뚜껑을 덮어 콩나물이 익을 때까지 끓인다.</li>
<li>뚜껑을 열고 대파, 마늘, 고춧가루를 넣고 끓인다.</li>
</ol>
```

## <dl>태그

dl(definition/description list) 태그는 용어와 그에 대한 정의를 표현할 때 사용합니다.

<dt> : definition term 용어를 나타내는 태그

<dd> : definition description 용어에 대한 정의 또는 설명을 나타내는 태그 \_ 여러개 가능

```
<dl>
```

```
<dt>리플리 증후군</dt>
```

```
<dd>허구의 세계를 진실이라 믿고 거짓된 말과 행동을 상습적으로 반복하는 반사회적 성격장애를 뜻하는 용어</dd>
```

```
<dt>피그말리온 효과</dt>
```

```
<dd>타인의 기대나 관심으로 인하여 능률이 오르거나 결과가 좋아지는 현상</dd>
```

```
<dt>언더독 효과</dt>
```

```
<dd>사람들이 약자라고 믿는 주체를 응원하게 되는 현상</dd>
```

```
</dl>
```

예제.

```
<body>
```

```
<h1>월드컵 조 편성 </h1>
```

```
<ol>
```

```
<li>
```

```
A조
```

```
<ul>
```

```
<li>나라1</li> <li>나라2</li>
```

```
</ul>
```

```
</li>
```

```
<br>
```

```
<li>
```

```
B조
```

```
<ul>
```

```
<li>나라1</li> <li>나라2</li>
```

```
</ul>
```

```
</li>
```

```
<br>
```

```
<li>
```

```
C조
```

```
<ul>
```

```
<li>나라1</li> <li>나라2</li>
```

```
</ul>
```

```
</li>
```

```
</ol>
```

```
</body>
```

## 7) 이미지 요소

### <img>

<img>는 문서에 이미지를 삽입하는 태그로, 닫는 태그가 없는 빈 태그입니다.

```

```

#### src 속성

<img>의 필수 속성으로 이미지의 경로를 나타내는 속성입니다.

#### alt 속성

이미지의 대체 텍스트를 나타내는 속성입니다.

대체 텍스트는 이미지를 대체하는 글을 뜻하며, 웹 접근성 측면에서 중요한 속성입니다.

src 속성과 더불어 반드시 들어가야 하는 속성입니다.

#### width/height 속성

이미지의 가로/세로 크기를 나타내는 속성입니다.

값의 단위는 필요하지 않으며, 값을 입력하면 자동으로 픽셀 단위로 계산됩니다.

width/height는 선택적인 속성이지만 이미지의 크기가 고정적이라면 width/height 속성을 선언하는 것이 성능적인 측면에서 좋습니다.

width/height 속성이 없으면 이미지는 원본 크기대로 노출되며,

둘 중 하나만 선언하면 나머지 한 속성은 선언한 속성의 크기에 맞춰 자동으로 비율에 맞게 변경됩니다.

반면 두 속성 모두 선언하면 이미지는 비율과 무관하게 선언한 크기대로 변경됩니다.

#### 상대경로와 절대경로

src 속성에는 이미지의 경로가 들어가며, 이미지의 상대경로와 절대경로가 있습니다.

상대경로는 현재 웹 페이지를 기준으로 상대적으로 이미지의 위치를 나타내는 경로이고,

절대경로는 실제 그 이미지가 위치한 곳의 전체 경로입니다.

<!-- 상대경로 -->

```

```

<!-- 절대경로 -->

```

```

```

```

#### 이미지 파일 형식

- gif : 제한적인 색256을 사용하고 용량이 적으며 투명 이미지와 애니메이션 이미지를 지원하는 형식
- jpg : 사진이나 일반적인 그림에 쓰이며 높은 압축률과 자연스러운 색상 표현을 지원하는 형식 (투명X)
- png : 이미지 손실이 적으며 투명과 반투명을 모두 지원하는 형식

## 8) 테이블 요소 1

<table>:테이블 태그

<td>데이터 셀

<th>셀 제목

<tr>행

<caption>표의 제목

<thead>제목행 그룹

<tbody>본문행 그룹

<tfoot>

<td colspan = "2"> </td>: col 먹기, 가로로길어짐  
rowspan:세로 방향 병합

```
<!doctype html>
<html lang = "ko">
  <head>
    <meta charset="utf- 8">
    <title>table</title>
    <style media="screen">
      th, td {border: 1px solid; width: 50px; height: 50px}
    </style>
  </head>

  <body>
    <table>
      <caption> Monthly Savings! </caption>

      <thead>
        <tr>
          <th>Monthly</th><th>Savings</th>
        </tr>
      </thead>

      <tbody>
        <tr>
          <td>January</td><td>100</td>
        </tr>
        <tr>
          <td>Feb</td><td>100</td>
        </tr>
        <tr>
          <td>March</td><td>100</td>
        </tr>
      </tbody>

      <tfoot> -> 순서 지키기!
        <tr>
          <td>Sum</td><td>300</td>
        </tr>
      </tfoot>

    </table>
  </body>
```



</html>

## 9) 테이블 요소 2

Specification values				
Grade.	Point.	Strength.		Percent.
		kg/mm	lb/in	
Hard	0.45	56.2	80,000	20
Medium	0.45	49.2	70,000	25
Soft	0.45	42.2	60,000	30

실습, 코드는 파일에. 표를 그릴 때는 행렬을 먼저 세자.

<colgroup>, <col> 태그와 속성인 scope, headers 잘 사용되니까 숙제래

## 10) 폼 요소 1

**type="text"**

주로 아이디, 이름, 주소, 전화번호 등 단순한 텍스트를 입력할 때 사용합니다.

type="text"에는 placeholder 속성이 존재합니다. placeholder 속성은 사용자가 입력하기 전 미리 화면에 노출하는 값으로, 입력하는 값의 양식을 표현할 수 있습니다.

아이디: <input type="text" placeholder="영문으로만 써주세요"> <br>

**type="password"**

암호와 같이 공개할 수 없는 내용을 입력할 때 사용합니다.

화면에는 type="text"와 같게 나타나지만 실제로 입력할 때 값을 노출하지 않습니다.

비밀번호: <input type="password"> <br>

**type="radio"**

라디오 버튼은 중복 선택이 불가능하며 하나의 항목만을 선택해야 합니다.

성별: 남자<input type="radio" name="gender" checked> 여자<input type="radio" name="gender"> <br>

**type="checkbox"**

체크박스를 만들 때 사용합니다. 체크박스는 중복 선택이 가능합니다.

취미: 영화감상<input type="checkbox" name="hobby" checked>, 음악감상<input type="checkbox" name="hobby">, 독서<input type="checkbox" name="hobby">

**checked** 속성은 값이 별도로 존재하지 않는 boolean 속성입니다.

boolean 속성은 true/false 둘 중 하나의 값을 가지며 속성이 존재하면 true, 속성이 존재하지 않으면 false를 가집니다.

**name** 속성은 라디오 버튼과 체크박스를 그룹화시켜주는 속성입니다.

## 11) 폼 요소 2

**type="file"**

파일을 서버에 올릴 때 사용합니다.

브라우저에 따라 표현되는 형태는 조금씩 다르지만, 모두 같은 역할을 합니다.

프로필 사진 업로드: <input type="file"> <br>

type="submit|reset|image|button"

submit, reset, image, button 타입은 모두 클릭 가능한 버튼을 만듭니다.

- submit : form의 값을 전송하는 버튼
- reset : form의 값을 초기화하는 버튼
- image : 이미지를 삽입할 수 있는 버튼 (submit과 동작이 동일함)
- button : 아무 기능이 없는 버튼

```
<input type="submit"><br>
<input type="submit" value="전송"><br>
<input type="reset"><br>
<input type="reset" value="취소"><br>
<input type="button" value="등록"><br>
```

이미지 버튼은 이미지 관련 속성인 src, alt 속성이 반드시 필요하며 width/height 속성을 적용할 수도 있습니다.

```
<input type="image" src="C:\Users\Jueun\Pictures\sally.png" alt="로그인" width="100" height="100">
```

## 12) 폼 요소 3

### <select>

<select>는 선택 목록 상자 또는 콤보박스라고도 합니다.

몇 개의 선택지를 리스트 형태로 노출하고 그중 하나를 선택할 수 있게 하는 태그입니다. (multiple 속성을 사용하면 다중 선택도 가능합니다.) <select>내부의 <option>으로 각 항목을 나타냅니다.

<option>의 속성으로는 selected가 있으며 이는 선택된 항목을 의미합니다.

사는 지역: <select>

```
<option>서울</option>
<option>인천</option>
<option>경기</option>
<option selected>제주</option>
</select><br>
```

### <textarea>

한 줄만을 입력할 수 있는 <input type="text" >와 달리 여러 줄의 텍스트를 입력할 때 사용합니다. <textarea>에는 텍스트 상자의 크기를 조절하는 rows, cols 속성이 있습니다.

```
자기소개 <textarea rows="5" cols="30" placeholder="자기소개 짧게"> </textarea> <br>
```

### <button>

버튼을 만들 때 사용하며 submit, reset, button 3가지의 타입이 있습니다.

각 버튼은 이전에 배웠던 input 타입의 submit, reset, button과 모두 같은 기능을 가진 버튼입니다.

다만, 빈 태그가 아니며 내용을 안에 직접 넣을 수 있으므로 좀 더 자유로운 스타일 표현이 가능합니다.

```
<button type="submit"> 전송 </button><br>
<input type="submit" value="전송"><br>
<button type="reset"> 취소 </button><br>
<input type="reset" value="취소"><br>
```

## 13) 폼 요소 4

### <label>

<label>은 form 요소의 이름과 form 요소를 명시적으로 연결시켜주기 위해 사용됩니다.  
모든 form 요소에 사용할 수 있습니다.

### <fieldset>, <legend>

<fieldset>, <legend>는 form 요소를 구조화 하기 위해 필요한 태그입니다. 네모칸 생김!

- <fieldset> : 여러 개의 폼 요소를 그룹화하여 구조적으로 만들기 위해 사용
- <legend> : 폼 요소의 제목으로 <fieldset> 내부에 작성

<fieldset>은 보통 form의 성격에 따라 구분합니다.

<legend>는 <fieldset>의 자식으로 반드시 최상단에 위치해야 합니다.

<body>

<h1>Form 관련 요소</h1>

<fieldset>

<legend>기본 정보</legend>

<label for="userid">아이디:</label> <input type="text" placeholder="영문으로만 써주세요" id="userid">

<br>

<label for="userpw">비밀번호:</label> <input type="password" id="userpw"><br>

성별: <label for="male">남자</label> <input type="radio" name="gender" id="male" checked> <label for="female">여자</label> <input type="radio" name="gender" id="female"><br>

</fieldset>

### <form>

<form>은 form 요소들을 감싸는 태그로 데이터를 묶어서 실제 서버로 전송해주는 역할을 하는 태그입니다.

만약 <fieldset>으로 구조화되어있다면 <fieldset>도 함께 감싸는 역할을 합니다.

<form>에는 대표적인 2가지 속성이 있습니다.

action: 데이터를 처리하기 위한 서버의 주소

method: 데이터를 전송하는 방식을 지정

method 속성값에는 get/post 2가지 방식이 존재합니다.

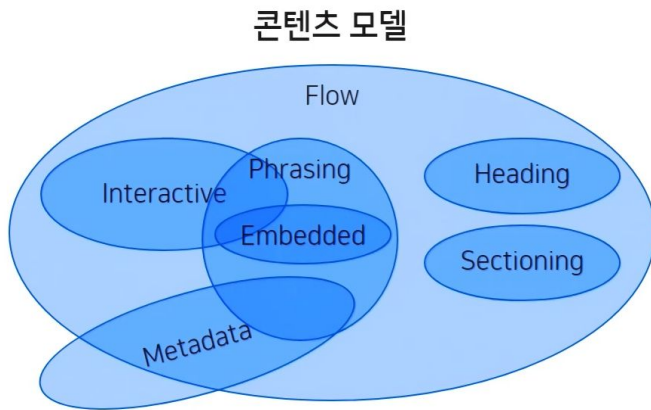
get 방식은 데이터가 전송될 때 주소창에 파라미터 형태로 붙어 데이터가 노출됩니다.

반면, post 방식은 데이터가 전송될 때 데이터가 노출되지 않습니다.

<form action="/result.html" method="post || get">

### 3. 콘텐츠모델, 시멘틱마크업, 블록 & 인라인

#### 1) 콘텐츠 모델



##### 1. Metadata

" base, link, meta, noscript, script, style, title "

Metadata에는 콘텐츠의 스타일, 동작을 설정하거나 다른 문서와의 관계 등 정보를 포함하는 요소들이 포함됩니다. 메타 태그, 타이틀 태그, 스타일 태그, 링크 태그가 이에 해당하며 대부분 **<head>**내에 들어간다는 것이 특징입니다.

##### 2. Flow

Flow에는 문서의 자연스러운 흐름에 의해 배치되는 요소들이 포함됩니다.

Metadata에 해당하는 일부 태그들만 Flow에서 제외되며 요소 대부분이 Flow에 포함됩니다.

##### 3. Sectioning

" article, aside, nav, section "

Sectioning에는 문서의 구조와 관련된 요소들이 포함됩니다.

HTML5에서 새로 생긴 <article>, <aside>, <nav>, <section> 등이 포함되며 이 태그들은 문서의 구조, 아웃라인에 영향을 주게 됩니다.

##### 4. Heading

" h1, h2, h3, h4, h5, h6 "

Heading에는 각 section의 header를 정의하는 heading 태그가 포함됩니다.

##### 5. Phrasing

"a, abbr, map>area, audio, b, bdo, br, button, canvas, cite, code, datalist, del, dfn, em, embed, i, iframe, img, input, ins, kbd, keygen, label, map, mark, math, meter, noscript, object, output, progress, q, ruby, samp, script, select, small, span, strong, sub, sup, svg, textarea, time, var, video, wbr"

Phrasing에는 문서의 텍스트 또는 텍스트를 꾸며주는 문단 내부 레벨로 사용되는 요소들이 포함됩니다.

##### 6. Embedded

" audio, canvas, embed, iframe, img, math, object, svg, video "

Embedded에는 외부 콘텐츠를 표현하는 요소들이 포함되며 오디오나 비디오, 이미지 등 멀티미디어 관련 요소들이 이에 해당합니다.

##### 7. Interactive

" a, audio[controls], button, details, embed, iframe, img[usemap], input, keygen, label, menu, object[usemap], select, textarea, video[controls] "

Interactive에는 사용자와 상호작용을 하는 요소들이 포함되며 대표적으로 form 요소들이 이에 해당합니다.

## 2) 시멘틱 마크업

### 시멘틱 마크업이란?

시멘틱은 즉, 기계(컴퓨터, 브라우저)가 잘 이해할 수 있도록 하는 것을 뜻합니다.

애초에 프로그래밍 언어는 사람과 기계와의 정해진 약속이며 HTML 역시 마찬가지입니다.

시멘틱 마크업은 적절한 HTML 요소를 올바르게 사용하는 것에서 시작합니다.

### 시멘틱 마크업 하기

구체적으로 설명하자면 마크업 할 때는 의미에 맞는 태그, 요소를 사용하는 것이고 문서를 표현할 때는 구조화를 잘 해주는 것입니다. 정해진 약속대로 코드를 작성하게 되면 결국 기계뿐 아니라 사람도 이해하기 쉬운 코드가 됩니다.

<b>굵은</b> vs <strong>중요한</strong>

<i>기울어진</i> vs <em>강조하는</em>

<u>밑줄친</u> vs <ins>새롭게 추가된</ins>

<s>중간선이 있는</s> vs <del>삭제된</del>

## 3) HTML5 시멘틱 요소

- <article>
- <aside>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <time>

<https://developer.mozilla.org/en-US/docs/Glossary/Semantics>

## 4) 블록&인라인

### 블록 레벨 요소

부모 요소의 가로 영역에 맞게 꽉 채워져 표현되는 요소입니다.

양옆으로 다른 요소가 배치되지 않게 박스를 생성하므로 박스의 위아래로 줄 바꿈이 생기게 됩니다.

블록 레벨 요소는 일반적인 모든 요소(블록, 인라인 레벨 등)를 포함할 수 있습니다.

" div, h1~h6, p, ul, li, table ..."

예외

<h1> ~ <h6>(headings) 요소와 <p> 요소는 블록 레벨 요소지만, 내부 요소로 Phrasing Content만 허용합니다.

### 인라인 레벨 요소

하나의 라인 안에서 자신의 내용만큼의 박스를 만드는 요소입니다.

라인의 흐름을 끊지 않고 요소 앞 뒤로도 줄 바꿈이 되지 않아 다른 인라인 요소들이 자리할 수 있습니다.

인라인 레벨 요소는 블록 레벨 요소의 자식으로 분류되기 때문에 **자손으로 블록 레벨 요소를 가질 수 없습니다.**

즉, 인라인 레벨 요소는 블록 레벨 요소를 포함할 수 없습니다.

" span, i, img, em, strong, a ..."

다만, HTML5 버전에서 생겨난 한가지 예외 경우가 있는데 <a>는 인라인 레벨 요소지만 자손으로 블록 레벨 요소를 가질 수 있습니다.

