

# 03.

## Outline / Headings

---

HTML 개요 알고리즘 이해.



# Outline

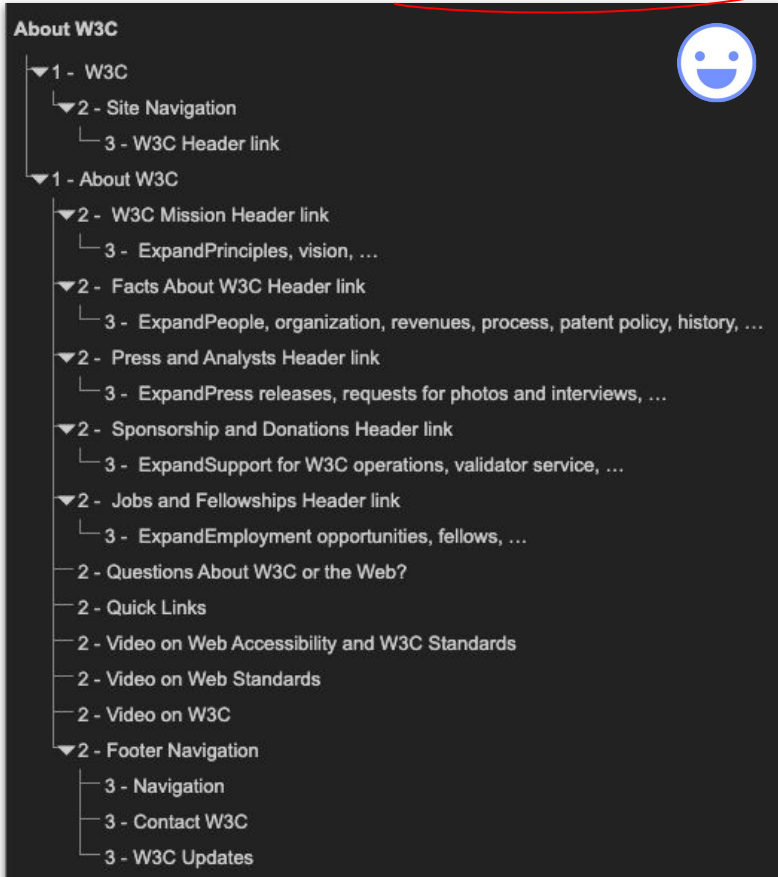
"개요, 윤곽"

"간결하게 추려 낸 주요 내용"


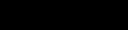
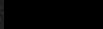
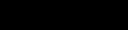
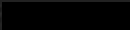
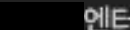
"웹 문서의 개요는 헤딩과 섹션으로 형성"

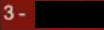
# Outline

## Chrome HeadingsMap Extension



# Outline

- 1 - 
- 2 -  promotion
- 2 - EVENT
- 2 - 기다리면무료작품
- 2 - BL의 세계로!! -  추천 BL 입문작, BL도 한걸음부터!
- 2 - 판타지는 거들 뿐 - 로맨스 먹후 여기 모여라!!
- 2 - 꿈같은 캠퍼스 라이프 - 우리 신입생 애그들을 위해 준비했어 ㄹㅇ
- 2 -  notice
- 1 -  Entertainment (주)  엔터테인먼트

- 국민가전 페스타 
- 3 -  단독 혜택
- 3 - 1 국민가전 공동구매 100명 이상 응치면, 베스트 국민가전 최대 60만원 할인!
  - 4 - 남은 시간
- 3 - 공동구매 2차 선착순 구매하기
- 3 - 공동구매 1차 선착순 구매하기
- 3 -  국민가전 공동구매 친구와 함께 참여해 보세요!
- 3 - 2 나에게 딱 맞는 대표 국민가전 BEST
- 3 - 3 포토 상품평 이벤트
- 3 - 더 많은 국민가전을 만나보세요
- 3 - TV | Lifestyle TV
- 3 - 냉장고
- 3 - 김치냉장고
- 3 - 세탁기 | 건조기
- 3 - 에어드레서
- 3 - 식기세척기
- 3 - 에어컨
- 3 - 공기청정기
- 3 - 인덕션 | 정수기
- 3 - 오디오
- 3 - 다양한 신제품도 만나보세요
- 2 - 페이지 링크
  - 3 - 제품
  - 3 - 고객센터
  - 3 - 부가정보
  - 3 - 기업정보
  - 3 - 윤리&준법경영

# Outline - Keyword

- 헤딩
- 섹셔닝
- 명시적/암시적 섹션
- 어색한 섹션

“

# TITLE vs HEADING



# Title **vs** Heading. What is the different?

- **<title>** 요소는 문서의 제목.  
문서에서 한 번만 사용할 수 있다.
- **<h\*>** 요소는 섹션의 제목.  
문서에서 여러 번 사용할 수 있다.

“

HEADING





# Heading

문서 개요를 형성하는 기본(==필수) 아이템.

웹 브라우저와 화면낭독기에 문서 개요를 드러내는 방법.



# Heading

## HTML

<h1>A

<h2>B

<h3>C



브라우저/화면낭독  
기



A



B



C

# Heading

## Check Point

"헤딩 없이 개요 없다"

“

## SECTIONING ROOT

! 브라우저 제조사가 오랜 시간  
HTML5 아웃라인 알고리즘을 구현하지 않았기 때문에  
이 개념을 실무에 사용하는 것을 권장하지 않음.



# Sectioning Root

- `<body>`
- `<blockquote>`
- `<details>`
- `<dialog>`
- `<figure>`
- `<fieldset>`
- `<td>`



브라우저 제조사가 오랜 시간  
HTML5 아웃라인 알고리즘을 구현하지 않았기  
때문에

이 개념을 실무에 사용하는 것을 권장하지 않음

# Sectioning Root

- HTML5에서 새롭게 추가된 명세.
- 독립적인 개요를 생성하는 개요 컨테이너.
- 섹셔닝 루트 외부에서 내부 개요에 접근 불가.
- 문맥 아닌 콘텐츠를 전체 개요에서 격리하는 역할.



브라우저 제조사가 오랜 시간  
HTML5 아웃라인 알고리즘을 구현하지 않았기  
때문에

이 개념을 실무에 사용하는 것을 권장하지 않음

# Sectioning Root

`<h1>A`  
`<h2>B`  
`<blockquote>`  
`<h3>C`

HTML5 명세의 의도

□ A  
□ B



브라우저/화면낭독기

□ A  
□ B  
□ C



브라우저 제조사가 오랜 시간  
HTML5 아웃라인 알고리즘을 구현하지 않았기  
때문에

이 개념을 실무에 사용하는 것을 권장하지 않음

“

# SECTIONING CONTENT





# Sectioning Content

- **<article>** // 기사, 본문, 맥락 독립적으로 배포 가능
- **<aside>** // 페이지의 주요 내용이 아닌
- **<nav>** // 사이트의 주된 탐색 메뉴
- **<section>** // 주제별로 나누거나 묶는

# Sectioning Content

- HTML5에서 새롭게 추가된 명세.
- 명시적 개요를 생성하는 개요 컨테이너.
- 적절한 수준의 헤딩을 자식 요소로 사용해야 한다.



웹 제작자가 잘못 설정한 헤딩 수준을 UA(브라우저, 화면낭독기)가 자동으로 보정하는 아웃라인 알고리즘 명세가 있지만, UA 제조사들이

# Sectioning Content

## HTML

<h1>A

<u>article

<h1>B

<u>section

<h1>C

## HTML5 명세의 의도

□ A

□ B

□ C



## 브라우저/화면낭독기

□ A

□ B

□ C



브라우저 제조사가 오랜 시간  
HTML5 아웃라인 알고리즘을 구현하지 않았기  
때문에

이 개념을 실무에 사용하는 것을 권장하지 않음

# Sectioning Content

## HTML

```
<h1>A  
<article>  
  <h2>B  
    <section>  
      <h3>C
```



## 브라우저/화면낭독기

```
□ A  
  □ B  
    □ C
```

# Heading

## Check Point

"섹셔닝 요소를 적극 사용하되  
아웃라인 알고리즘(섹셔닝 루트, 헤딩 수준 자동 보정)  
명세에는 의존하지 말 것"

“

명시적/암시적 섹션



## 명시적 섹션

헤딩과 함께  
섹셔닝 콘텐츠 (article, aside, nav, section) 를 사용하여  
섹션의 범위를 명시적으로 알 수 있는 섹션.

## 암시적 섹션

섹셔닝 콘텐츠(article, aside, nav, section)를 사용하지 않고 헤딩만을 사용하여 암시적으로 개요가 생성된 섹션.



# 명시적/암시적 섹션

## 명시적 섹션 😊

<h1>A

<article>

<h2>B

<section>

<h3>C

## 암시적 섹션 😞

<h1>A

<h2>B

<h3>C

“

어색한 섹션



# 어색한 섹션 - 최상위 헤딩이 없는

## HTML

```
<body>  
  <p>A  
  <article>  
    <h2>B
```



## 브라우저/화면낭독기



?



B

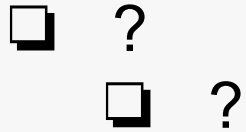
# 어색한 섹션 - 헤딩 없는 섹션

## HTML

```
<u>article>  
  <p>A  
  <u>section>  
    <p>B
```



## 브라우저/화면낭독기



“

# SUMMARY



## “ Summary

1. 헤딩을 사용하고 헤딩과 섹션(article, aside, nav, section)을 1:1로 맵핑하기.
2. HTML5 개요 알고리즘(섹셔닝 루트, 헤딩 수준 자동 보정)에 의존하지 않기.

Thank you !

