

9. 시맨틱(Semantic) HTML

9.1 시맨틱 태그란?

시맨틱 태그(Semantic Tag)란 단순히 화면에 콘텐츠를 보여주지만 하는 것이 아니라, 콘텐츠의 의미(semantic)와 역할(role)을 브라우저와 개발자, 그리고 검색 엔진·보조 기술(스크린 리더)에게 명확하게 전달하는 HTML 태그이다.

✓ "시맨틱"의 의미

- Semantic = "의미론적인", "의미를 담은"
- 예: `<div>` 는 "그냥 구획"일 뿐이지만, `<header>` 는 "헤더 역할"을 가진다는 뜻이 있음

✓ 시맨틱 태그의 목적

- 웹 접근성 향상
스크린 리더 등 보조 기술이 웹 콘텐츠의 구조와 의미를 이해할 수 있음
- SEO(Search Engine Optimization) 최적화
검색 엔진이 문서 구조를 정확히 분석 가능 → 검색 순위 향상에 도움
- 코드 가독성 향상
`<div id="header">` 대신 `<header>` 를 사용하면 의미가 명확해짐
- CSS/JS 구조화 용이
의미 기반 구조가 잡히면 스타일링과 기능 분리가 쉬워짐

✓ 시맨틱 태그 vs 비시맨틱 태그

구분	예시	설명
시맨틱 태그	<code><header></code> , <code><footer></code> , <code><article></code> , <code><nav></code> , <code><section></code> , <code><main></code> , <code><aside></code>	콘텐츠 의미를 명확히 나타냄
비시맨틱 태그	<code><div></code> , <code></code>	의미 없이 단순 레이아웃 구획 또는 인라인 영역용

✓ 시맨틱 태그가 도입된 계기

- 과거 HTML4/XHTML 시절에는 모든 블록 요소를 `<div>` 로만 나눴음
- 개발자 간 구조를 해석하기 어렵고, 보조 기술 및 검색 엔진도 의미를 알 수 없었음
- HTML5에서 의미를 가진 태그들이 대거 도입되며 시맨틱 구조가 정립됨

✓ 주요 시맨틱 태그 미리보기

태그	의미
<code><header></code>	페이지나 섹션의 머릿말
<code><footer></code>	페이지나 섹션의 바닥글
<code><main></code>	문서의 주요 콘텐츠
<code><nav></code>	주요 탐색 링크 영역
<code><article></code>	독립적인 글/게시글/뉴스/포스트 등
<code><section></code>	문서 내 의미 있는 구역
<code><aside></code>	보조 콘텐츠 (광고, 관련 링크 등)
<code><figure></code>	이미지, 도표, 코드 등과 캡션 묶음
<code><figcaption></code>	<code><figure></code> 요소에 대한 설명

✓ 예시 비교

✗ 비시맨틱 구조 (구식)

```
1 <div id="header">사이트 헤더</div>
2 <div id="nav">메뉴</div>
3 <div id="content">본문</div>
4 <div id="footer">푸터</div>
```

✓ 시맨틱 구조 (현대적 HTML5)

```
1 <header>사이트 헤더</header>
2 <nav>메뉴</nav>
3 <main>
4   <section>본문</section>
5 </main>
6 <footer>푸터</footer>
```

✓ 브라우저 호환성

- 대부분의 최신 브라우저는 HTML5 시맨틱 태그를 완벽 지원
- 구형 IE(Internet Explorer)에서 사용 시 CSS에서 `display: block;` 을 명시해줘야 했지만 현재는 거의 무관

✓ 한 줄 요약

시맨틱 태그는 콘텐츠의 역할과 의미를 명확히 표현하는 HTML 태그로, 웹 접근성, 검색 최적화, 코드 유지보수성을 향상시키는 핵심 기술이다.

9.2 <header>, <footer>, <main>, <section>, <article>, <aside>

이 단락에서는 HTML5에서 가장 많이 사용되는 **6가지 시맨틱 태그**의 의미와 용도, 그리고 언제 어떤 기준으로 사용해야 하는지를 예제와 함께 정리한다.

✓ 1. <header>

문서 또는 구획의 머리말(헤더) 역할

- 주로 사이트 제목, 로고, 내비게이션 메뉴, 검색창 등을 포함
- 문서 전체의 헤더뿐 아니라 <section> 이나 <article> 내에서도 사용 가능

```
1 <header>
2   <h1>My Blog</h1>
3   <nav>...</nav>
4 </header>
```

✓ 2. <footer>

문서 또는 구획의 바닥글(푸터) 역할

- 작성자 정보, 저작권, 사이트 링크, 연락처, 저작일자 등을 포함
- <header> 와 마찬가지로 문서 전체나 개별 섹션에서도 사용 가능

```
1 <footer>
2   <p>© 2025 by o jeongseok</p>
3 </footer>
```

✓ 3. <main>

페이지의 핵심 콘텐츠 영역

- 한 문서에 단 1번만 사용 가능
- <header>, <footer>, <nav>, <aside> 를 제외한 주요 콘텐츠가 들어감

```

1 <main>
2   <article>
3     <h2>HTML 소개</h2>
4     <p>HTML은 HyperText Markup Language입니다.</p>
5   </article>
6 </main>

```

🔴 스크린 리더는 이 영역을 "주 콘텐츠"로 인식함 → 접근성 향상

✓ 4. <section>

주제별 구획을 정의

- 문서 내에서 논리적으로 묶이는 주제별 콘텐츠를 나눌 때 사용
- 보통 h1~h6 제목 태그와 함께 사용

```

1 <section>
2   <h2>뉴스</h2>
3   <article>...</article>
4 </section>

```

🔴 의미 없는 레이아웃 구분은 <div>를 써야 하며,
<section>은 제목이 있는 논리적 구획이어야 함

✓ 5. <article>

독립적인 콘텐츠 단위

- 블로그 글, 뉴스 기사, 포럼 글, 리뷰, 카드 등 혼자서도 완전한 정보 덩어리
- 제목, 본문, 푸터 등을 내부에 모두 포함 가능

```

1 <article>
2   <h2>사용자 후기</h2>
3   <p>이 제품은 정말 훌륭합니다!</p>
4   <footer>작성자: 홍길동</footer>
5 </article>

```

✓ 6. <aside>

보조 콘텐츠 영역

- 광고, 관련 링크, 인용, 참고 정보 등 본문과 간접적으로 관련된 콘텐츠
- 사이드바 또는 기사 중 보조 박스로 쓰임

```
1 <aside>
2   <h3>관련 글</h3>
3   <ul>
4     <li><a href="#">HTML 기초</a></li>
5     <li><a href="#">CSS 소개</a></li>
6   </ul>
7 </aside>
```

✓ 종합 예제

```
1 <body>
2   <header>
3     <h1>Web Dev School</h1>
4     <nav>
5       <a href="#">HTML</a>
6       <a href="#">CSS</a>
7     </nav>
8   </header>
9
10  <main>
11    <section>
12      <h2>최신 글</h2>
13      <article>
14        <h3>HTML5의 변화</h3>
15        <p>HTML5는 시맨틱 태그를 도입했습니다...</p>
16        <footer>작성일: 2025-05-22</footer>
17      </article>
18    </section>
19
20    <aside>
21      <h2>관련 정보</h2>
22      <p>HTML5 태그 참조 링크</p>
23    </aside>
24  </main>
25
26  <footer>
27    <p>Copyright © 2025 web dev school</p>
28  </footer>
29 </body>
```

✓ 한 줄 요약

시맨틱 태그는 콘텐츠의 역할과 구조를 명확하게 정의하여 웹 접근성과 유지보수성을 향상시키며, 각 태그는 특정 문맥에서 의미에 맞게 사용해야 한다.

9.3 <figure>, <figcaption>

<figure>와 <figcaption>은 이미지, 코드, 표, 차트 등 시각적 콘텐츠에 설명(caption)을 명시적으로 연결하기 위한 시맨틱 태그이다.
HTML5에서 도입된 이 구조는 시각자료와 설명을 하나의 의미 있는 단위로 묶어준다.

✓ 1. <figure>란?

- 독립된 삽화나 콘텐츠 묶음을 의미
- 주로 이미지, 도표, 인용, 코드 블록 등과 함께 사용
- 본문과 관계가 있거나, 때로는 독립적으로도 의미를 가짐

```
1 <figure>
2   
3 </figure>
```

✓ 2. <figcaption>이란?

- <figure>의 자막(caption) 역할
- 설명 텍스트는 <figure> 내부에 위치해야 하며, 앞/뒤 어디든 배치 가능

```
1 <figure>
2   
3   <figcaption>2025년 4월 슈퍼문</figcaption>
4 </figure>
```

✦ <figcaption>은 반드시 <figure> 내부에 있어야 한다.
외부에 있으면 HTML 유효성 검사에서 오류 발생

✓ 3. 시맨틱한 구조의 이점

- 스크린 리더 및 보조 기술이 그림과 설명을 같은 단위로 인식함
- 검색 엔진이 시각 자료의 문맥과 설명을 정확히 이해할 수 있음
- CSS로 묶어서 쉽게 스타일링 가능 (<figure> > , <figcaption> 등)

✓ 4. 실제 예시: 이미지 + 설명

```
1 <figure>
2   
3   <figcaption>2023년 에과도르 화산 폭발 장면</figcaption>
4 </figure>
```

✓ 5. 실제 예시: 코드 스니펫 + 설명

```
1 <figure>
2   <pre><code>console.log('Hello, world!');</code></pre>
3   <figcaption>JavaScript에서 콘솔 출력</figcaption>
4 </figure>
```

✓ 6. 스타일링 예시 (CSS)

```
1 figure {
2   margin: 1em auto;
3   text-align: center;
4 }
5
6 figcaption {
7   font-size: 0.9em;
8   color: gray;
9 }
```

✓ 주의할 점

항목	주의사항
위치	<code><figcaption></code> 은 반드시 <code><figure></code> 안에 있어야 함
의미	단순한 이미지 레이아웃이면 <code><div></code> 도 가능하지만, 설명이 필요하다면 꼭 <code><figure></code> 를 사용
중첩	<code><figure></code> 안에 여러 요소가 올 수 있으나, 너무 복잡하게 구성하지 말 것

✓ 한 줄 요약

`<figure>` 와 `<figcaption>` 은 이미지나 콘텐츠와 그 설명을 시맨틱하게 묶는 구조로, 웹 접근성과 의미 전달에 중요한 역할을 한다.

9.4 `<time>`, `<mark>`, `<progress>`, `<meter>`

이 항목은 HTML5에서 도입된 **의미 있는 특수 목적 시맨틱 태그들**을 다룬다.

이들은 특정 데이터 유형이나 상태를 **명확하게 표현하고**,
웹 접근성과 자동 처리 능력을 높이기 위해 만들어졌다.

✓ 1. <time> - 시간 또는 날짜 표현

📌 목적:

- 날짜/시간 데이터임을 명확히 표시
- 검색 엔진이나 브라우저가 자동으로 이해하도록 도와줌

✓ 사용법:

```
1 <time datetime="2025-05-22">2025년 5월 22일</time>
```

- `datetime` 속성은 ISO 8601 형식을 사용함 (YYYY-MM-DD, T00:00 등)
- 이 태그는 이벤트 시간, 포스트 작성일, 마감 기한 등 다양한 곳에서 활용됨

예:

```
1 <p>작성일: <time datetime="2025-05-22T14:00">2025년 5월 22일 오후 2시</time></p>
```

✓ 2. <mark> - 강조 표시

📌 목적:

- 본문 내에서 현재 문맥상 강조할 부분을 표시 (예: 검색 키워드)

✓ 사용법:

```
1 <p>검색어 <mark>HTML</mark>이 포함된 결과입니다.</p>
```

- 기본적으로 노란 배경이 입혀짐
- 의미 없는 강조라면 ``, 중요성 강조라면 ``

✓ 3. <progress> - 작업 진행 상황 표시

📌 목적:

- 비율 기반 작업 진행 상태를 시각적으로 표현
- 진행 상태: 로딩, 업로드, 처리 중 등

✓ 사용법:

```
1 <progress value="70" max="100">70%</progress>
```

- `value`: 현재 진행된 수치
- `max`: 전체 수치

예:

```
1 <label for="upload">업로드 진행률:</label>
2 <progress id="upload" value="32" max="100">32%</progress>
```

- 브라우저에서 자동으로 프로그레스 바로 렌더링됨

✓ 4. <meter> - 범위 내의 값 시각화

📌 목적:

- 정량적 값(점수, 수치 등)을 시각적으로 표현
- 성적, 평점, 배터리 상태 등 절대값보다는 범위 기반 표시에 적합

✓ 사용법:

```
1 <meter value="0.7" min="0" max="1">70%</meter>
```

- low, high, optimum 속성도 존재하여 색상 힌트 가능 (브라우저마다 다름)

예:

```
1 <p>배터리 상태: <meter value="0.3" min="0" max="1" low="0.2" high="0.8" optimum="1">30%
  </meter></p>
```

📌 <progress> 는 작업의 흐름, <meter> 는 수치의 상태값을 표현할 때 사용해야 한다.

✓ 비교 요약 표

태그	의미	주 사용 목적
<time>	날짜/시간 표현	게시일, 마감일, 예약 시간 등
<mark>	문맥상 강조	검색 결과 강조, 키워드 하이라이트 등
<progress>	진행률 (0 ~ max)	로딩, 업로드 진행 상태
<meter>	값의 범위 내 상태 표시	점수, 성능 지표, 레벨, 배터리 상태 등

✓ 시맨틱 + 접근성 고려

- <time>: 스크린 리더가 날짜/시간으로 인식
- <mark>: 강조 음성 톤으로 읽을 수 있음
- <progress>: 진행률을 청각적으로 안내
- <meter>: 상태값을 알기 쉬운 음성으로 해석 가능

✓ 한 줄 요약

`<time>`, `<mark>`, `<progress>`, `<meter>` 는 시각적 요소의 의미를 명확히 부여하는 HTML5 시맨틱 태그로, 데이터 표현의 정확성과 접근성 향상에 핵심적인 역할을 한다.

9.5 시맨틱 구조 설계 예시

HTML5의 핵심은 시맨틱(semantic) 구조이다.

이 항목에서는 실제 웹페이지 예시를 바탕으로

`<header>`, `<main>`, `<section>`, `<article>`, `<aside>`, `<footer>`,

그리고 `<nav>`, `<figure>`, `<time>`, `<mark>` 등 주요 시맨틱 태그를 실전에서 어떻게 배치하고 활용하는지 구조적으로 보여준다.

✓ 예시 시나리오: 블로그 홈페이지

📌 이 웹페이지는 다음과 같은 정보를 포함한다:

- 사이트 제목, 내비게이션 메뉴
- 블로그 포스트 목록
- 각 포스트에는 제목, 날짜, 이미지, 내용, 태그가 있음
- 오른쪽에는 인기 글, 검색 창 등 사이드 콘텐츠
- 하단에는 저작권 정보

✓ 시맨틱 구조 예시

```
1  <!DOCTYPE html>
2  <html lang="ko">
3  <head>
4    <meta charset="UTF-8">
5    <title>시맨틱 블로그</title>
6    <link rel="stylesheet" href="style.css">
7  </head>
8  <body>
9
10   <!-- 사이트 헤더 -->
11   <header>
12     <h1>Semantic Blog</h1>
13     <nav>
14       <ul>
15         <li><a href="#home">홈</a></li>
16         <li><a href="#about">소개</a></li>
17         <li><a href="#contact">연락처</a></li>
18       </ul>
19     </nav>
20   </header>
21
22   <!-- 본문 콘텐츠 -->
```

```

23 <main>
24   <!-- 메인 콘텐츠 영역 -->
25   <section aria-label="최근 포스트">
26
27     <article>
28       <header>
29         <h2>HTML5의 진화</h2>
30         <p><time datetime="2025-05-22">2025년 5월 22일</time></p>
31       </header>
32       <figure>
33         
34         <figcaption>HTML5 로고</figcaption>
35       </figure>
36       <p>HTML5는 웹 개발을 <mark>시맨틱 중심</mark>으로 변화시켰습니다...</p>
37       <footer>
38         <p>작성자: 홍길동 | 태그: <a href="#">#HTML</a>, <a href="#">#웹표준</a></p>
39       </footer>
40     </article>
41
42     <article>
43       <header>
44         <h2>CSS Flexbox로 레이아웃 만들기</h2>
45         <p><time datetime="2025-05-21">2025년 5월 21일</time></p>
46       </header>
47       <p>Flexbox는 CSS3에서 도입된 레이아웃 시스템입니다...</p>
48     </article>
49
50   </section>
51
52   <!-- 사이드 정보 -->
53   <aside aria-label="사이드바">
54     <section>
55       <h3>인기 글</h3>
56       <ul>
57         <li><a href="#">CSS Grid 기초</a></li>
58         <li><a href="#">웹 접근성 가이드</a></li>
59       </ul>
60     </section>
61
62     <section>
63       <h3>검색</h3>
64       <form>
65         <input type="search" placeholder="검색어 입력">
66         <button type="submit">검색</button>
67       </form>
68     </section>
69   </aside>
70 </main>
71
72 <!-- 하단 푸터 -->
73 <footer>
74   <p>&copy; 2025 Semantic Blog. All rights reserved.</p>
75 </footer>

```

```
76 |
77 | </body>
78 | </html>
```

✓ 핵심 시맨틱 설계 포인트

태그	역할 및 설명
<header>	전체 사이트의 머리말 + 각 포스트의 헤더에도 사용
<nav>	전역 탐색 메뉴 제공
<main>	주요 콘텐츠. 문서당 1회 사용
<section>	포스트들을 묶는 논리적 그룹
<article>	각 블로그 포스트 단위
<aside>	부가 정보(인기글, 검색 등)
<figure>	이미지와 설명 묶음
<figcaption>	이미지에 대한 설명
<time>	날짜 표현. ISO 8601 사용
<mark>	강조(문맥 강조 목적)
<footer>	사이트 전체 또는 포스트의 바닥글 정보

✓ 시맨틱 구조 설계의 이점

- 접근성(A11y): 스크린 리더가 각 영역을 명확히 인식
- 검색 엔진 최적화(SEO): 의미 구조 기반 콘텐츠 분석 가능
- 유지보수/협업 용이: 구조와 역할이 직관적으로 분리됨
- 스타일링이 간결해짐: CSS 설계 시 의미 기반 선택자 활용 가능

✓ 한 줄 요약

시맨틱 구조는 단순한 레이아웃 설계를 넘어, 의미와 역할이 드러나는 웹 구조로서 HTML5의 핵심 철학이며, 이를 구현한 코드는 접근성, 검색성, 유지보수성 모두를 향상시킨다.