

HTML



+

CSS



CHAPTER_2

HTML 문법

▶ HTML 문서

HTML 문서 기초

요소와 태그

태그 중첩

빈 요소

특성

불리언 특성

▶ XHTML5

여러 버전을 지원하는 HTML 문서

XHTML5 정의

▶ 문서 타입과 구조

MIME 타입

일반적인 MIME 타입

문서 객체 모델

의미적 요소

▶ 문법 요약

일반 문법 규칙

(X)HTML5 빈 태그 문법

선택적 태그를 사용하는 HTML5 요소

요소의 타입

브라우저 호환성을 위한 스크립팅

Modernizr

HTML5를 활성화하는 스크립트

- Polyfills

HTML 문서

HTML 문서 기초

일반적으로 HTML 문서는 독립적인 HTML 파일로 표현된다.

- HTML 문서는 **요소**와 텍스트가 혼합된 코드로 구성된다.
- 요소는 기본적인 HTML 빌딩 블록^{Buiding Block}이며, HTML 태그에 의해 표현된다.
- 요소는 계층적이며, 중첩된 구조로 이루어진다.

요소와 태그

```
<!DOCTYPE html>
<html>
<head>
<title>HTML5 레퍼런스</title>
<!-- 이 부분은 주석이며, 브라우저에 렌더링되지 않는다. -->
</head>
<body>
<h1>둘러보기</h1>
<p>HTML은 HyperText Markup Language의 줄임말이다.</p>
</body>
</html>
```

각 요소는 `<title>`처럼 여는 태그와 `</title>`처럼 그에 대응하는 닫는 태그로 구성된다. 몇몇 태그는 닫는 태그가 필요치 않다.

요소는 HTML 태그의 개념적인 표현으로 특성과 자식 태그를 포함하고 있다는 점에서 태그와 다르다. 요소에 대한 자세한 내용은 제3장과 제4장을 참고하기 바란다.

태그 중첩

이것은 `잘못된` 중첩입니다.

이것은 `올바른` 중첩입니다.

- 태그는 불균등 기호로 사용되는 작다 기호(<)와 크다 기호(>)로 구성된다.
- 중첩된 태그 쌍은 반드시 일치해야 하며, 도중에 다른 태그 쌍이 나타나서는 안 된다.

빈 요소

```
<!--빈 요소!-->
<img src=logo.gif><br />
<input type=text>
```

빈 요소는 여는 태그만 존재하기 때문에 다른 콘텐츠를 가질 수 없다. 종료를 의미하는 슬래시는 요소 태그의 마지막에 태그를 닫는 크다(>) 기호 바로 앞에 선택적으로 사용할 수 있다. 빈 요소가 아닌 요소에 대해 슬래시를 사용하면 안 된다.

특성

```
<input type=text autofocus name='first name'>
```

오른쪽 예제에서 `input` 요소는 `type`, `autofocus`, `name` 등 다양한 특성을 가질 수 있다.

- 특성은 요소의 속성이다.
- 특성은 여는 태그 내부에 위치한다.
- 태그 이름과 특성은 모두 공백으로 분리한다.
- 특성은 이름(`type`)과 값(`text`)으로 구성되며, 같대(=) 기호로 분리된다.
- 태그 내에 특성이 중복되어서는 안 된다.
- HTML5 특성 값은 공백, 따옴표 혹은 불균등 기호가 없는 경우 따옴표를 사용하지 않아도 무방하다. 그렇지 않다면 `'first name'`처럼 작은따옴표나 큰따옴표로 특성 값을 둘러싸야 한다.

불리언 특성

불리언 특성이란 false 혹은 true 값을 갖는 속성을 의미한다.

```
<!-- 불리언 특성 값이 false이며, 필드가 비활성화되지 않는다. -->
<input>
```

- 불리언 특성을 명시하지 않으면 해당 속성 값이 "false"임을 의미한다.

```
<!-- 불리언 특성 값이 true이다. -->
<input disabled=disabled>
<input disabled="disabled">
<input disabled="">
<input disabled>
```

- 불리언 특성이 존재한다는 것은 해당 속성의 값이 "true"임을 의미한다.
- 불리언 특성은 다음과 같이 특성의 이름을 값으로 가질 수 있다.
`<input disabled=disabled>`
- 여러 버전과 호환되는 HTML/XHTML 문서에서 true 값을 가지는 불리언 특성은 다음과 같이 특성 이름과 동일한 값을 따옴표로 표현한다.
`<input disabled="disabled">`
- 한 가지 버전에만 호환되는 HTML 문서에서 특성 값은 다음과 같이 생략 가능하다.
`<input disabled>`

XHTML5

여러 버전을 지원하는 HTML 문서

여러 버전을 지원하는 HTML 문서에서는 HTML 및 XHTML이 모두 유효하다.

```
<!-- HTML4, HTML5 문법 --><input disabled><input disabled=disabled>
<!-- XHTML 1.0 문법 --><input disabled="disabled" />
<!-- HTML4, HTML5, XHTML 1.0 혼합 문법 --><input disabled="disabled" />
```

- 여러 버전 HTML 문서는 HTML과 XHTML 문법의 교집합에 해당하는 HTML과 XHTML 문법 규칙을 따른다.
- 여러 버전 문서는 브라우저의 지원과 **MIME 타입**에 따라 HTML과 XHTML 형식 중 어떤 것으로도 사용이 가능하다.
- HTML과 여러 버전 문법 중 어떤 것을 선택할 것인지는 프로젝트의 목적과 브라우저의 지원 및 기타 여러 가지 사실을 토대로 선택해야 한다.

XHTML5 정의

여러 버전을 지원하는 **HTML5** 코드는 기본적으로 `[application/xhtml+xml]`이라는 XML MIME 타입을 통해 제공될 경우 XHTML5 문서가 된다. HTML5 여러 버전 문서를 간단히 요약해보면 다음과 같다.

- HTML5 DOCTYPE/네임스페이스. HTML5는 더는 SGML 형식을 토대로 하지 않으므로 더 이상 문서 타입 정의(DTD: Document Type Definition)를 참조할 필요가 없다. 그러나 하위 호환성을 위해 DOCTYPE 선언이 필요하다.
- XHTML 기반의 잘 정의된 문법을 사용해야 한다.

여러 버전을 지원하는 문서는 HTML 혹은 XHTML로 제공할 수 있으며, 이는 브라우저의 지원과 MIME 타입에 따라 달라진다. 여러 버전을 지원하는 HTML5 코드는 기본적으로 `application/xhtml+xml` MIME 타입을 사용할 경우 XHTML5 문서로 취급된다. XHTML5 문서에 대해 간략히 살펴보면 다음과 같다.

- 기본적으로 UTF-8 인코딩을 사용한다면 XML 선언 `<?xml version="1.0" encoding="UTF-8" ?>` 이 필요치 않다.
- HTML DOCTYPE: `<!DOCTYPE html>` 선언은 선택적이지만 여러 버전을 지원하는 문서가 HTML 혹은 XHTML 형식을 모두 지원할 필요가 있다면 사용해야 한다.
- XHTML 기반의 잘 정의된 문법을 사용해야 한다.
- XML MIME 타입: `application/xhtml+xml`을 사용한다. 이 MIME 타입은 소스 코드에서는 보이지 않지만 웹 서버의 설정에 따라 HTTP Content-Type 헤더에 이 MIME 타입이 나타날 수 있다. IE는 XHTML 문서를 렌더링할 수 있기는 하지만 이 XML MIME 타입은 현재 버전의 IE에서는 아직 지원되지 않는다.
- 기본 XHTML 네임스페이스: `<html xmlns="http://www.w3.org/1999/xhtml">`
- SVG, MathML, Xlink 등을 위한 2차 네임스페이스: 필자에게 있어 이는 마치 테스트 같은 것이다. 2차 네임스페이스가 필요하지 않다면 XHTML을 사용하는 것은 과분한 것이다. 그러나 기본적으로 HTML5와 XHTML5의 선택은 단순히 미디어 타입을 선택하는 문제가 아니다.

마지막으로 기본적인 XHTML5 문서는 다음과 같다.

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title></title>
  <meta charset="UTF-8" />
</head>
<body>
  <svg xmlns="http://www.w3.org/2000/svg">
    <rect stroke="black" fill="blue" x="45px" y="45px" width="200px"
    height="100px" stroke-width="2" />
  </svg>
</body>
</html>
```

XML 선언 `<?xml version="1.0" encoding="UTF-8"?>`은 기본 인코딩으로 UTF-8을 사용한다면 필요치 않다. XHTML5 검증기는 이 선언이 생략되어 있어도 문제 삼지 않는다.

그러나 서버에서 HTTP Content-Type 헤더를 사용하여 인코딩 방식을 설정하거나 문서 내에 메타 태그를 통해 `<meta charset="UTF-8" />`과 같이 추가할 것을 강력하게 권장한다.

이 인코딩 선언은 문서가 여러 버전을 지원하는 경우 문서 자체가 HTML인지 XHTML인지에 관계없이 UTF-8 인코딩을 사용하도록 하기 위해 필요하다.

파이어폭스 플러그인 및 데스크톱 애플리케이션 형태로 제공하는 Total Validator 도구는 XHTML5 명세의 유효성 검사에 대해 사용자가 직접 선택할 수 있는 옵션들을 제공한다.

XHTML5를 사용할 때의 주요 이점은 HTML5의 특징을 SVG와 MathML과 같은 XML을 기반으로 한 기술로 확장할 수 있다는 점이다. 비록 HTML5 명세에 따라 SVG와 MathML을 인라인 형태로 사용할 수는 있지만 현재 브라우저의 지원은 제한적이다. 단점은 IE가 아직 지원하지 않는다는 점과 코드가 뒤섞여 알아보기 어렵다는 점, 오류 처리가 되지 않는다는 점 등이다. 이런 확장 기능이 필요치 않다면 HTML5를 사용하는 것이 좋다.

HTML5와 XHTML5 사이의 선택은 결국 여러분이 사용하는 문서가 어떤 문서인지를 결정하는 MIME 타입이나 콘텐츠의 타입을 선택하는 것이다. XHTML1과 HTML4 사이에서의 선택의 문제와 달리, XHTML5와 HTML5 사이의 선택은 DOCTYPE이 아니라 MEMIE 타입을 통해서만 가능하다.

문서 타입과 구조

MIME 타입

“MIME”은 “Multiple Internal Mail Extensions”의 약자이다. MIME 타입은 **인터넷 미디어 타입**^{Internet Media Type} 혹은 **콘텐츠 타입**^{Content Type}이라고도 한다. 이는 정보의 타입을 식별하기 위한 파일 확장자와 유사하며, 최소한 타입과 서브 타입 등 두 가지 컴포넌트로 구성되어야 하고, 필요한 경우 몇 가지 선택적 매개변수를 사용할 수도 있다.

```
<!DOCTYPE html>
  <head>
    <title>HTML5</title>
    <link media=screen type=text/css href=styles.css
      rel=stylesheet>
  </head>
  <body></body>
</html>
```


일반적인 MIME 타입

타입	콘텐츠 타입/서브 타입 코드	설명
애플리케이션	<code>application/javascript</code>	자바스크립트
	<code>application/xhtml+xml</code>	XHTML
오디오	<code>audio/mpeg</code>	MPEG, MP3 오디오
	<code>audio/x-ms-wma</code>	Windows Media Audio 형식
	<code>audio/vnd.rn-realaudio</code>	RealAudio 오디오 형식
이미지	<code>image/gif</code>	GIF 이미지
	<code>image/jpeg</code>	JPEG 이미지
	<code>image/png</code>	Portable Network Graphics 이미지
	<code>image/svg+xml</code>	SVG 벡터 그래픽
메시지	<code>message/http</code>	메시지
텍스트	<code>text/css</code>	스타일 시트
	<code>text/csv</code>	콤마로 구분된 값들
	<code>text/html</code>	HTML
	<code>text/javascript</code>	자바스크립트. 이 형식은 더는 사용을 권하지 않으나 HTML5에서는 여전히 허용된다. <code>application/javascript</code> 와 달리 크로스 브라우저를 지원한다.
	<code>text/plain</code>	기본 텍스트
	<code>text/xml</code>	XML 형식
비디오	<code>video/mpeg</code>	MPEG-1 비디오
	<code>video/mp4</code>	MP4 비디오
	<code>video/quicktime</code>	애플 QuickTime 비디오
	<code>video/x-ms-wmv</code>	Windows Media Video 형식

주요 미디어 타입에 대해서는 IANA(Internet Assigned Numbers Authority, <http://iana.org/assignments/media-types>)에서 찾아볼 수 있다.

문서 객체 모델

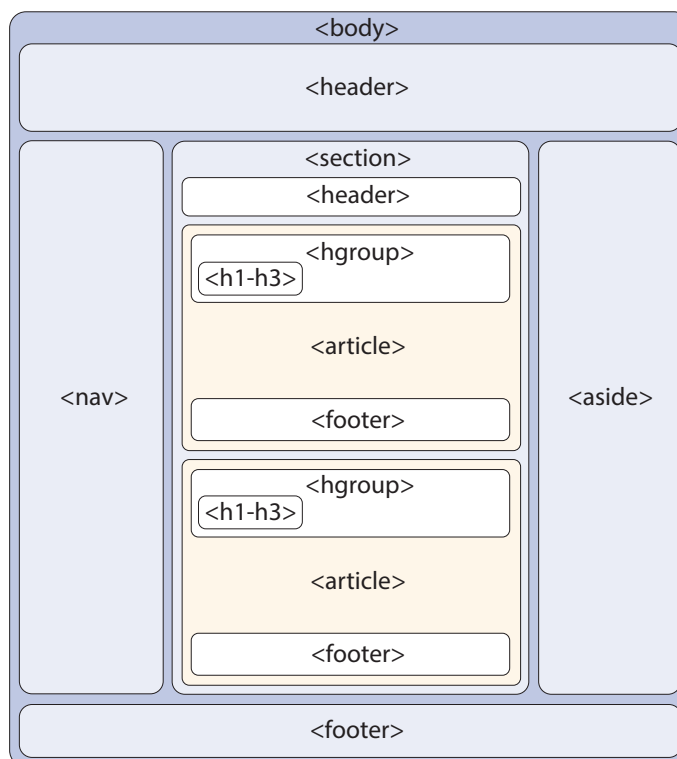
웹 브라우저는 요소, 특성 및 텍스트 등과 같은 객체의 트리 형태로 표현되는 HTML 문서를 토대로 모델을 구축한다. 이 모델은 **문서 객체 모델** DOM: Document Object Model이라고 부른다. DOM 객체는 자바스크립트에 의해 조작이 가능하다. 계층적 DOM 트리의 객체 인스턴스를 **노드** Node라고 부른다.

```
<div onclick="document.getElementById('description').style.display = 'none';">
  설명 숨기기</div>

<div id=description>둘러보기</div>
```

의미적 요소

(X)HTML5는 새로운 **블록 수준** block-level과 **인라인 수준** inline-level 요소들을 제공한다. 새로운 HTML5의 블록 수준과 요소들은 웹 페이지의 의미적 구조를 형성한다.



요소	주요 콘텐츠	주소 부모 객체	자식 객체
<header>	제목, 로고, 배너, 산업 정보 등	Body, Section, Article	Nav, Section
<hgroup>	h1부터 h6까지의 제목	Article, Header	h1, h2, h3, h4, h5, h6
<nav>	주 내비게이션 메뉴	Body	Section, Nav
<section>	일반적인 페이지 섹션	Body	Article, Header, Footer, Aside, Nav
<article>	문서, 하위 섹션, 블로그 포스트 등	Body, Section	Section, Header, Footer
<address>		Footer	p, a
<aside>	사이드바 콘텐츠, 팁, 인용문	Body	Section, Article
<footer>	꼬리글, 요약, 저작권 정보, 2차 내비게이션 등	Body, Section, Article	Nav, Section

다음에 나열하는 요소들은 **인라인 수준** 의미적 요소들이다.

, <s>, <ins>, , <small>, , <cite>, <i>, <q>, , <code>, <mark>

문법 요약

일반 문법 규칙

규칙	HTML5 문법	XHTML5 문법
XML 선언	N/A	필수 아님 UTF-8 인코딩 사용 시 생략 가능 <?xml version="1.0" encoding="UTF-8"?>
DOCTYPE	필수	필수 아님
MIME 타입	text/html	application/xhtml+xml
특성 값 대소문자 구분	필수 아님	필수
특성 값 내 공백 일반화	공백이 일반화되지 않음	몇몇 예외를 제외하고 공백은 하나의 공백 문자로 일반화됨
특성에 소문자 사용	필수 아님	필수
특성에 따옴표 사용	특성 값에 공백 및 " ' ' = < > 등이 사용되지 않을 경우 필수 아님	필수
	 	

규칙	HTML5 문법	XHTML5 문법
볼리언 특성	필수 아님. 특성 값 최소화가 허용됨 <pre><input disabled=disabled> <input disabled="disabled"> <input disabled=""> <input disabled></pre>	필수. 특성 값 최소화는 허용하지 않음 <pre><input disabled="disabled" /></pre>
빈 요소에 대한 종료 슬래시	필수 아님 <pre>
 혹은
 <hr> 혹은 <hr /> 혹은 <input> 혹은 <input /></pre>	필수 <pre>
 <hr /> <input /></pre>
요소 이름	대소문자 구분 없음	대소문자 구분, 소문자만 허용
빈 요소에 대한 열기 및 닫기 태그	선택적. 특정 상태를 가진 일부 요소의 경우는 다음에 보여지는 표를 참조할 것	필수
예외처리 되지 않은 특수 문자	앰퍼센드(&)와 작다 기호(<)는 통상 요소 및 특성 값 내부에서 사용 가능. 몇 가지 예외가 적용됨	앰퍼센드(&)와 작다 기호(<)를 요소 및 특성 값에서 사용할 수 없으며, 반드시 & 및 <로 변환해서 사용해야 함

(X)HTML5 빈 태그 문법

- 빈 태그는 닫는 태그가 없다: `<input><input>`
- HTML5 — 빈 요소에 대한 종료 슬래시가 필요하지 않다: `<input>`
- XHTML5 — 빈 요소에 대한 종료 슬래시가 필요하다: `<input />`

빈 태그 목록

area	command	img	meta
base	embed	input	param
br	hr	link	source
col			

선택적 태그를 사용하는 HTML5 요소

- XHTML5 문서에서는 선택적 태그를 사용할 수 없다.
- 조건 열에서는 태그가 선택적으로 사용될 수 있는 조건을 표시한다.
 - 통상 태그를 생략하기 위해서는 최소 하나의 조건이 일치해야 한다.
 - 일부 태그는 현재 시점에 사용 가능한 조건 정보가 존재하지 않는다.

요소	시작 태그	조건	종료 태그	조건
<code><body></code> <code></body></code>	선택적	<ul style="list-style-type: none"> ■ Body 태그가 비어있는 경우 ■ 공백 문자 뒤에 나타나지 않는 경우 ■ 주석 요소 다음에 나타나지 않는 경우 ■ 다음 요소가 script나 style인 경우는 제외 	선택적	<ul style="list-style-type: none"> ■ 주석 요소 다음에 나타나지 않는 경우
<code><colgroup></code> <code></colgroup></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ■ 공백 문자 다음에 나타나지 않는 경우
<code><dd></code> <code></dd></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ■ dt나 dd 요소 다음에 나타나지 않는 경우
<code><dt></code> <code></dt></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ■ dt나 dd 요소 다음에 나타나지 않는 경우 ■ 그룹 내 마지막 요소인 경우
<code><head></code> <code></head></code>	선택적	<ul style="list-style-type: none"> ■ head 태그가 비어있는 경우 또는 ■ 다른 요소 다음에 나타나지 않는 경우 	선택적	<ul style="list-style-type: none"> ■ 공백 문자 다음에 나타나지 않는 경우 ■ 주석 요소 다음에 나타나지 않는 경우
<code><html></code> <code></html></code>	선택적	<ul style="list-style-type: none"> ■ 주석 요소 다음에 나타나지 않는 경우 	선택적	<ul style="list-style-type: none"> ■ 주석 요소 다음에 나타나지 않는 경우
<code></code> <code></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ■ 다른 li 요소 다음에 나타나는 경우 ■ 부모 요소 내의 요소 중 마지막 아이탬인 경우
<code><optgroup></code> <code></optgroup></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ■ 마지막 optgroup 요소인 경우 ■ 부모 요소 내의 마지막 요소인 경우
<code><option></code> <code></option></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ■ 다른 option 요소 다음에 나타나는 경우 ■ optgroup 요소 다음에 나타나는 경우 ■ 부모 요소 내의 마지막 요소인 경우
<code><p></code> <code></p></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ■ address, article, aside, blockquote, dir, div, dl, fieldset, footer, form, h1, h2, h3, h4, h5, h6, header, hgroup, hr, menu, nav, ol, p, pre, section, table 또는 ul 태그 다음에 나타나는 경우 ■ 부모 요소 내의 마지막 요소인 경우
<code><tbody></code> <code></tbody></code>	선택적	N/A	선택적	<ul style="list-style-type: none"> ■ tbody 요소 다음에 나타나는 경우 ■ tfoot 요소 다음에 나타나는 경우 ■ 부모 요소 내의 마지막 요소인 경우

요소	시작 태그	조건	종료 태그	조건
<code><td></code> <code></td></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ▪ <code>td</code> 요소 다음에 나타나는 경우 ▪ <code>th</code> 요소 다음에 나타나는 경우 ▪ 부모 요소 내의 마지막 요소인 경우
<code><tfoot></code> <code></tfoot></code>	선택적	N/A	선택적	<ul style="list-style-type: none"> ▪ <code>tbody</code> 요소 다음에 나타나는 경우 ▪ 부모 요소 내의 마지막 요소인 경우
<code><th></code> <code></th></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ▪ <code>td</code> 요소 다음에 나타나는 경우 ▪ <code>th</code> 요소 다음에 나타나는 경우 ▪ 부모 요소 내의 마지막 요소인 경우
<code><thead></code> <code></thead></code>	선택적	N/A	선택적	<ul style="list-style-type: none"> ▪ <code>tbody</code> 요소 다음에 나타나는 경우 ▪ <code>tfoot</code> 요소 다음에 나타나는 경우
<code><tr></code> <code></tr></code>	필수	N/A	선택적	<ul style="list-style-type: none"> ▪ <code>tr</code> 요소 다음에 나타나는 경우 ▪ 부모 요소 내의 마지막 요소인 경우

HTML5 문서 내에서 선택적 태그를 생략하는 유효한 경우와 그에 대항하는 소위 **잘 정의된** XHTML5 문서를 Total Validator 도구로 검사해보자.

HTML5

```

<!DOCTYPE html>

<title>HTML5 문서</title>

<table title=Report>
<tr>
  <td>첫 번째 칸
  <td>두 번째 칸
  <td>세 번째 칸
</tr>
<tr>
  <td>네 번째 칸
  <td>다섯 번째 칸
  <td>여섯 번째 칸
</tr>
</table>

```

XHTML5

```

<html xmlns="http://www.w3.org/1999/
xhtml">
<head>
  <title>XHTML5 문서</title>
</head>
<body>
  <table title="Report">
    <tr>
      <td>첫 번째 칸</td>
      <td>두 번째 칸</td>
      <td>세 번째 칸</td>
    </tr>
    <tr>
      <td>네 번째 칸</td>
      <td>다섯 번째 칸</td>
      <td>여섯 번째 칸</td>
    </tr>
  </table>
</body>
</html>

```

요소의 타입

타입	요소
루트	<ul style="list-style-type: none"> html
메타데이터	<ul style="list-style-type: none"> head <ul style="list-style-type: none"> base meta title link style
스크립팅	<ul style="list-style-type: none"> script noscript
섹션	<ul style="list-style-type: none"> body <ul style="list-style-type: none"> nav <ul style="list-style-type: none"> h1, h2, h3, h4, h5, h6 section 요소 <ul style="list-style-type: none"> article address <ul style="list-style-type: none"> aside <ul style="list-style-type: none"> header
그룹핑	<ul style="list-style-type: none"> p pre ol dl hr dialog ul dt br blockquote li dd
인라인 의미 요소	<ul style="list-style-type: none"> a dfn samp bdo q abbr kdb ruby cite time sub rt em progress sup rp strong meter span small code i mark var b
편집	<ul style="list-style-type: none"> ins del
포함 콘텐츠	<ul style="list-style-type: none"> figure embed video canvas img object audio map iframe param source area
표 데이터	<ul style="list-style-type: none"> table col tfoot th caption tbody tr colgroup thead td
폼	<ul style="list-style-type: none"> form input datalist textarea fieldset button optgroup output label select option
인터랙티브 요소	<ul style="list-style-type: none"> details command bb menu
기타 요소	<ul style="list-style-type: none"> legend div

브라우저 호환성을 위한 스크립팅

HTML5에 대한 브라우저의 지원이 아직 완벽하지 않기 때문에 특정 브라우저의 기능을 대상으로 하거나 누락된 기능을 추가하기 위해 자바스크립트를 활용한다.

Modernizr

- 가장 효율적인 HTML5 호환성 감지 도구는 **Modernizr**라는 오픈소스 자바스크립트 라이브러리이며, HTML5와 CSS3에 대한 지원 여부를 감지한다.
- 최신 버전의 **Modernizr** 라이브러리는 modernizr.com 웹사이트에서 다운로드할 수 있다.

```
<!DOCTYPE html>
<html>
<head><title></title>
  <script src="modernizr.min.js"></script>
</head>
```

HTML5를 활성화하는 스크립트 – Polyfills

- HTML5 Modernizr는 감지 도구이다. 즉, 이 라이브러리는 HTML5 기능이 오래된 브라우저에서 지원되지 않는다고 해서 이 기능을 추가하지는 않는다. 그러나 Modernizr 라이브러리 사이트는 “HTML5-크로스-브라우저-Polyfill”이라는 페이지를 제공하며, 이 페이지에서는 다양한 솔루션들에 대한 링크를 제공한다.
- **Polyfill**은 개발자가 특정 기능이 지원되지 않는 브라우저를 위해 사용할 수 있는 코드 조각이나 플러그인을 말한다. 폴리필은 HTML5 및 CSS3와 오래된 브라우저 사이의 간격을 메꾸는 역할을 담당한다.