

원칙 4 견고성

"Robust"

2011 웹 접근성 향상 전략세미나 (5.2)
신현석

KWCAG vs. KWCAG2

KWCAG

- ◎ 기술적 진보성

- 구성된 콘텐츠는 웹 브라우저의 종류, 버전 등에 관계없이 사용될 수 있어야 한다.

KWCAG2

- ◎ 견고성

- 웹 콘텐츠는 미래의 기술로도 접근할 수 있도록 견고하게 만들어야 한다.

KWCAG vs. KWCAG2

KWCAG

- 4.1 (신기술의 사용) 스크립트, 애플릿 또는 플러그인(plug-in) 등과 같은 프로그래밍 요소들은 현재의 보조기술의 수준에서 이들 프로그래밍 요소들의 내용을 사용자에게 전달해 줄 수 있을 경우에만 사용하여야 한다.
- ~~4.2 (별도 웹사이트 제공) 콘텐츠가 항목 1.1에서 4.1에 이르는 13개 검사 항목을 만족하도록 최대한 노력하였으나 해결되지 않는 부분이 남아있다면 텍스트만의 콘텐츠를 제공하는 웹 페이지(또는 웹사이트)를 별도로 제공해야 한다.~~

KWCAG2

- 4.1 (문법 준수) 웹 콘텐츠는 마크업 언어의 문법을 준수해야 한다.
 - 4.1.1 (마크업 오류 방지) 마크업 언어의 요소는 열고 닫음, 중첩 관계 및 속성 선언에 오류가 없어야 한다.
- 4.2 (웹 애플리케이션 접근성) 웹 애플리케이션은 접근성이 있어야 한다.
 - 4.2.1 (웹 애플리케이션 접근성 준수) 콘텐츠에 포함된 웹애플리케이션은 접근성이 있어야 한다.

마크업 오류 방지

마크업 언어의 요소는 열고 닫음, 중첩 관계 및 속성 선언에 오류가 없어야 한다.

- 마크업 언어를 해석하는데 있어서의 모호함과 문법적인 오류를 피해 다양한 사용자 도구(user agent)에서 웹 콘텐츠에 접근할 때 발생할 수 있는 문제를 최소화 한다.



마크업 오류 방지

정확한 태그의 용법 준수

- ◎ 태그의 닫음이 불명확 한 경우

`<p` 요소를 여는 태그의 닫음 표시가 불명확한 문단입니다.`</p>` (X)

- ◎ 닫는 태그의 표현이 잘못된 경우

`<p>`요소를 닫는 태그에 슬래시(slash)가 빠진 문단입니다.`<p>` (X)

마크업 오류 방지

정확한 속성 이름과 속성 값의 사용

- 닫는 따옴표가 누락된 경우

```
<input title="name type="text"> (X)
```

- 속성 사이의 공백이 누락된 경우

```
<input title="name"type="text"> (X)
```

- 공백이 있는 속성 값을 따옴표로 묶지 않은 경우

```
<input title=이름을 입력하세요. type=text> (X)
```

- 속성 값에 따옴표가 들어간 경우

```
<input type="text" value=""가나다""> (X)  
<input type="text" value="&quot;가나다&quot;"> (O)
```

마크업 오류 방지

정확한 속성 이름과 속성 값의 사용

◎ 아이디가 중복 선언된 사례

```
<div id="header"> ... </div>
<div id="content"> ... </div>
...
<form ...>
  ...
  <textarea id="content" name="content"></textarea>
  ...
</form>
...
</div>
<script type="text/javascript">
function validateForm()
  ...
  var el = document.getElementById('content');
  if (el.value)
    ...
  ...
</script>
```

마크업 오류 방지

정확한 태그의 중첩관계

◎ 태그의 열고 닫음이 잘못된 경우

```
<p>중첩관계가 <strong>명확해야 한다.</p></strong> (X)  
<p><strong>명확해야 한다.</strong></p> (O)
```

◎ 링크가 닫히지 않은 사례

```
<!-- 전략 -->  
    <li><a href="notice_list?id=748">내 글 목록 페이지 개선</li>  
  </ul>  
</div>  
<div>  
  <h3></h3>  
<!-- 후략 -->
```




Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Validate by **URI**

Validate by **File Upload**

Validate by **Direct Input**

Validate by URI

Validate a document online:

Address:

► [More Options](#)

Check

This validator checks the [markup validity](#) of Web documents in HTML, XHTML, SMIL, MathML, etc. If you wish to validate specific content such as [RSS/Atom feeds](#) or [CSS stylesheets](#), [MobileOK content](#), or to [find broken links](#), there are [other validators and tools](#) available.



The W3C validators are hosted on server technology donated by HP, and supported by community donations.

[Donate](#) and help us build better tools for a better web.



Flattr

[Home](#) [About..](#) [News](#) [Docs](#) [Help & FAQ](#) [Feedback](#) [Contribute](#)



This service runs the W3C Markup Validator, [v1.2](#).
COPYRIGHT © 1994-2010 W3C® (MIT, ERCIM, KEIO), ALL RIGHTS RESERVED. W3C LIABILITY, TRADEMARK, DOCUMENT USE AND SOFTWARE LICENSING RULES APPLY. YOUR INTERACTIONS WITH THIS SITE ARE IN ACCORDANCE WITH OUR PUBLIC AND MEMBER PRIVACY STATEMENTS.



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

HTML VALIDATOR (based on Tidy and OpenSP)

HTML Validator is a Mozilla extension that adds HTML validation inside Firefox and Mozilla. The number of errors of a HTML page is seen on the form of [an icon](#) in the status bar when browsing. The details of the errors are seen when looking the HTML source of the page.



The extension is based on [Tidy](#) and [OpenSP](#). Both algorithms were originally developed by the Web Consortium [W3C](#). And now extended and improved by a lot of persons. Both algorithms are embedded inside Mozilla/Firefox and makes the validation locally on your machine, without sending HTML to a third party server.



1. HTML Tidy is a helpful program that tries to help people to correct their HTML errors. It finds HTML errors and classifies them in 3 categories:

- errors: **HTML errors** that **Tidy cannot fix** or understand.
- warnings: **HTML errors** that **Tidy can fix** automatically
- (optional) **accessibility warnings**: HTML warnings for the 3 priority levels defined in [W3c WAI](#)



2. OpenSP, is SGML parser, is a professional algorithm. It is the same program running inside <http://validator.w3.org/>. And now, you can embed it in Firefox !

There is also an automatic "Clean up" button that will do his best to propose you a cleaned version of your page, without errors.

웹 애플리케이션 접근성 준수

콘텐츠에 포함된 웹 애플리케이션은 접근성이 있어야 한다.

- 접근성 적용 범위를 특정 기술에 국한하지 않고 웹 애플리케이션 일반으로 한다.
- 접근성을 지원하는 플랫폼이나 기술을 적극 권장한다.
- 현재의 보조기술이 지원하지 못하는 경우에도 접근성 기술을 사용해야 하고 대체 콘텐츠를 제공해야 한다.

웹 애플리케이션 접근성 준수

플러그인 기술이 제공하는 접근성 API 활용

- Best Practices for Accessible Flash Design
(http://www.adobe.com/resources/accessibility/best_practices/best_practices_acc_flash.pdf)
- Flex accessibility (<http://www.adobe.com/accessibility/products/flex/>)
- UI Automation and Microsoft Active Accessibility
(<http://msdn.microsoft.com/en-us/library/ms788733.aspx>)
- Silverlight Accessibility Overview
([http://msdn.microsoft.com/en-us/library/cc707824\(VS.95\).aspx](http://msdn.microsoft.com/en-us/library/cc707824(VS.95).aspx))
- Java Accessibility Helper Early Access v0.8
(<http://java.sun.com/developer/earlyAccess/jaccesshelper/>)
- Accessible Explorer (<http://msdn.microsoft.com/en-us/library/ms696082.aspx>)

웹 애플리케이션 접근성 준수

Adobe Flash

- Adobe Flash Player 10의 MSAA(Microsoft Active Accessibility) 지원 기능을 이용하여 보조기기에 정보를 제공할 수 있다.
- Adobe Flash의 접근 가능성(Accessibility) 패널이나 액션 스크립트(ActionScript)를 사용해서 접근성있는 플래시 콘텐츠를 제작할 수 있다.
 - WebAIM: Creating Accessible Flash Content(영문)
 - <http://www.webaim.org/techniques/flash/>
 - Adobe: Creating Accessible Sites in Flash(영문)
 - <http://www.adobe.com/accessibility/products/flash/tutorial/>

웹 애플리케이션 접근성 준수

Adobe Flash

● Accessibility panel

- Make Object Accessible, Make Child Objects Accessible: 체크를 해제하여 내용이 없는 무비가 접근이 되지 않게 함
- Auto label: 내부의 텍스트나 주변 텍스트로 자동 레이블 설정
- Name: 대체 텍스트
- Description: 상세한 설명
- Shortcut(professional): 단축키 지정
- Tab Index(professional): 키보드 접근 순서 설정



웹 애플리케이션 접근성 준수

웹 애플리케이션의 대체 콘텐츠 제공

- 보조기기에 따라서 플러그인을 사용 못하는 경우도 있다.
- 웹 콘텐츠에 기계적으로 접근하거나 저사양의 보조기기, PC 등의 경우 플러그인을 사용하지 못하는 경우가 많다.
 - 검색엔진, 낮은 플래시 플레이어 버전, 모바일 환경
- object 요소를 사용하여 웹 페이지에 콘텐츠를 제공할 때 대체 콘텐츠를 제공할 수 있다.
 - 외부 콘텐츠를 웹페이지에 포함시킬 때 사용
 - 플러그인 콘텐츠 외에 text, html, image 등을 삽입할 때에도 사용

웹 애플리케이션 접근성 준수

대체 콘텐츠 제공 방법

◎ <object> 대체 콘텐츠 제공 방법

- <object> 하위(child)의 콘텐츠를 대체 콘텐츠로 인식
- 대체 콘텐츠 제공의 갯수 제한은 없음

```
<!-- 플래시 플러그인 -->  
<object data="hello.swf" type="application/x-shockwave-flash">  
  <!-- PNG 이미지 -->  
  <object data="hello.png" type="images/png">  
    <!-- GIF 이미지 -->  
    <object data="hello.gif" type="images/gif">  
      <!-- 일반 text -->  
      <p>Hello!</p>  
    </object>  
  </object>  
</object>
```


웹 애플리케이션 접근성 준수

대체 콘텐츠 제공 방법

- ◎ 조건부 주석(conditional comment) 사용 방법
 - 일명 히시(Hixie) 방법
 - HTML 표준 문법을 따르며 하위 콘텐츠 갯수에 제한이 없음

```
<object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/
swflash.cab#version=7,0,0,0" width="550" height="400"
id="flash_movie" align="middle">
    <param name="movie" value="flash_movie.swf">
    <!--[if !IE]> <-->
    <object type="application/x-shockwave-flash"
data="flash_movie.swf" width="550" height="400">
        <p></p>
    </object>
    <!--> <![endif]-->
</object>
```

웹 애플리케이션 접근성 준수

잘못된 대체 콘텐츠

- 플러그인 설치 요구나 사용할 없는 환경임을 알리는 내용은 적절한 대체 콘텐츠가 아니다.
 - "해당 콘텐츠를 보려면 Flash Player가 필요합니다."
 - "Flash를 지원하는 브라우저가 필요한 미디어입니다."
 - "스크립트가 작동하지 않아 화면이 제대로 보이지 않을 수 있습니다."
- 완벽하게 내용을 대체 하지는 못하더라도 핵심 기능(텍스트 내용, 링크 등)을 대체 할 수 있어야 함

감사합니다

<http://hyeonseok.com>, hyeonseok@gmail.com