

## 1. HTML 기본 구조

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>문서 제목</title>
  </head>
  <body>
    <!-- 본문 내용 -->
  </body>
</html>
```

---

## 2. 문서 구조 태그 (Semantic 태그)

<code>&lt;header&gt;</code>	머리말 영역
<code>&lt;nav&gt;</code>	네비게이션 메뉴
<code>&lt;main&gt;</code>	주요 콘텐츠 영역
<code>&lt;section&gt;</code>	구획/섹션 구분
<code>&lt;article&gt;</code>	독립된 콘텐츠 블록
<code>&lt;aside&gt;</code>	보조 콘텐츠 영역
<code>&lt;footer&gt;</code>	꼬리말 영역

---

## 3. 텍스트 관련 태그

<code>&lt;h1&gt; ~ &lt;h6&gt;</code>	제목 (1~6단계)
<code>&lt;p&gt;</code>	문단
<code>&lt;br&gt;</code>	줄바꿈
<code>&lt;strong&gt;</code>	굵게 강조 (의미 강조)
<code>&lt;b&gt;</code>	굵게 (시각 강조)
<code>&lt;em&gt;</code>	기울임 강조 (의미 강조)
<code>&lt;i&gt;</code>	기울임 (시각 강조)
<code>&lt;u&gt;</code>	밑줄
<code>&lt;span&gt;</code>	인라인 구간 묶기
<code>&lt;div&gt;</code>	블록 구간 묶기

---

## 4. 링크, 이미지, 멀티미디어

<code>&lt;a href="URL"&gt;</code>	하이퍼링크
<code>&lt;img src="경로" /&gt;</code>	이미지 삽입
<code>&lt;video&gt;</code>	비디오 삽입
<code>&lt;audio&gt;</code>	오디오 삽입
<code>&lt;iframe&gt;</code>	외부 페이지 삽입

---

## 5. 목록 태그

<code>&lt;ul&gt;</code>	순서 없는 목록 (●)
<code>&lt;ol&gt;</code>	순서 있는 목록 (1. 2. 3.)
<code>&lt;li&gt;</code>	목록 항목

---

## 6. 폼(Form) 관련 태그

<code>&lt;form&gt;</code>	사용자 입력 양식
<code>&lt;input&gt;</code>	텍스트, 체크박스, 라디오 등 입력
<code>&lt;textarea&gt;</code>	여러 줄 텍스트 입력
<code>&lt;select&gt;</code>	드롭다운 목록
<code>&lt;option&gt;</code>	드롭다운 항목
<code>&lt;label&gt;</code>	입력 필드의 설명
<code>&lt;button&gt;</code>	버튼

---

## 7. 기타 주요 태그

<code>&lt;table&gt;</code>	표 생성
<code>&lt;tr&gt;</code>	표의 행
<code>&lt;td&gt;</code>	표의 셀
<code>&lt;th&gt;</code>	표의 제목 셀
<code>&lt;style&gt;</code>	CSS 작성 (내부 스타일)
<code>&lt;script&gt;</code>	자바스크립트 작성
<code>&lt;link&gt;</code>	외부 파일 연결 (ex. CSS)
<code>&lt;meta&gt;</code>	문서 정보 설정

## 8. 폰트 크기 바꾸기

웹페이지는 기본적으로 픽셀(px) 단위로 구성되며, 폰트 크기도 주로 px 단위를 사용합니다. 그러나 다음과 같은 상대 단위도 종종 사용됩니다:

속성명	단위	설명
font-size	px	픽셀 단위로 폰트 크기를 설정합니다. (절대 단위)
	rem0.5 = 50%	**최상위 요소(html 태그)**를 기준으로 폰트 크기를 설정합니다. (상대 단위)
	em	부모 요소를 기준으로 폰트 크기를 설정합니다. (상대 단위)

- 💡 px: 절대 크기 → 고정된 크기
- 💡 rem: root element(html)의 font-size를 기준
- 💡 em: 부모 요소의 font-size를 기준

## 9. 너비와 높이 지정하기

HTML 태그의 영역 크기를 지정할 때는 width와 height 속성을 사용합니다. 이때 사용 가능한 단위는 다음과 같습니다:

속성명	단위	설명
width, height padding, margin	px	픽셀 단위로 너비와 높이를 고정된 크기로 설정합니다.
	%	부모 태그의 크기를 기준으로 상대적인 백분율로 설정합니다.
	vw	**브라우저 창 너비(Viewport Width)**를 기준으로 백분율로 설정합니다. 브라우저 전체 너비의 1% 단위.
	vh	**브라우저 창 높이(Viewport Height)**를 기준으로 백분율로 설정합니다. 브라우저 전체 높이의 1% 단위.

## 10. 블록 요소 vs 인라인 요소

인라인 요소	strong, em, a, span, img	<ol style="list-style-type: none"><li>1. 자동 줄바꿈이 없음, 같은 줄에 이어짐</li><li>2. width, height 설정 불가</li><li>3. 요소 자체의 텍스트 크기만큼의 크기</li><li>4. 인라인 요소끼리만 묶을 수 있음</li></ol>
블록 요소	h1~h6, p, ol, ul, dl, div, header, footer, section, article, aside, nav	<ol style="list-style-type: none"><li>1. 줄을 자동으로 바꿈</li><li>2. width, height 설정 가능</li><li>3. 너비를 지정하지 않으면 부모 너비의 100% 사용</li><li>4. 블록 요소 + 인라인 요소 모두 포함 가능</li></ol>

---

## ✓ 비교 요약

구분	블록 요소	인라인 요소
줄바꿈	자동 줄바꿈	줄바꿈 없음 (옆으로 나열됨)
크기 지정	<code>width</code> , <code>height</code> 지정 가능	크기 지정 불가 ( <code>font-size</code> 정도만 영향)
포함 관계	블록+인라인 모두 포함 가능	인라인 요소끼리만 포함 가능

## 📌 11. `position` 속성 정리

<code>relative</code>	요소의 현재 위치를 기준으로 상대적인 위치를 설정합니다.
<code>absolute</code>	가장 가까운 <code>position</code> 이 지정된 조상 요소를 기준으로 절대 위치를 설정합니다. 조상 중 없으면 <code>body</code> 기준.
<code>fixed</code>	**브라우저 화면(뷰포트)**을 기준으로 절대 위치를 설정합니다. 화면을 스크롤해도 고정됩니다.

## ✓ 12. Flexbox 속성 정리

<code>display</code>	<code>flex</code>	자식 요소들을 <code>flexbox</code> 환경으로 설정 (기본적으로 <code>block</code> 처럼 동작).
	<code>inline-flex</code>	자식 요소들을 <code>flexbox</code> 로 설정하되, 요소 자체는 <code>inline</code> 처럼 동작..
<code>flex-direction</code>	<code>row</code>	가로 방향(수평)으로 아이টে을 배치 (기본값).
	<code>column</code>	세로 방향(수직)으로 아이টে을 배치.
	<code>row-reverse</code>	가로 방향이지만 순서를 반대로 배치.
	<code>column-reverse</code>	세로 방향이지만 순서를 반대로 배치.
<code>flex-wrap</code>	<code>nowrap</code>	줄 바꿈 없이 한 줄에 모두 배치.
	<code>wrap</code>	공간이 부족하면 다음 줄로 줄 바꿈.
<code>flex-flow</code>	<code>row wrap</code>	<code>flex-direction</code> 과 <code>flex-wrap</code> 을 한 번에 설정하는 축약형 속성.

## ● justify-content

목적: **flex** 레이아웃의 자식요소 정렬. **flex** 컨테이너 내에서 **\*\*주 축(main axis) : direction 속성 즉 행 또는 열 방향\*\***을 따라 자식 요소들을 어떻게 정렬할지 설정

flex-start	자식 요소들을 시작 지점에 정렬 (기본값).
flex-end	자식 요소들을 끝 지점에 정렬.
center	자식 요소들을 가운데에 정렬.
space-between	자식 요소들의 사이 간격만 균등하게 배치 (양 끝은 붙음).
space-around	자식 요소들의 양쪽 여백을 균등하게 배치 (양 끝 여백은 다른 여백의 절반).
space-evenly	자식 요소들의 모든 여백을 균등하게 배치 (양 끝 포함).

## ● align-content / align-items

목적:

- **align-items**: 단일 행(한 줄)의 교차 축(**cross axis**: 메인 축의 반대 방향) 정렬
- **align-content**: 여러 행(줄바꿈이 발생한 경우)의 교차 축 정렬

속성

flex-start	자식 요소들을 시작 지점(위쪽 등)에 정렬.
flex-end	자식 요소들을 끝 지점(아래쪽 등)에 정렬.
center	자식 요소들을 가운데 정렬.
space-between	자식 요소들의 사이 간격만 균등하게 배치 (양 끝은 붙음).
space-around	자식 요소들의 양쪽 여백을 균등하게 배치 (양 끝 여백은 절반).
space-evenly	자식 요소들의 모든 여백을 균등하게 배치 (양 끝 포함).

## ● flex-grow 속성 정리

목적	<b>Flex</b> 컨테이너의 남은 공간을 각 자식 요소에 어떻게 비율로 나눌지 설정. 안쪽 여백을 조정함.
기본값	0 → 남은 공간을 차지하지 않음
단위	숫자 (비율 단위, 예: 1, 2, 3 등)
적용 대상	<b>Flex</b> 컨테이너의 자식 요소

## ● flex-shrink 정리

목적	부모 요소의 공간이 부족할 때, 자식 요소가 얼마나 줄어들 수 있는지 비율로 설정
기본값	1 → 공간이 부족하면 줄어들 수 있음.
단위	숫자 (비율 단위, 예: 0, 1, 2 등). 값이 클수록 다른 자식 요소 보다 더 많이 줄어듬.
적용 대상	Flex 컨테이너의 자식 요소

## ● flex-basis 개념 요약

목적	Flex 아이템의 기본 크기 설정. width 또는 height보다 우선 적용되며, 이후 flex-grow 또는 flex-shrink에 의해 크기가 조정됩니다.
기본값	auto (자식 콘텐츠 크기 또는 width/height 따름)
단위	px, %, em 등 가능
우선순위	flex-basis > width / height

## ● flex 속성 정리

목적	flex-grow, flex-shrink, flex-basis를 한 줄로 설정하는 축약형 속성
형식	flex: [grow] [shrink] [basis];
기본값	flex: 0 1 auto;
적용 대상	Flex 컨테이너의 자식 요소

## ✓ 13. Grid 레이아웃

상황	설명
2차원 배치 필요할 때	행(Row)과 열(Column)을 동시에 제어해야 할 때 (예: 사진 갤러리, 대시보드 등)
정렬된 반복 레이아웃	카드형 목록, 격자 리스트, 포트폴리오, 달력 등
정해진 레이아웃 템플릿	헤더 / 사이드바 / 본문 / 푸터처럼 지정된 구조가 있을 때
CSS만으로 빠르게 레이아웃 구성	미디어 쿼리 없이도 열 개수 조절 <code>grid-template-columns</code> 또는 <code>repeat(auto-fit, minmax(...))</code> 속성 사용