

1. 웹 접근성과 웹표준

웹 접근성의 개요

“모든 사용자가 신체적, 환경적 조건에 관계없이
웹에 접근하여 이용할 수 있도록 보장하는 것”

웹 접근성(Web Accessibility)

웹 사이트에서 제공하는 정보를 차별 및 제한 없이
동등하게 이용할 수 있도록 보장하는 것

장애인 및 고령자 등
모든 사람

다양한 Platform 및 Device와
웹 브라우저 등 모든 환경

웹 콘텐츠 접근성 지침(WCAG)

콘텐츠 접근성 지침(WCAG 2.0)

인지성(Perceivable)

운용성(Operable)

이해성(Understandable)

내구성(Robust)

웹 표준 관련 기술의 소개

웹표준 관련 기술

구조
(x)HTML

표현
CSS

동작
Script

구조언어 (HTML)

- HTML은 'HyperText Markup Language'의 약자로 웹 문서의 구조를 정의할 때 사용합니다. 제목, 본문, 목록, 링크, 이미지 등의 다양한 콘텐츠를 의미있게 마크업 할 수 있습니다.

사용 예

```
<html>
  <head>
    <title>문서 제목 영역</title>
    <meta http-equiv="content-type" content="text/html; charset=euc-kr">
  </head>
  <body>
    <h1>본문 제목 영역</h1>
    <p>본문 내용 영역 1
    <p>본문 내용 영역 2
  </body>
</html>
```

구조언어 (XML)

- XML(eXtensible Markup Language)은 1996년 W3C(World Wide Web Consortium)에서 제안한 것으로, 웹에서 구조화된 문서를 전송할 수 있도록 설계된 표준화된 텍스트 형식입니다.
- XML은 인터넷에서 기존에 사용하던 HTML의 한계를 극복하고 SGML의 복잡함을 해결하는 방안으로 HTML에 담겨져 있는 형식적 요소를 완전히 배제하는 방식입니다.

사용 예

```
<?xml version="1.0" encoding="euc-kr" ?>
```

```
<일기장>
```

```
<작성자>홍길동</작성자>
```

```
<날짜>7월25일</날짜>
```

```
<제목>웹표준</제목>
```

```
<본문>웹표준은 W3C에서 정한 기술사양을 말합니다.</본문>
```

```
</일기장>
```

구조언어 (XHTML)

- W3C에서는 XHTML(eXtensible Hypertext Markup Language)을 'XML 응용으로서의 HTML4를 다시 공식화한 것'이라고 정의하고 있습니다.
- W3C는 좀 더 원활하게 기계적으로 처리하도록 XML의 형식을 빌어 HTML 4.01를 재정의하게 되는게 이것이 바로 XHTML 입니다.

사용 예

```
<html>
  <head>
    <title>문서 제목 영역</title>
    <meta http-equiv="content-type" content="text/html; charset=euc-kr" />
  </head>
  <body>
    <h1>본문 제목 영역</h1>
    <p>본문 내용 영역 1</p>
    <p>본문 내용 영역 2</p>
  </body>
</html>
```

구조언어 (HTML5)

- HTML5는 HTML의 다음 버전으로 HTML 4를 업그레이드한 것입니다. HTML5는 특정 플러그 인에 의존하지 않고 콘텐츠를 제공하는 것이 목표
- 많은 기업들이HTML5의 표준화에 힘을 보태고 있고, Firefox, Opera, Safari, Chrome 등 최신의 웹 브라우저에서 기본적으로 HTML5를 지원하고 있으며 MS 또한 앞으로 인터넷 익스플로러에서 HTML5를 지원하겠다고 발표한 상태

사용 예

```
<!doctype html>
<html>
  <head>
    <title>문서 제목 영역 </title>
  </head>
  <body>
    <header> 헤더 영역 </header>
    <nav> 내비게이션 영역</nav>
    <section>
      <article> 섹션 영역 </article>
    </section>
    <figure>
      <video src="URI"...</video>
      <legend>캡션</legend>
    </figure>
    <footer> 푸터 영역 </footer>
  </body>
</html>
```

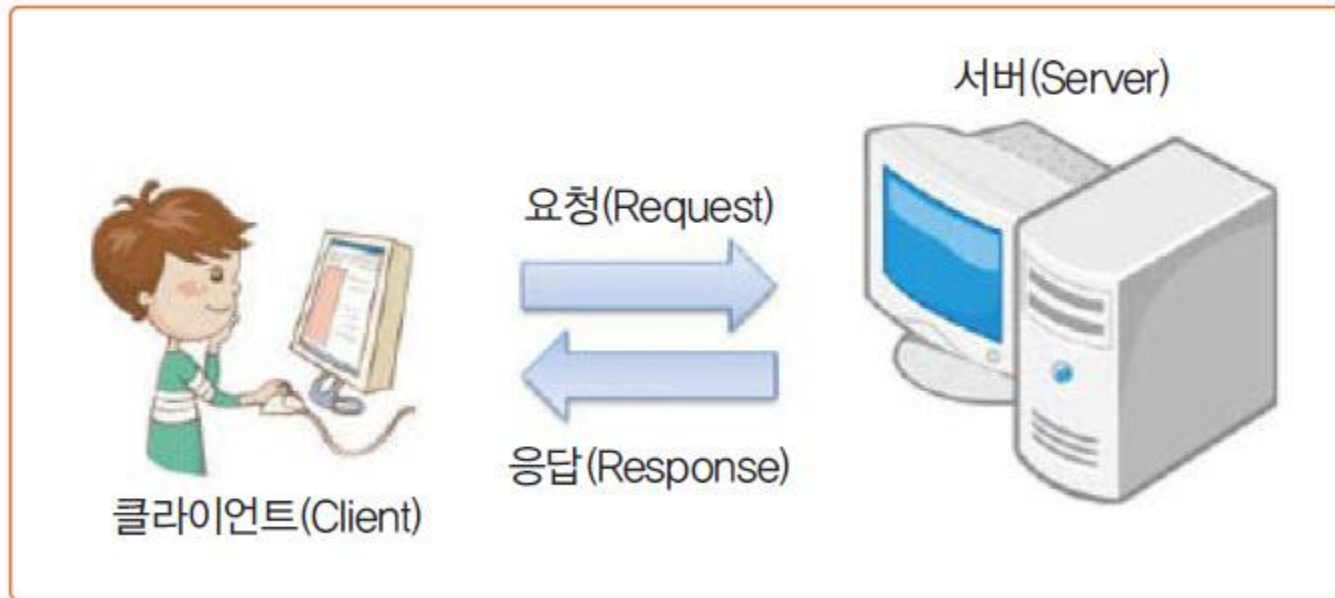
표현언어 (CSS)

- CSS(Cascading Style Sheet)는 HTML 3.2부터 지원하기 시작한 것으로, 웹 디자이너와 사용자들의 필요에 의해 특별히 개발되었습니다.
- CSS에서는 폰트, 색상, 공백, 공간과 그 밖의 문서 표현 측면을 자유롭게 지정할 수 있는 기능을 제공합니다.
- 기존의 HTML은 웹 문서를 다양하게 설계하고 수시로 변경하는 데 많은 제약이 있었기 때문에 이것을 보완하기 위해 CSS를 만든 것입니다.

동작 및 제어 언어(DOM & ECMA Script)

- 웹 페이지의 요소를 객체화해서 동작을 제어하기 위해 사용하는 웹 표준 기술에는 DOM과 ECMA Script가 있습니다.
- DOM(Document Object Model)은 웹 페이지의 구성 체계를 말하는데 이것은 HTML을 작성하면서 생성되는 논리적 규칙입니다.
- 웹 브라우저는 이 논리적 구성 체계인 DOM을 해석하여 페이지를 표시하거나 ECMA Script 등의 기술을 통하여 DOM의 구조를 변경할 수 있습니다.

웹(WWW) 서비스의 이해



웹의 동작 방식은 서버(server)와 클라이언트(client)의 형태를 띄며 클라이언트가 웹 브라우저를 통해 특정 콘텐츠를 요청하면 서버는 클라이언트의 요청에 응답하는 형식으로 이루어져 있습니다.

웹 프로그래밍의 이해

클라이언트측(client side)	서버측(server side)
(x)HTML, CSS, DOM, ECMA Script, XML 등	ASP, JSP, PHP 등

- 웹 프로그램은 수행되는 위치에 따라 클라이언트측과 서버측으로 분류할 수 있으며, 컴파일의 여부에 따라 '컴파일 방식'과 '비컴파일 방식(Script 포함)'으로 나눌 수 있습니다.
- 클라이언트가 요청한 파일이 서버의 우선 처리 없이 클라이언트에 도착한 후 웹 브라우저로 수행되면 '클라이언트측'으로 분류하고 서버에서 우선 처리한 후 클라이언트에서 수행되면 '서버측'으로 분류합니다.

2. 구조설계 (XHTML)

XHTML 서식

`<P>` -(HTML)

`<p></p>` -(XHTML)

1. 요소 사용 시 종료 태그의 생략 불가능

- HTML의 경우 P, TR, TH, TD, LI 등 요소명을 사용할 때 종료 태그를 생략할 수 있지만, XHTML에서는 이를 허용하지 않습니다. 따라서 모든 요소는 반드시 시작과 함께 종료를 표현(한 칸의 공백과 /)해야 합니다.

XHTML 서식

```
<P><IMG SRC="images/back.gif" alt="뒤로"></P> -(HTML)  
<p></p> -(XHTML)
```

2. 요소명과 속성 명에 소문자 사용

- HTML의 경우에는 요소명과 속성명에 대소문자를 구분하지 않고 사용했지만, XHTML에서는 모든 요소명과 속성명에 소문자만 사용할 수 있습니다.



XHTML 서식

`<P></P> -(HTML)`

`<p></p> -(XHTML)`

3. 빈 요소 사용 시 <요소명 /> 형식으로 기술

- HTML의 경우 빈 요소는 시작 태그만 기술해도 되지만, XHTML에서는 모든 요소에 종료의 표현을 기술해야 합니다. 따라서 빈 요소는 HTML과 XHTML이 다르게 표현합니다.

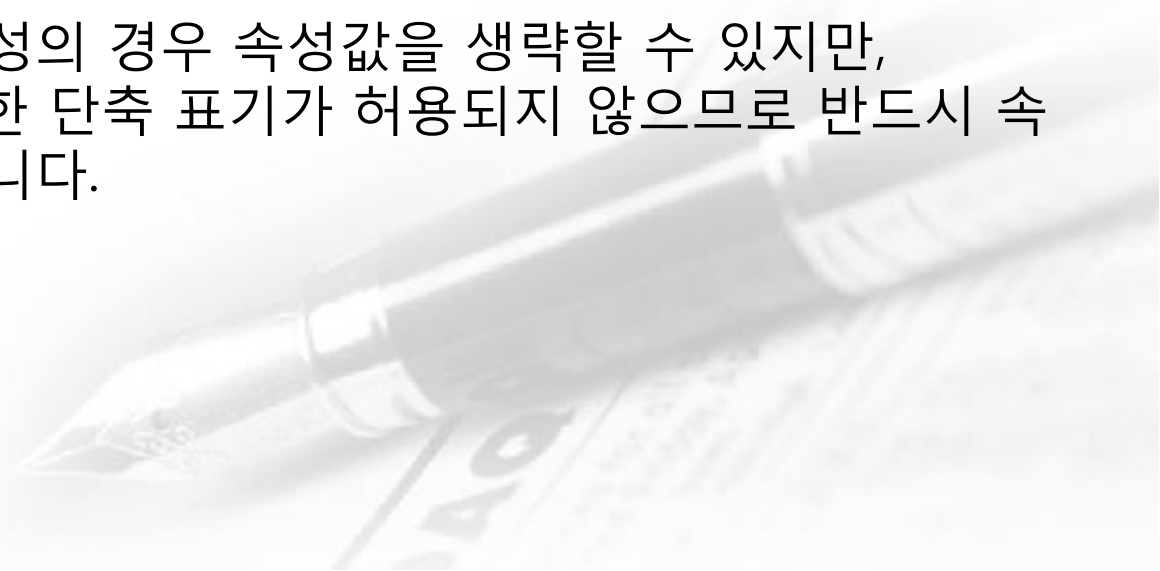
XHTML 서식

`<P <SELECT selected>...</SELECT> -(HTML)`

`<p: <select selected=" selected" > ...</select> -(XHTML)`

4. 속성에 속성 값 생략 불가능

- HTML은 몇 가지 속성의 경우 속성값을 생략할 수 있지만, XHTML에서는 이러한 단축 표기가 허용되지 않으므로 반드시 속성값을 지정해야 합니다.



XHTML 서식

`<P>...</P>` -(HTML)

`<p>...</p>` -(XHTML)

5. 잘못된 중첩의 사용 불가능

- HTML은 규칙을 느슨하게 적용하여 중첩을 잘못해도 문제가 발생하지 않는 경우가 있었지만 XHTML에서는 잘못된 중첩이 허용되지 않습니다.

XHTML 서식

```
<IMG SRC=cake.jpg alt=케이크>  

```

6. 모든 속성값에 인용 부호 사용

- HTML의 경우 속성에 값을 지정할 때 인용 부호를 생략할 수 있지만 XHTML의 경우에는 속성값은 시작과 끝을 반드시 인용 부호로 감싸야 합니다.



XHTML 서식

<h1>웹표준 & 웹 접근성</h1>

<h1>웹표준 & 웹 접근성</h1>----- Characters Entity Name

<h1>웹표준 & 웹 접근성</h1>----- Characters Entity Code

7. <, >, & 을 <, >, &로 변환하여 사용

- HTML과 XHTML 모두 문자 참조(특수 문자)의 경우 Characters Entity Name이나 Characters Entity Code로 변환해야 합니다.
- 다음의 예와 같이 &의 문자 참조를 Entity Name으로 지정할 경우에는 &로, Entity Code로 지정할 경우에는 &로 지정합니다.

XHTML 서식

<!-- 주석 삽입 -->

<!-- — 잘못된 사용 방법. 주석의 처음과 끝에만 ? 사용 가능하다. — ——>

<!-- ***** 유효한 주석 표현 방법 ***** -->

<!-- ————— 유효한 주석 표현 방법 ————— -->

8. 주석은 바르게 사용하고 그 안에 --(더블대시) 사용 불가
- HTML과 XHTML에서 사용하는 올바른 주석은 <!-- 주석 --> 입니다. 특히 해당 주석에 영역을 구분할 목적으로 더블대시를 반복해서 사용하지 않아야 합니다.

XHTML 구성요소



1. 태그(tag)

- 태그는 <와 >로 묶어서 표현하는 명령어
- 기본 형식은 `<tag>~</tag>`이며 empty element(빈 요소)의 경우 `<tag/>` 형태로 기술

XHTML 구성요소

2. 요소(element)

시작 <tag>~종료 </tag>까지의 모든 명령어 집합을 '요소(element)'라고 합니다. XHTML 문서에서 콘텐츠는 이러한 요소들로 구성되며 각 요소의 의미에 따라 콘텐츠의 제목이나 본문의 구조를 가집니다.

3. 속성(attribute)

시작 태그는 태그의 의미와 필요에 따라 개별적인 옵션을 가질 수 있는데, 이러한 옵션을 '속성(attribute)'이라고 합니다. 속성은 태그마다 다를 수 있으며, 여러 개의 속성을 하나의 태그에 지정할 때는 공백으로 구분하여 시작 태그에 지정할 수 있습니다.

4. 값(value)

각 속성이 가지는 값을 의미하며, 속성에 값을 할당할 때는 대입 연산자인 =과 함께 지정합니다.

5. 모든 태그에 사용할 수 있는 속성

id: 스크립트에서 구별하기 위한 용도로 사용되는 것으로 중복될 수 없습니다.

class: 스타일시트에서 동일한 그룹에 동일한 디자인을 적용하기 위한 속성

name: 전송할 때 전송 받은 쪽에서 사용할 이름으로 주로 form 태그 안의 요소들에 사용합니다.

XHTML 문서의 기본구조

```

<!DOCTYPE >----- DTD 선언
<html>----- xhtml 문서의 시작
    <head>..... </head>----- head 영역
    <body>..... </body>----- body 영역
</html>----- xhtml 문서의 종료

```

1. DTD 선언

웹 페이지를 제작할 때 사용할 문서형 정의를 선언합니다. 문서형 정의의 종류는 엄격형(strict)과 호환형(transitional), 프레임형(frameset)

XHTML 문서의 기본구조

2. <html> ~ </html>

모든 웹 페이지의 콘텐츠는 <html> 태그와 </html> 태그 안에 선언해야 하며, <html>~</html> 태그는 웹 페이지의 시작과 종료를 의미

3. <head> ~ </head>

➤html 문서는 머리 부분과 본문 부분으로 나눌 수 있는데, 머리 영역의 시작은 <head> 태그로 종료는 </head> 태그로 선언

➤<head> 태그 안에는 문서의 일반적인 정보와 title 등의 제목을 선언

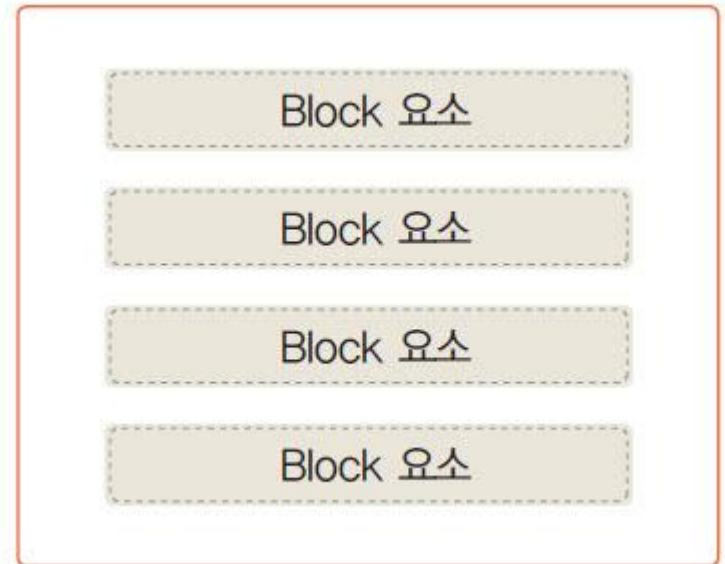
4. <body> ~ </body>

body 요소는 html 문서의 본문 부분에 해당하며 웹 브라우저 화면에 나타나는 모든 콘텐츠는 <body> 태그와 </body> 태그 사이에 선언해야 합니다.

블록 요소와 인라인 요소

1. 블록 요소

블록 요소(block element)는 독립적인 형태의 상자



블록 요소는 제목(<h1>~<h6>), 문단(<p> 요소), div, ul, ol, dl 등이 있으며 대부분의 블록 요소는 인라인 요소와 텍스트를 포함할 수 있고 일부는 또 다른 블록 레벨 요소를 포함시킬 수 있습니다. 블록 요소와 블록 요소 사이에는 기본적으로 여백이 있는데 이 크기는 웹 브라우저마다 약간씩 다를 수 있습니다.

블록 요소와 인라인 요소

2. 인라인 요소

인라인 요소(inline element)는 '블록 상자 안의 일부'라는 의미로, 링크(a 요소), 이미지(img 요소), 강조(em, strong 요소) 등이 해당됩니다.

인라인 요소 안에는 인라인 요소와 텍스트를 포함할 수 있지만 블록 요소를 포함할 수는 없습니다.



인라인 요소와 텍스트는 반드시 블록 레벨 요소에 포함시켜 나타내야만 합니다. 즉 BODY 요소의 바로 하위에는 블록 레벨 요소만 자식 요소로 올 수 있고, 인라인 요소를 BODY 요소의 직접적인 하위 내용으로 하는 것은 옳지 않습니다.