

ch01.HTML

디지털마케팅 프로그래밍

by Wooyoung Choi, KMU Digital Marketing MBA, 2025

Agenda

ch01.HTML

- HTML, Web Standard, Accessibility
- Anatomy of an HTML document
 - head elements
 - body elements
- semantic elements
- data, user input

Web Standards, Accessibility

HTML

HyperText Markup Language

- 웹 콘텐츠의 구조와 의미를 정의하기 위해 사용
- Hypertext: 참조(Hyperlink)를 통해 다른 문서에 접근할 수 있는 텍스트
- Markup: 텍스트, 이미지 등의 콘텐츠를 웹 브라우저에서 표현하기 위한 지시사항

Web Standards

- 웹에서 표준적으로 사용되는 기술이나 규칙
- W3C(the World Wide Web Concoortium)의 토론을 통해 나온 권고안(Recommandation):
<https://www.w3.org/standards/>

기대효과

1. 웹 접근성 수준 향상
2. 검색친화적 웹사이트 구현
3. 구조와 표현의 분리
4. 유지보수 용이, 비용 절감
5. 호환성

디지털 마케팅 관점에서 웹 표준의 중요성

1. 유저에게 일관성 있는 콘텐츠 제공
2. 검색엔진 최적화
3. 호환성

Web Standards(HTML)

```
<!doctype html>
<html lang="ko-KR">
<head>
  <meta charset="utf-8">
  <title>Title</title>
</head>
<body>
  <header>Page header</header>
  <nav>Navigation menu</nav>
  <main>
    <section>Content</section>
  </main>
  <aside>Sub content</aside>
  <footer>Page footer</footer>
</body>
</html>
```

Web Accessibility

- 웹사이트에서 제공하는 정보를 차별 및 제한없이 동등하게 이용할 수 있도록 보장하는 것
- W3C 웹 접근성 이니셔티브에서 지침 마련

장점

1. 정보 소외 계층을 포함한 사용자층 확대
2. 법적 요구사항 준수(장애인 차별 금지 및 권리구제 등에 관한 법률)
3. 다양한 환경, 기기에서의 사용
4. 개발 및 운영 효율성 제고

W3C 웹 접근성 이니셔티브(<https://www.w3.org/WAI/>)

- 인지성(Perceivable): 정보와 사용자 인터페이스 요소는 인지할 수 있도록 사용자에게 표시되어야 함
- 운용성(Operable): 사용자 인터페이스 요소와 탐색은 운용가능해야 함
- 이해성(Understandable): 정보와 사용자 인터페이스 운용은 이해할 수 있어야 함
- 견고성(Robust): 다양한 사용자 에이전트에 의존하여 해석될 수 있도록 내구성을 가져야 함

인지성

- 모든 텍스트가 아닌 콘텐츠에 대체 텍스트를 사람들이 원하는 인쇄, 점자, 음성, 기호 또는 간단 언어 등과 같은 형태로 제공해야 한다.
- 시간을 바탕으로 한 미디어에 대한 대안을 제공해야 한다.
- 정보와 구조의 손실 없이 콘텐츠를 다른 방식(예를 들면 더욱 간단한 형태로)들로 표현할 수 있어야 한다.
- 사용자들이 보다 쉽게 보고 들을 수 있는 전경에서 배경을 분리한 콘텐츠를 만들어야 한다.

운용성

- 키보드로 모든 기능을 사용할 수 있도록 해야 한다.
- 읽기 및 콘텐츠를 사용하는 사용자에게 충분한 시간을 제공해야 한다.
- 알려진 방법으로 발작을 일으킬 수 있는 콘텐츠를 디자인하지 않아야 한다.
- 사용자가 탐색하고, 콘텐츠를 찾고 그들이 어디에 위치하고 있는지를 알 수 있도록 도와주는 방법을 제공해야 한다.

이해성

- 텍스트 콘텐츠를 판독하고 이해할 수 있도록 만들어야 한다.
- 웹 페이지의 탐색과 운용을 예측 가능한 방법으로 제작해야 한다.
- 사용자의 실수를 방지하고 수정할 수 있도록 도와야 한다.

견고성

- 보조 기술을 포함한 현재 및 미래의 사용자 에이전트의 호환성을 극대화해야 한다.

HTML: HyperText Markup Language

Element

- HTML이 지시하기 위해 사용하는 기본 단위
- `<tagname attribute="value" empty-attribute>`의 구조로 이루어짐 `</tagname>`
- Empty elements: 내용이 없는 tag(img, link, meta, input, ..)
- tagname: 태그의 종류를 표현
- attribute="": 태그의 속성을 정의
- empty-attribute: 값이 존재하지 않는 binary attribute

Anatomy of HTML Document


```
<!DOCTYPE html>
<html lang="ko-KR">
  <head>
    <!-- commentary -->
    <meta charset="utf-8">
    <title>First page</title>
  </head>
  <body>
    <!--
      multiline commentary
    -->
    <p>첫 페이지 내용</p>
  </body>
</html>
```

- <!DOCTYPE html>: 문서의 종류를 선언
- <html></html>: 한 페이지의 시작과 끝을 표현하기 위한 요소 (root element)
- <head></head>: 페이지에 표현되지 않지만 필요한 정보를 정의하기 위한 요소
- <body></body>: 페이지에 표현될 정보를 정의하기 위한 요소
- <!-- Commentary -->: 주석 표현을 위한 요소(사용자에게 표현되지 않음)

head elements

- `<meta>`: HTML의 메타정보 정의를 위한 요소
- `<title></title>`: HTML 문서의 제목을 표현하기 위한 요소
- `<link>`: 외부 리소스 연결을 위한 요소(이미지, 스타일시트 등)
- `<script></script>`: 실행가능한 코드 첨부를 위한 요소

Metadata - <meta>

- <meta charset="utf-8">: 문서의 문자인코딩을 설정하기 위한 요소(웹 표준 인코딩: utf-8)
- <meta name="title" content="My first page">
- <meta name="author" content="John Doe">
- <meta name="viewport" content="width=device-width, initial-scale=1.0">면영역 설정
- <meta name="description" content="This site is to practice Web essentials">
- <meta property="og:image" content="img source url">
- <meta property="og:title" content="My first page">
- <meta property="og:description" content="This site is to practice Web essentials">

Metadata - <meta>

- <https://developers.google.com/search/docs/advanced/crawling/special-tags>
- Open Graph Data <https://ogp.me/>

External resources - <link>

<link href="" rel="" type="">

ex) <link rel="stylesheet" href="public/style.css" type="text/css">

Style elements - <style>

<style type="">

ex)

<style type="text/css">

body {

background-color:#fff;

line-height:1.4;

}

</style>

Script elements - <script>

- define and Execute script code
- <script src=""></script>
- default: 파일 다운로드가 끝날때 까지 HTML 파싱을 중단
- async: 파일 다운로드 완료시 바로 실행하며 HTML 파싱 중단
- defer: 파일 다운로드가 완료되어도 바로 실행하지 않고 HTML 파싱이 끝난 후 스크립트 실행

<script>

```
window.open("https://www.google.com/")
```

</script>

alt content - <noscript>

- 웹 브라우저가 script 요소를 지원하지 않을 때 보여줄 대체 콘텐츠 지정

<noscript>

Go to google

</noscript>

Structured Data

- 구조화된 데이터: 페이지에 관한 정보를 제공하고 콘텐츠를 분류하기 위한 표준화된 형식
- Rotten Tomatoes: 구조화된 데이터 제공 후 페이지 클릭률 25% 향상
- Rakuten: 구조화된 데이터가 포함된 페이지의 체류시간 1.5배
- JSON-LD 권장, schema.org 표현양식

```
<script type="application/ld+json">
```

```
{
```

```
  "@context": "https://schema.org",
```

```
  "@type": "Recipe",
```

```
}
```

```
</script>
```