[1]웹접근성

1 웹접근성(Web Accessibility) : 웹 사이트에서 제공하는 정보를 차별(장애인 및 고령자 등을 포함한 모든 사람) 및 제한(다양한 플랫폼 및 장치, 웹 브라우저 등의 모든환경) 없이 동등하게 이용할 수 있도록 보장하는 것.

2 웹을 구성하는 가장 중요한 요소인 HTML은 다양한 플랫폼과 장치에 독립적인 정보교환 수단을 제공하기 위해 탄생되었다.

3 웹접근성 향사의 기대효과

1) 장애인, 고령자 등을 포함한 사용자층 확대

2) 규정과 법적 요구 사항에 따라 준수

3) 다양한 환경, 새로운 기기에서의 이용

4) 개발 및 운용의 효율성 제고

5) 사회 공헌 및 복지 기업으로서의 이미지 향상

4 웹콘텐츠 접근성 지침(WCAG)

1) 웹 접근성 이니셔티브(WAI: Web Accessibility Initiative) : 1997년에 웹의 표준화 관련 국제 기구인 월드 와이드 웹 컨소시엄(W3C: World Wide Web Consortium)에서는 1990년대 중반 이후 폭발적으로 성장한 웹 서비스에서 장애인의 접근성에 관련된 문제가 발생하자 이를 해결하기 위해 설립한 산하단체기관이다

2) 한국형 웹 콘텐츠 접근성지침

우리나라의 경우에는WAI에서 정한 웹 콘텐츠 접근성 지침을 바탕으로 한국적 특수성을 고려하여 웹 접근성을 준수하기 위한 KWCAG 1.0(2005. 12. 21 제정)을 발표했습니다. 이후 2009년 12월 23일 KWCAG 2.0이 확정되었으며, 2010년말부터 국가 표준으로 사용할 예정이다.

출처:한국정보통신기술협회(http://www.tta.or.kr/data/ttas\_view.jsp?pk\_num=TTAK.OT-10.0003/R1)

|  |  |  |
| --- | --- | --- |
| 한국형 웹 콘테츠 접근성 지침2.0  (TTAS.KO-10.0003/R1) | 한국형 웹컨텐츠 접근성 지침1.0  (TTAS.KO-10.0003) | 비고 |
| 1.1 대체 텍스트 | 1.1 텍스트가 아닌 콘텐츠의 인식 | 동일 |
| 1.2 멀티미디어 대체 수단 | 1.2 영상 매체의 인식 | 동일 |
| 1.3 명료성 | 1.3 색상에 무관한 인식 | 유사(추가) |
| 2.1 키보드 접근성 | 2.4 키보드만으로 운용가능 | 동일 |
| 2.2 충분한 시간 제공 | 2.6 반응 시간의 조절기능 | 동일 |
| 2.3 광과민성 발작 예방 | 2.3 깜빡거리는 객체 사용 제한 | 동일 |
| 2.4 쉬우 네비게이션 | 2.2 프레임의 사용 제한  2.5 반복 네비게이션 링크 | 동일 |
| 3.1 가독성 |  | 추가 |
| 3.2 예측 가능성 |  | 추가 |
| 3.3 콘텐츠의 논리성 | 3.1 데이터 테이블 구성  3.2 논리적 구성 | 동일 |
| 3.4 입력도움 | 3.3 온라인 서식 구성 | 추가 |
| 4.1. 문법준수 |  | 축소 |
| 4.2. 웹 어플리케이션 접근성 | 4..1 신기술의 사용  2.1 이미지 기법 사용 제한 | 삭제 |

4 웹 접근성 관련 참고 웹 사이트

1) 웹 접근성 연구소(<http://www.wah.or.kr>)

정보문화진흥원에서 운영하는 웹사이트로 웹접근성 관련 소식, 세미나 안내, 웹접근성 마크 소개, 웹접근성 콘텐츠 제작 사례 및 전문가 자문을 받을 수 있다

2) 한국 웹 접근성 그룹 콱(KWAG, <http://kwag.net>)

국내 웹 접근성을 향상시키기 위한 자발적인 모임으로써, 회원 상호간의 정보 공유와 스터디, 세미나 등을 진행하는 온,프라인 모임이다.

3) 정보 통신 접근성 향상 표준화 포럼(<http://www.iabf.or.kr>)

웹 접근성의 실태 파악, 표준화 정책 개발 및 확산 지원, 국제 표준화 대응 모색 등의 일을 하며, 관련 정보 자료를 볼 수 있다.

[2] 웹표준(Web Standards)

1 웹표준이란, ‘웹에서 표준적으로 사용되는 기술이나 규칙’을 의미하는데, 이는 우리가 흔히 생각하는 표준(standard)과는 조금 다르다. 웹에서의 표준은 W3C의 토론을 통해 나온 권고안(recomendation)을 말하며, 권고안 이외 단계 수준의 스펙은 비표준이거나 독자확장 요소를의미하기 때문에 구분하여 사용해야 한다.

W3C(World Wide Web Consortium)는 1994년 10월 미국의 MIT 컴퓨터 과학 연구소, 정보수학유럽연구 컨소시엄, 그리고 일본의 게이오 대학이 연합하여 만든 국제적인 웹 기술 표준 기구를 말한다. W3C의 역할은 정보, 의견 교환, 아이디어 창출, 독립적 사고, 그리고 공동의 이해를 위하여 명세, 가이드라인, 소프트웨어, 그리고 도구 및 규칙 등의 표준안을 제정함으로써 웹의 모든 잠재력(lead the web to its full potential)을 이끌어 내는 것이다.

W3C 사이트 -http://www.w3.org

2 웹 표준 관련기술

1) 구조언어 : (X)HTML & XML

① HTML : HTML은 ‘HyperText Markup Language’의 약자로, 하이퍼텍스트를 표현하기 위한 마크업 언어이다. HTML은 SGML(Standard Generalizes Markup Language)을 모체로 하여 국제표준기구인 ISO가 1986년에 채택한 웹에서 사용하는 마크업언어이다.

② XML(eXtensible Makeup Language)은 1996년 W3C(World Wide Web Consortium)에서 제안한 것으로, 웹에서 구조화된 문서를 전송할 수 있도록 설계된 표준화된 텍스트 형식이다. 이것은 인터넷에서 기존에 사용하던 HTML의 한계를 극복하고 SGML의 복잡함을 해결하는 방안으로 HTML에 담겨져 있는 형식적 요소를 완전히 배제하고 순수 데이터 포맷으로 작성된다.

③ XHTML은 W3C에서는 XHTML(eXtensible Hypertext Markup Language)을 ‘XML 응용으로써의 HTML4를 다시 공식화한 것’이라고 정의한다. HTML4.01과 XHTML1.0의 실제 내용은 큰 차이가 없다. HTML은 사람들에게는 느슨한 규칙이 좋았을지 모르지만, 이것을 처리하고 가공해야 하는 기계들에게는 좋지않은 것이다. 그래서 W3C는 좀 더 원활하게 기계적으로 처리되도록 XML의 형식을 빌어 HTML4.01를 재정의했다. 이후 기계에 더욱 친화적인 XHTML2.0을 정의했지만, 여러 가지 이유로 중단되고 HTML5로 전환하여 진행중이다.

④ HTML5은 HTML의 다음버전으로 HTML4를 업그레이드한 것이다. HTML5는 특정 플러그인에 의존하지 않고 콘텐츠를 제공하는 것이 목표인데, 아직 초기 단계이다. 그러나 많은 기업들이 HTML5의 표준화에 힘을 보태고, Firefox, Opera, Safari, Chrome등 최신의 웹 브라우저에서 기본적으로 HTML5를 지원하고 있으며, 마이크로소프트사 또한 앞으로 인터넷 익스플로러에서 HTML5를 지원하겠다고 발표했다.

2) 표현 언어 : CSS

CSS(Cascading Style Sheet)는 HTML3.2부터 지원하기 시작한 것으로, 웹 제작자와 사용자들의 필요에 의해 특별히 개발되었다. CSS에서는 폰트, 색상, 공백, 공간과 그 밖의 문서 표현 등을 자유롭게 지정할수 있는 기능을 제공한다. 기존의 HTML은 웹 문서를 다양하게 설계하고 수시로 변경하는 데에 많은 제약이 있었기 때문에 이것을 보완하기 위해 CSS를 만든 것이다. CSS는 W3C표준이므로 CSS를 이용하여 만든 문서는 사용자의 웹 브라우저 환경에 따라 홈페이지가 다르게 나타나지 않고, 어느 환경에서나 제작자가 의도한 대로 표현할 수 있다.

3) 동작 및 제어언어 : DOM & ECMA Script

웹 페이지의 요소를 객체화해서 동작을 제어하는데 사용하는 웹 표준 기술에는 DOM과 ECMA Script가 있다. DOM(Document Object Model)은 웹페이지의 구성 체계를 말한다. 이것은 HTML을 작성하면서 생성되는 논리적 규칙이다. 웹 브라우저는 이 논리적 구성 체계인 DOM을 해석하여 페이지를 표시하거나 ECMA Script등의 기술을 통하여 DOM의 구조를 변경할 수 있다. 즉, 사용자 측에서 작동하는 많은 동적 요소들이 DOM을 이용하여 객체 모델에 접근한 후, 스크립트 언어인 ECMA Script를 이용하여 웹 페이지의 요소의 동작을 제어하는 방법으로 작성한다.

3 웹 표준의 장점

1) 웹 접근성 수준의 향상

2) 검색 친화적인 웹 사이트 구현

3) 구조와 표현의 분리

4) 손쉬운 유지 보수 및 비용절감효과

4 웹 브라우저의 종류



[3] 구조를 위한 XHTML

1 마크업 언어의 분류

1. 순차적 마크업 언어(procedural markup language)

한 문서에 기본 데이터, 구조, 표현 정보가 저장되며, 문서 내용을 어떻게 시각적으로 표현할 것인가에 대한 정보를 비롯하여 폰트, 색상, 여백, 줄, 간격등 표현에 관련된 다수의 추가 정보가 포함된다. 대표적인 순차적 마크업 언어로 HTML이 있다.

1. 서술적 마크업 언어(descriptive markup language)

기본 데이터와 구조만으로 이루어져 있기 때문에 제목, 부제목, 작가, 서론, 주소, 참고 도서 등의 구조 정보만 가진다. 대표적이 서술적 마크업 언어로 XML이 있다.

순차적인 마크업 언어와 서술적 마크업 언어의 장점을 모두 수용한 마크업 언어로 XHTML이 있다.

2 XHTML의 서식

1. 요소 사용시 종료 태그의 생략불가능

HTML에서는 P, TR, TH, TD, LI 등의 요소명을 사용할 때 종료 태그를 생략할 수 있지만, XHTML은 이를 허용하지 않는다. 따라서 모든 요소는 반드시 시작과 종료를 선언한다.

1. 요소명과 속성명에 소문자 사용

HTML에서는 요소명과 속성명에 대소문자를 구분하지 않고 사용했지만, XHTML에서는 모든 요소명과 속성명에 소문자만 사용한다.

1. 빈 요소 사용 시 <요소명/> 형식으로 기수

HTML에서는 빈 요소의 경우 시작 태그만 기술해도 되지만, XHTML에서는 모든 요소에 종료의 표현을 기술해야 한다.

1. 속성에 속성값 생략 불가능

HTML에서는 몇 가지 속성의 경우 속성값을 생략할 수 있지만, XHTML에서는 이러한 단축 표기가 허용되지 않으므로 반드시 속성값을 지정한다.

1. 잘못된 중첩 사용 불가능

HTML에서는 규칙을 느슨하게 적용하여 중첩을 잘못해도 문제가 발생하지 않는 경우가 있었지만, XHTML에서는 잘못된 중첩이 허용되지 않는다.

1. 모든 속성값에 인용 부호 사용

HTML에서는 속성에 값을 지정할 때 인용 부호를 생략하 수 있지만, XHTML에서는 속성값은 시작과 끝을 반드시 인용 부호로 감싸야 한다.

<, >, & 을 &lt;, &ampl; 로 변환하여 사용한다

**[4] DTD(Document Type Declaration) 선언**

**1. DTD 선언의 이유**

**- 현재 문서가 어떠한 구조를 갖고 있는지, 어떠한 표준을 준수하고 있는지를**

**브라우저에게 알려주는 역할**

**- HTML과 XHTML은 여러 버전이 있으므로 정의해주어야 브라우져가 맞게 해석**

**- 올바른 DOCTYPE이 아니라면 마크업과 유효성 검사 불가능**

**- DOCTYPE에 따라 원하는 결과가 수행되지 않을 수도 있음**

**2. DOCTYPE의 종류**

**① Transitional : 유연하고 덜 까다로운 DTD**

**② Strict : 표현을 위한 마크업 요소와 속성을 사용하는 엄격한 DTD**

**③ Frameset : 90년대 스타일로 frameset을 사용하는 DTD**

**3. DOCTYPE 선언의 내용**

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  "http://www.w3.org/TR/html4/loose.dtd"> |

**① 공개 식별자와 시스템 식별자로 구성됨**

**- 공개 식별자 : -//W3C//DTD HTML 4.01 Transitional//EN**

**- 시스템 식별자 : http://www.w3.org/TR/html4/loose.dtd**

**② 공개 식별자는 문서형 정의를 참조하기 위한 식별자로 임의로 바꿀 수 없음**

**③ 시스템 식별자는 그 문서가 준수하는 문서형 정의를 참조하는 URI**

**(상대경로로 지정 가능)**

**④ HTML에서는 시스템 식별자 생략 가능 / XHTML에서는 생략 불가**

**4. DTD 버전의 종류**

**■ http://en.wikipedia.org/wiki/Document\_Type\_Declaration**

|  |
| --- |
| **HTML 4.01 Strict** |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  "<http://www.w3.org/TR/html4/strict.dtd>"> |
| **HTML 4.01 Transitional** |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  "<http://www.w3.org/TR/html4/loose.dtd>"> |
| **HTML 4.01 Frameset** |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"  "http://www.w3.org/TR/html4/frameset.dtd"> |
| **XHTML 1.0 Strict** |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  "<http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>"> |
| **XHTML 1.0 Transitional** |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> |
| **XHTML 1.0 Frameset** |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd"> |
| **XHTML 1.1** |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"  "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd"> |

**5. 사용할 DOCTYPE 선정**

**① 가장 무난한 DTD는 XHTML 1.0 Transitional**

**- a요소의 target 속성 사용 가능**

**- iframe 사용가능**

**■ 네임스페이스와 언어코드**

|  |
| --- |
| <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko" lang="ko"> |

**1. <html> 요소에 xmlns 속성과 lang 속성에 코드를 지정**

**2. xmlns 속성 : 네임스페이스를 지정**

**XHTML에는 [http://www.w3.org/1999/xhtml]이라는 네임스페이스가**

**주어져 있으므로 이를 값으로 지정**

|  |
| --- |
| 네임스페이스는 단지 구분의 의미밖에 없음 |

**3. 언어코드는 xml:lang 속성으로 지정함**

**- 언어코드는 ISO 639 로 규정되어 있음**

**(http://ko.wikipedia.org/wiki/ISO\_639)**

**- 한국어는 [ko], 영어는 [en], 프랑스어는 [fr], 중국어는 [zh] 등**

**- XHTML1.0에서는 후방호환성(오래된 브라우저도 문제없이 표시해, 안정적**

**으로 동작하도록 하는 것) 을 고려하여 lang 속성으로 xml:lang 속성과**

**같은 값은 지정한다. (XHTML1.1에서는 lang속성자체가 인정되지 않기**

**때문에 xml:lang 속성만 언어코드로 지정한다.)**

**■ 타이틀**

**1. <head> 요소 안에 반드시 한번만 지정**

**2. 페이지의 내용을 구체적으로 표현**

|  |
| --- |
| **잘못된 예 :**  <title>설명</title> |
| **바뀐 예 :**  <title>XHTML과 HTML에 대한 설명</title> |

**■ MIME 타입과 문자코드 셋**

|  |
| --- |
| <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  <title>한글로 되어 있는 제목</title> |

**1. <head>요소 안의 자식 <meta>요소의 http-equiv 속성, content속성**

**2. 한글등 비 ASCII문자가 나오는 부분보다 앞(title 태그보다 앞부분)에 지정**

**3. http-equiv 속성은 HTTP헤더와 동일한 정보를 추가적 또는 우선적으로**

**출력지시**

|  |  |  |  |
| --- | --- | --- | --- |
| **MIME 타입** | **HTML4** | **XHTML1.0** | **XHTML 1.1** |
| **Text/html** | **추천** | **가능** | **비추천** |
| **Application/xhtml+xml** | **금지** | **추천** | **추천** |
| **Application/xml** | **금지** | **가능** | **가능** |
| **Text/xml** | **금지** | **가능** | **가능** |

**4. 문자코드셋은 xml선언 속성에서 지정한 경우 중복하지 않아도 되나, 일부**

**브라우져에서 인코딩 속성의 정의를 정상적으로 인식 못할수도 있으므로**

**“charset=UTF-8” 으로 지정하는 편이 좋음**

**■ 설명문과 키워드**

|  |
| --- |
| <meta name="Keywords" content="xml,tutorial,html,dhtml,css,xsl,xhtml,javascript,asp,ado,  vbscript,dom,sql,colors,soap,php,authoring,programming,training,learning,beginner's guide,  primer,lessons,school,howto,reference,examples,samples,source code,tags,demos,tips,links,  FAQ,tag list,forms,frames,color table,w3c,cascading style sheets,active server pages,dynamic  html,internet,database,development,Web building,Webmaster,html guide" />  <meta name="Description" content="Free HTML XHTML CSS JavaScript DHTML XML DOM XSL XSLT RSS AJAX ASP ADO PHP SQL tutorials, references, examples for web building." /> |

**1. meta요소에 name 속성과 content 속성을 추가해 메타데이터를 지정가능**

**2. name 속성 : description이라 지정해 설명문을 기술**

**3. name 속성 : keyword라 지정해 keyword를 기술**

**4. 검색엔진이 읽어드림 / 하지만 악용하는 사이트가 있었기 때문에 현재는**

**많이 반영되지는 않음**

**5. 하지만 일부 검색엔진에서는 아직도 리스트하고 있으므로 지정하는 것이 좋음**

**6. 길이는 권고안에 없지만 적절한 길이 / 키워드는 동의어와 유의어 포함**