



GETTING STARTED WEB

HTML5

CSS

Y-A, DOMINICA KIM

목차

5

HTML5 기초

웹 문서의 구조를 정의하는 핵심 요소들을 학습합니다.

3

CSS 스타일링

웹 페이지의 디자인과 레이아웃을 꾸미는 방법을 배웁니다.

JS

JavaScript 동작

웹 페이지에 동적 기능을 추가하는 프로그래밍을 익힙니다.

웹 개발 기술 소개



HTML5

웹 페이지의 구조와 콘텐츠를 정의하는 마크업 언어로, 웹의 기본 골격을 만드는데 사용됩니다.



CSS

웹 페이지의 디자인, 레이아웃, 시각적 표현을 담당하며 HTML 요소의 스타일을 지정합니다.



JavaScript

웹 페이지에 동적 기능을 추가하는 프로그래밍 언어로, 사용자 상호작용과 동적 콘텐츠를 구현합니다.



jQuery

JavaScript 라이브러리로, 문서 탐색, 이벤트 처리, 애니메이션 및 Ajax 기능을 단순화하여 제공합니다.



React.js

사용자 인터페이스를 구축하기 위한 JavaScript 라이브러리로, 컴포넌트 기반 개발과 효율적인 DOM 관리를 지원합니다.

프론트엔드 필수 기술과 아키텍처

핵심 기술

- HTML5 - 웹 문서 구조 정의
- CSS3 - 스타일링 및 레이아웃
- JavaScript - 동적 기능 구현
- TypeScript - 타입 안전성 강화

프레임워크/라이브러리

- React - 컴포넌트 기반 UI 라이브러리
- Vue.js - 웹/모바일 모두 지원 프레임워크
- Angular - 완전한 프론트엔드 솔루션
- jQuery - DOM 조작 간소화

웹 서버/백엔드

- Node.js - JavaScript 런타임 환경
- Express - Node.js 웹 프레임워크
- Next.js - 서버 사이드 렌더링
- REST/GraphQL API - 데이터 통신

모바일 전용 프레임워크

- React Native - 크로스 플랫폼 앱 개발
- Flutter - 구글의 UI 프레임워크
- Ionic - 하이브리드 모바일 앱 개발
- Swift/Kotlin - 네이티브 앱 개발

웹 개발 필수 도구 및 핵심 기술

HTML5, CSS, JavaScript, jQuery, React부터 Spring/Spring Boot까지 효과적인 학습 경로



개발 환경 설정

웹 개발을 위한 필수 도구 설치

- VS Code 또는 WebStorm - 코드 편집기
- Node.js 및 npm - 패키지 관리
- Git - 버전 관리 시스템
- Chrome DevTools - 웹 디버깅 도구



프론트엔드 핵심 기술

웹 클라이언트 측 기술 학습

- HTML5 - 시맨틱 태그, 폼, 멀티미디어 요소
- CSS3 - Flexbox, Grid, 반응형 디자인
- JavaScript - ES6+, DOM 조작, 비동기 프로그래밍
- jQuery - 선택자, 이벤트 처리, Ajax 통신



React 생태계

현대적인 UI 개발 프레임워크

- Create React App - 프로젝트 생성 도구
- JSX - JavaScript 확장 문법
- 컴포넌트 라이프사이클 및 Hooks
- 상태 관리 - Redux, Context API



백엔드 기술 (Spring 준비)

서버 측 개발을 위한 기초

- Java 기본 문법 및 객체지향 개념
- Spring Framework - DI, AOP, MVC
- Spring Boot - 자동 구성, 내장 서버
- RESTful API 설계 및 구현

HTML5란?

HTML5의 정의 :

HTML5는 웹 문서 구조를 정의하는 마크업 언어의 최신 표준입니다.

2014년에 W3C에 의해 정식 권고안으로 발표되었으며, HTML4.01 이후 약 14년 만에 등장한 차세대 웹 표준입니다.

웹 개발에 있어 HTML5는 구조(HTML), 표현(CSS), 동작(JavaScript)이라는 웹의 세 가지 핵심 기술 중 '구조'를 담당합니다.

HTML5의 발전

HTML5는 WHATWG와 W3C가 협력하여 개발했으며, 웹 애플리케이션 개발을 위한 플랫폼으로서의 웹의 역할을 강화하는 방향으로 발전했습니다. HTML5는 단순한 마크업 언어를 넘어 웹 애플리케이션 구축의 기반이 되는 핵심 기술로 자리잡았습니다.

HTML5의 주요 특징

- 시맨틱 태그를 통한 문서 구조의 명확한 정의
- Canvas, SVG를 이용한 그래픽 처리 기능
- 오디오, 비디오 등 멀티미디어 요소 네이티브 지원
- 웹 스토리지, 웹소켓 등 다양한 API 제공
- 모바일 환경 최적화 및 반응형 웹 디자인 지원

HTML5의 온라인 리소스

HTML5 학습과 개발에 도움이 되는 주요 온라인 리소스와 사이트입니다.

분류	리소스	URL
공식 문서 및 표준	W3C HTML5 스펙 문서	https://www.w3.org/TR/html5/
	MDN Web Docs	https://developer.mozilla.org/ko/docs/Web/HTML
인터랙티브 학습 플랫폼	W3Schools	https://www.w3schools.com/html/
	Codecademy	https://www.codecademy.com/learn/learn-html
	freeCodeCamp	https://www.freecodecamp.org/learn/responsive-web-design/
개발 도구	CodePen	https://codepen.io/
	JSFiddle	https://jsfiddle.net/
	Chrome DevTools	https://developer.chrome.com/docs/devtools/
커뮤니티 및 포럼	Stack Overflow	https://stackoverflow.com/questions/tagged/html5
	GitHub	https://github.com/topics/html5

HTML5 - 개발 도구

HTML5 개발을 더 효율적으로 할 수 있게 도와주는 다양한 도구들이 있습니다:

- **코드 에디터:** Visual Studio Code, Sublime Text, Atom, Brackets - 코드 작성 및 문법 강조 기능 제공
- **웹 브라우저 개발자 도구:** Chrome DevTools, Firefox Developer Tools - 실시간 HTML 요소 검사 및 수정 가능
- **코드 검증 도구:** W3C Validator, HTML5 Validator - HTML 문법 오류 확인 및 표준 준수 검사
- **반응형 테스트 도구:** Responsive Design Checker, Chrome의 Device Mode - 다양한 화면 크기에서 테스트
- **프레임워크:** Bootstrap, Foundation - 빠른 HTML5 레이아웃 구성 및 컴포넌트 제공
- **버전 관리:** Git, GitHub - 코드 변경사항 추적 및 협업 지원

이러한 도구들을 활용하면 HTML5 문서를 더 쉽고 효율적으로 개발할 수 있습니다.

VS Code 명령 팔레트 및 웹 개발 플러그인

Visual Studio Code에서 명령 팔레트를 사용하는 방법:

- Windows: **Ctrl+Shift+P** 또는 **F1** 키
- Mac: **Command+Shift+P** 또는 **F1** 키

명령 팔레트를 통해 VS Code의 모든 명령에 접근하고, 설정을 변경하며, 다양한 작업을 수행할 수 있습니다.

HTML5, CSS, JS 개발을 위한 필수 VS Code 플러그인

플러그인 이름	제작자	기능	추천 이유
Live Server	Ritwick Dey	실시간 서버 및 새로고침 기능	코드 변경 시 브라우저에서 즉시 확인 가능
HTML CSS Support	ecmel	HTML에서 CSS 클래스 자동완성	CSS 클래스 및 ID 빠른 참조
Auto Rename Tag	Jun Han	HTML 태그 자동 짹 맞춤	여는/닫는 태그 동시 수정
IntelliSense for CSS	Microsoft	CSS 속성 및 값 자동완성	CSS 작성 속도 향상
JavaScript (ES6) Snippets	charalampos karypidis	JS 코드 스니펫 제공	자주 사용하는 JS 패턴 빠르게 삽입
ESLint	Microsoft	JavaScript 코드 오류 검사	코드 품질 및 일관성 유지
Prettier	Prettier	코드 자동 포맷팅	일관된 코드 스타일 유지
Path Intellisense	Christian Kohler	파일 경로 자동완성	링크 및 파일 참조 작성 용이

위 플러그인을 설치하면 HTML5, CSS, JavaScript 개발 효율성을 크게 높일 수 있습니다.

HTML5 주요 태그



구조 태그

`section, article, aside, header, footer`: 문서의 구조적 블록을 정의합니다.

`div`: 블록 레벨 컨테이너입니다.



콘텐츠 태그

`h1~h6`: 웹 페이지의 제목과 부제목을 정의합니다.

`p`: 문단을 정의합니다.



목록 태그

`ul, ol, li`: 순서 있는/없는 목록을 생성합니다.



테이블 태그

`table`: 테이블의 전체 구조를 정의합니다.

`tr`: 테이블의 행(row)을 정의합니다.

`td`: 테이블의 데이터 셀(cell)을 정의합니다.

`th`: 테이블의 헤더 셀을 정의합니다.

`caption`: 테이블의 제목을 정의합니다.

`thead, tbody, tfoot`: 테이블의 헤더, 본문, 푸터 영역을 구분합니다.



하이퍼링크 태그

`a`: 하이퍼링크를 생성합니다.



미디어 태그

`img`: 이미지를 삽입합니다.

`audio, video`: 오디오와 비디오를 재생합니다.



폼 태그

`form, input, button`: 사용자 입력을 처리합니다.



텍스트 스타일 태그

`b, i, strong, em`: 텍스트 스타일과 강조를 표시합니다.

이러한 기본 태그들은 HTML5 문서의 구조와 콘텐츠를 구성하는 핵심 요소입니다.

HTML5 - 문서 구조의 기본

HTML 문서는 다음과 같은 기본 구조를 가집니다:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>문서 제목</title>
  </head>
  <body>
    <!-- 실제 내용이 들어가는 부분 -->
    웹 페이지에 표시될 내용
  </body>
</html>
```

HTML 태그

모든 HTML 문서의 최상위 태그로, 전체 내용을 감싸는 루트 요소입니다.

</>

```
<html>
  <!-- 문서의 모든 내용 -->
</html>
```

HEAD 태그

메타데이터, 타이틀, 스타일시트, 스크립트 등 문서의 정보를 담습니다.

◐

```
<head>
  <meta charset="UTF-8">
  <title>페이지 제목</title>
  <link rel="stylesheet" href="style.css">
</head>
```

BODY 태그

사용자에게 보여지는 모든 콘텐츠가 들어갑니다.

◎

```
<body>
  <h1>제목</h1>
  <p>문단 내용</p>
  
</body>
```

H1 : Main Heading Title

H1 : Main Title

H2 : Section Heading

H2 : H2 : Section Heading

H1 (Later Frontiection (heading tage)

H3 : Sub-Subsection

H4 : Minor Section Circlections

H6-dubssecction

H4 : Sub-section Heading

F5 : Minor Heading (Dithesizing

H5 : Un-Miiverst Heading

H5 : Minor Smallest Heading

Far : Minor Heading, Creastieds

H6 : H6 Smallest Heading

HTML5 - 제목 태그 (h1~h6)

h1 태그

가장 중요한 제목으로 페이지당 하나만 사용하는 것이 좋습니다.

h2-h4 태그

부제목이나 소제목으로 사용되며 계층 구조를 형성합니다.

h5-h6 태그

작은 크기의 제목으로 콘텐츠의 세부 구분에 활용됩니다.

HTML5 - 문단 태그 (p)

문단 태그 코드

```
<p>첫 번째 문단입니다.  
문단 내에서 연속으로 쓰면  
한 줄로 표시됩니다.</p>
```

```
<p>두 번째 문단입니다.  
새로운 p 태그는 새 문단을  
만듭니다.</p>
```

브라우저 출력 결과

문단은 자동으로 위아래 여백이 생깁니다.
문단 내 줄바꿈은 무시되고 자동 줄바꿈이 적용됩니다.
새 문단은 명확히 구분되어 표시됩니다.

HTML5 - 줄바꿈 및 수평선 (br, hr)



br 태그 (줄바꿈)

문단 내에서 강제로 줄을 바꿀 때 사용합니다.

닫는 태그가 없는 빈 요소(empty element)입니다.

hr 태그 (수평선)

주제 전환이나 구분을 시각적으로 나타냅니다.

특징

닫는 태그가 없는 빈 요소입니다.

사용 방법

<hr>

효과

페이지에 가로선을 그려 콘텐츠를 시각적으로 구분합니다.



사용 예시

안녕하세요!
반갑습니다!

첫 번째 주제<hr>두 번째 주제

HTML5 - 목록 태그 (ul, ol, li)

순서 없는 목록 (ul)

```
<ul>  
<li>커피</li>  
<li>차</li>  
<li>주스</li>  
</ul>
```

브라우저 출력: 글머리 기호(bullet)로 표시됩니다.

순서 있는 목록 (ol)

```
<ol>  
<li>1단계</li>  
<li>2단계</li>  
<li>3단계</li>  
</ol>
```

브라우저 출력: 숫자로 자동 번호가 붙습니다.

HTML5 - 링크 태그 (a)

하이퍼링크를 생성하여 다른 웹 페이지나 같은 페이지의 다른 부분으로 이동할 수 있게 합니다.

기본 구조

링크의 기본 형식은 다음과 같습니다:

```
<a href="URL주소">링크 텍스트</a>
```

href 속성은 이동할 목적지 주소를 지정합니다.

새 창에서 열기

target 속성을 사용하여 링크를 새 탭이나 창에서 열 수 있습니다:

```
<a href="URL" target="_blank">새 창에서 열기</a>
```

target="_blank"는 링크를 클릭할 때 새 탭이나 창에서 페이지를 엽니다.

외부 링크

다른 웹사이트로 이동하는 링크 예시:

```
<a href="https://google.com">구글로 이동</a>
```

브라우저 출력: 구글로 이동 (클릭 시 구글 홈페이지로 이동)

내부 링크(앵커)

같은 페이지 내의 특정 위치로 이동하는 앵커 링크:

```
<a href="#section1">섹션1로 이동</a>
```

id="section1"이 있는 요소로 페이지 내에서 이동합니다.

HTML5 - 이미지 태그 (img)

이미지를 웹 페이지에 삽입하는 태그입니다. 기본 문법:

속성	설명
src	이미지 파일의 경로를 지정합니다. (로컬 또는 URL)
alt	이미지를 표시할 수 없을 때 대체 텍스트를 제공합니다.
width	이미지의 가로 크기를 설정합니다. (픽셀 또는 %)
height	이미지의 세로 크기를 설정합니다. (픽셀 또는 %)

HTML5 - 표(table) 기본 구조

표(table)는 행과 열로 구성된 데이터를 표시하는 HTML 요소입니다.

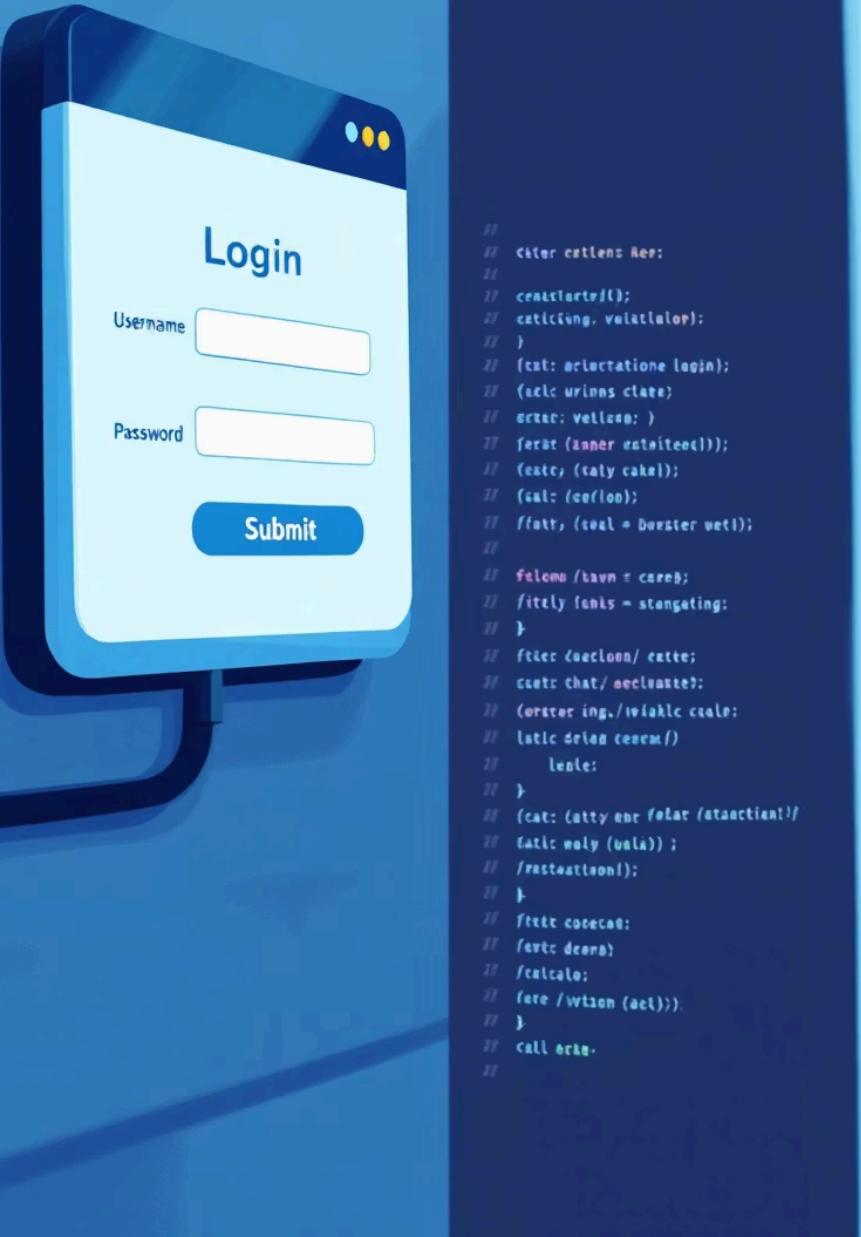
태그	설명	예시 코드
table	표 전체를 감싸는 태그	<table> ... </table>
tr	표의 행(row)을 나타내는 태그	<tr> ... </tr>
th	표의 제목 셀(header cell)을 나타내는 태그	<th>제목</th>
td	표의 일반 데이터 셀을 나타내는 태그	<td>데이터</td>
thead	표의 헤더 부분을 그룹화하는 태그	<thead> ... </thead>
tbody	표의 본문 부분을 그룹화하는 태그	<tbody> ... </tbody>
tfoot	표의 푸터 부분을 그룹화하는 태그	<tfoot> ... </tfoot>

표 구조 예시:

```
<table>
  <thead>
    <tr>
      <th>이름</th>
      <th>나이</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>홍길동</td>
      <td>25</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td colspan="2">표 하단 정보</td>
    </tr>
  </tfoot>
</table>
```

위 코드는 이름과 나이를 보여주는 표를 생성하며, **thead**, **tbody**, **tfoot** 태그로 표의 구조를 명확하게 구분합니다.

thead, **tbody**, **tfoot** 태그는 표의 구조를 논리적으로 구분하여 접근성을 향상시키고, CSS 스타일링을 더 효과적으로 적용할 수 있게 해줍니다. 이 태그들은 필수는 아니지만 대규모 표에서 특히 유용합니다.



```
//  
// eter etatens Rep:  
//  
// eteretrl();  
// eticating, valuator);  
//  
// (cat: etactatione login);  
// (etc urins clate)  
// etec: vellon; )  
// forat (anner etatiteel));  
// (etc; (aly cakel);  
// (alt: cu(lon);  
// flatt, (real = Duxter vet);  
//  
// felomo /avn = careB;  
// fitly lanks = stangating;  
//  
// ffilez (ueclon/ ette;  
// scatt that/ uecleante);  
// (orater ing,/veialic cale);  
// latic delia ceem/)  
// leale;  
//  
// (cat: (atty ent folat (ataction!)/  
// Eatic woly (wala);  
// /rastation!);  
//  
// fette coccat;  
// fetc deens)  
// /calate;  
// ferre /Wtien (act));  
//  
// call etek-
```

HTML5 - 입력 양식 (form, input)



form 태그

사용자 입력을 서버로 전송하기 위한 양식을 만듭니다.

input 태그

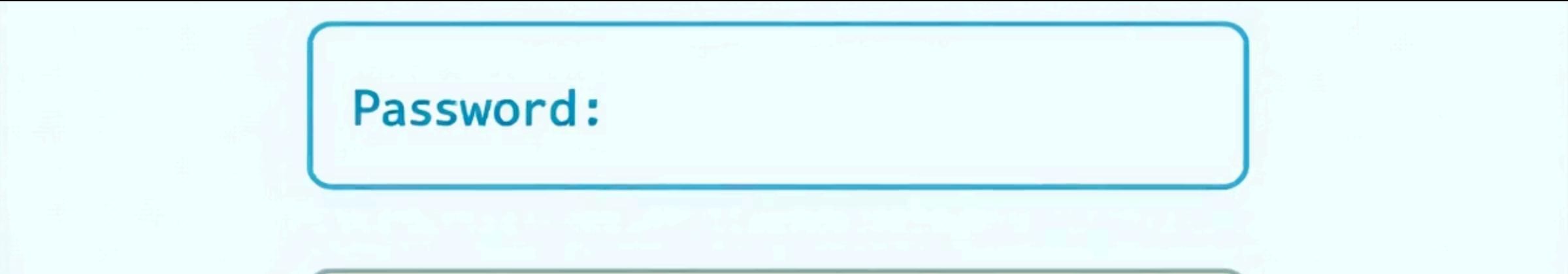
텍스트, 비밀번호 등 다양한 입력 필드를 만듭니다.

label 태그

입력 필드에 레이블을 붙여 사용성을 높입니다.

submit 버튼

form 데이터를 서버로 전송하는 버튼을 만듭니다.



Password:

HTML5 - 입력 타입 (text, password, email)

텍스트 입력

```
<input type="text"  
placeholder="이름을 입력하세요">
```

일반 텍스트를 입력받을 수 있는 한 줄 입력 필드입니다.

비밀번호 입력

```
<input type="password"  
placeholder="비밀번호">
```

입력한 내용이 점(*)으로 가려지는 보안 입력 필드입니다.

이메일 입력

```
<input type="email"  
placeholder="이메일 주소">
```

이메일 형식 검증 기능이 있는 입력 필드입니다.

HTML5 - textarea & select

textarea 태그

```
<textarea rows="4" cols="50"  
placeholder="여러 줄 입력...>  
</textarea>
```

여러 줄의 긴 텍스트를 입력받을 수 있는 필드입니다.

rows, cols 속성으로 크기를 지정할 수 있습니다.

select 태그

```
<select>  
  <option value="apple">사과</option>  
  <option value="banana">바나나</option>  
  <option value="orange">오렌지</option>  
</select>
```

드롭다운 선택 메뉴를 만듭니다.

option 태그로 선택 항목을 정의합니다.

HTML5 - 버튼 태그 (button)



기본 버튼

```
<button type="button">클릭하세요</button>
```



제출 버튼

```
<button type="submit">제출</button>
```



초기화 버튼

```
<button type="reset">초기화</button>
```



JavaScript 이벤트 연결

```
<button onclick="alert('안녕하세요')">경고창</button>
```



Standart

....



Butmoit



Button Alert butop ppoopup, ann clack and iniase ins
bear white: button all in scaralingte, for submigation.



Reset



Button

....

This Javascript alert alert button, clicked for and buttons
are cell you like your lapp, ho a into the mage.

HTML5 - 이미지 삽입과 링크 결합

웹 페이지에서 이미지를 클릭 가능한 링크로 변환하는 효과적인 웹 개발 기법입니다.



하이퍼링크 이미지

이미지 요소를 앵커 태그로 감싸 사용자 상호작용이 가능한 컨텐츠를 구현합니다. 이는 웹 사용성을 향상시키는 핵심 기법입니다.

마크업 구조

앵커 태그(<a>) 내부에 이미지 태그()를 중첩하여 구현합니다. 이는 HTML의 기본 요소 중첩 원칙을 활용한 구조입니다.

구현 예시

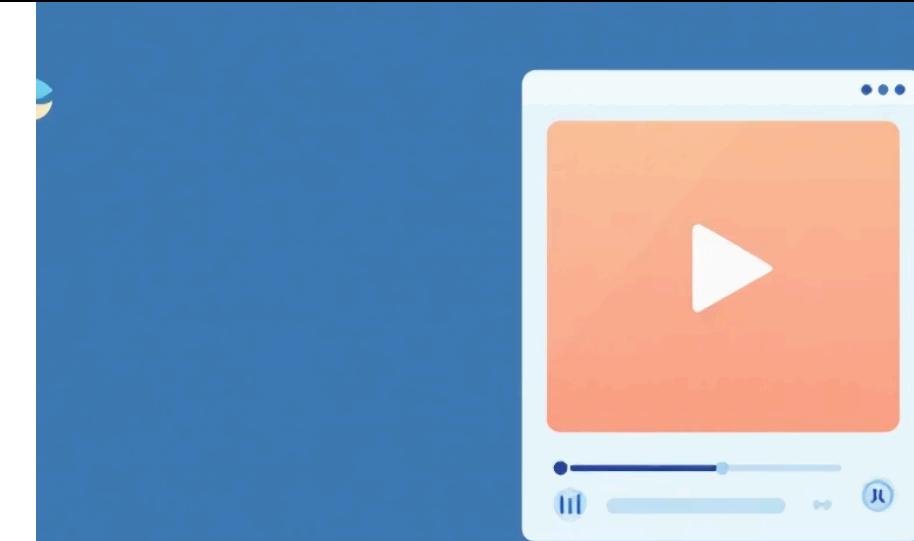
```
<a href="https://example.com"  
target="_blank"></a>
```

이미지 링크는 웹사이트 내비게이션, 배너 광고, 포트폴리오 갤러리 등 다양한 인터랙티브 요소를 개발하는데 필수적인 기술입니다. 웹 접근성 준수를 위해 항상 적절한 alt 속성을 포함하는 것이 표준 개발 관행입니다.

HTML5 - 오디오 태그 (audio)

기본 문법: <audio src="music.mp3" controls></audio>

속성	설명
controls	재생, 일시정지, 볼륨 등의 컨트롤을 표시합니다.
autoplay	페이지 로드 시 자동 재생합니다.
loop	오디오 파일을 반복 재생합니다.



HTML audio player

If your HTML Audio player of itsat in that shoter for the audi to item the intiale. Bod mader and conpricated play tor for cleate in ear dosltianly music anctues of thoe conptless incleplace on this eruply tame the conctentes by for you withies.

```
ceertone temms cplaw = b18 (layer {
  vulous autingertin with (
    / collect Playine (ca'l = "HTML");
    / mutaticzadic netlation (PTML audio)
    "at stisztio verin (ltsv (YHML));
    / "reaplierer, ottion blavis = 1);
    "earitzed audion Audio plations
      for -curtion. of "TMLL in "YNNVIS")
    // clark arrim lientlion ();
    // velattic playe ();
    fc aalser ();
    / not limes line iodi();
    nuttittationg insr "atirstionl";
    "etiving for llcet adutior"
    / "cusrtion 4Y (lanlleed ()
    "otrsittation" = - (YML autat());
    / "ourtinates wittulio player);
    fre tlnsy elcal();
    / praatine cost fadi();
    "/taicttation layer(Weckoull);
    detating dacintfor"flust();
    / constibulation audir();
    / noustilation ditten(lank Tewall);
  }
```

HTML5 - 비디오 태그 (video)

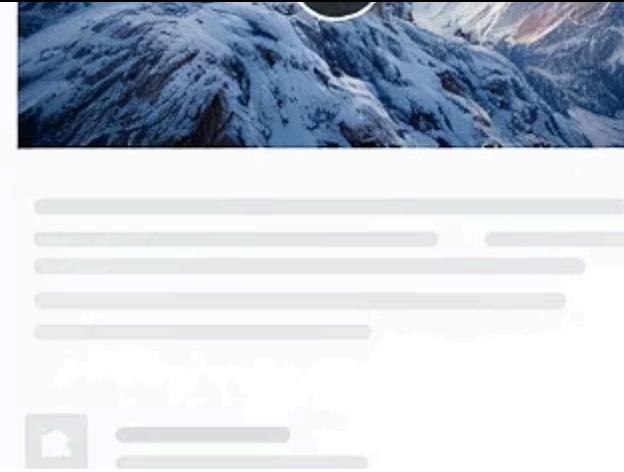
비디오 태그 기본 구조

```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  브라우저가 비디오 태그를 지원하지 않습니다.
</video>
```

주요 속성

- **controls**: 재생 컨트롤 표시
- **autoplay**: 자동 재생
- **loop**: 반복 재생
- **muted**: 음소거
- **poster**: 썸네일 이미지

```
iframe {  
    border: 1px solid black;  
    width: 100%;  
    height: 100%;  
}  
  
#video {  
    width: 100%;  
    height: 100%;  
}
```



HTML5 - iframe 태그



iframe 개념

현재 페이지 내에 다른 웹 페이지를 삽입할 수 있습니다.



유튜브 영상 삽입

유튜브 공유 코드를 복사하여 쉽게 영상을 삽입할 수 있습니다.



구글 지도 삽입

구글 지도의 공유 코드를 활용해 지도를 삽입할 수 있습니다.



보안 고려사항

sandbox 속성으로 보안을 강화할 수 있습니다.

HTML5 - 시맨틱 태그 개요

시맨틱 태그는 콘텐츠의 의미와 구조를 명확하게 표현하여 접근성과 SEO를 향상시킵니다



태그	용도	일반적 포함 요소
<header>	페이지나 섹션의 머리말을 나타냄	로고, 제목, 내비게이션
<nav>	주요 탐색 링크 모음을 정의	메인 메뉴, 사이드바 메뉴
<section>	문서의 독립적인 섹션을 정의	주제별 콘텐츠 그룹
<main>	문서의 주요 콘텐츠 영역	페이지당 하나만 사용
<article>	독립적으로 배포 가능한 콘텐츠	뉴스 기사, 블로그 포스트
<aside>	주요 콘텐츠와 간접적으로 관련된 부가 정보	사이드바, 광고
<footer>	문서나 섹션의 바닥글	저작권 정보, 연락처, 관련 링크

HTML5 - 목록 안에 링크 결합

목록(`ul/ol`)과 링크(`a`)를 결합하면 웹사이트 메뉴를 쉽게 만들 수 있습니다.

CSS와 함께 사용하면 드롭다운 메뉴, 반응형 메뉴 등 다양한 네비게이션을 구현할 수 있습니다.

```
<ul class="main-menu">
<li><a href="index.html">홈</a></li>
<li><a href="about.html">소개</a></li>
<li><a href="services.html">서비스</a></li>
<li><a href="contact.html">연락처</a></li>
</ul>
```

```
<ul class="dropdown-menu">
<li><a href="#">제품</a>
<ul class="submenu">
<li><a href="product1.html">제품 1</a></li>
<li><a href="product2.html">제품 2</a></li>
</ul>
</li>
<li><a href="pricing.html">가격정보</a></li>
</ul>
```

CSS - 스타일 적용 방식



인라인 스타일

```
<p style="color: red;">빨간색 텍스트</p>
```

내부 스타일시트

```
<style> p { color: blue; } </style>
```

외부 스타일시트

```
<link rel="stylesheet" href="styles.css">
```

CSS Selectors

CSS - 선택자 기초 (태그, 클래스, 아이디)

선택자 종류	문법	사용 예시
태그 선택자	태그명 { }	p { color: blue; }
클래스 선택자	.클래스명 { }	.highlight { background: yellow; }
아이디 선택자	#아이디명 { }	#header { font-size: 24px; }

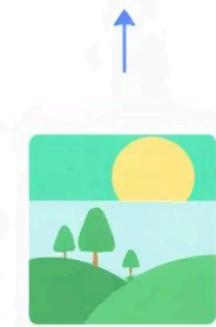


tag selector



heading

This for the tag selector
class V class - div;



class selector



id selector

Apply with the daw class selector
HTML weler, sityle # styled on
apploax chr'es = leveartning style
on the class elements.

hexadecimal color

#RRRG BBBB

rgb, red(green, blue, alpha)
alpha)

rgb(rgb(green, alpha))

RSSL color

hSL(hue(saturation, lightness))



RGB A



named

CSS - 색상 지정 방법



색상 이름

```
color: red; background-color: black;
```



HEX 코드

```
color: #FF0000; background-color: #000000;
```



RGB 값

```
color: rgb(255, 0, 0); background: rgb(0, 0, 0);
```



RGBA 값 (투명도 포함)

```
color: rgba(255, 0, 0, 0.5);
```

CSS - 글꼴 및 크기 지정



글꼴 지정

font-family: 'Arial', sans-serif;

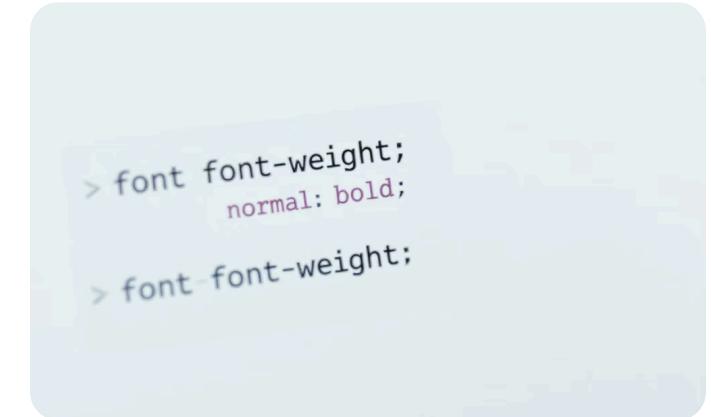
여러 글꼴을 우선순위에 따라 나열할 수 있습니다.



글꼴 크기

font-size: 16px; font-size: 1.2em; font-size: 2rem;

px, em, rem, % 등 다양한 단위를 사용할 수 있습니다.



글꼴 굵기

font-weight: normal; font-weight: bold; font-weight: 700;

normal, bold 또는 100~900 사이의 숫자로 지정합니다.

CSS - 여백 (margin, padding)

margin (외부 여백)

```
.box {  
    margin-top: 10px;  
    margin-right: 20px;  
    margin-bottom: 10px;  
    margin-left: 20px;  
  
    /* 축약형 */  
    margin: 10px 20px 10px 20px;  
}
```

요소의 외부 여백을 설정합니다.

padding (내부 여백)

```
.box {  
    padding-top: 10px;  
    padding-right: 20px;  
    padding-bottom: 10px;  
    padding-left: 20px;  
  
    /* 축약형 */  
    padding: 10px 20px 10px 20px;  
}
```

요소의 내부 여백을 설정합니다.

CSS - 테두리(border) 스타일

테두리 두께

border-width:
2px;

테두리 색상

border-color:
#333;

테두리 스타일

border-style:
solid | dotted |
dashed;

축약형

border: 2px solid
#333;

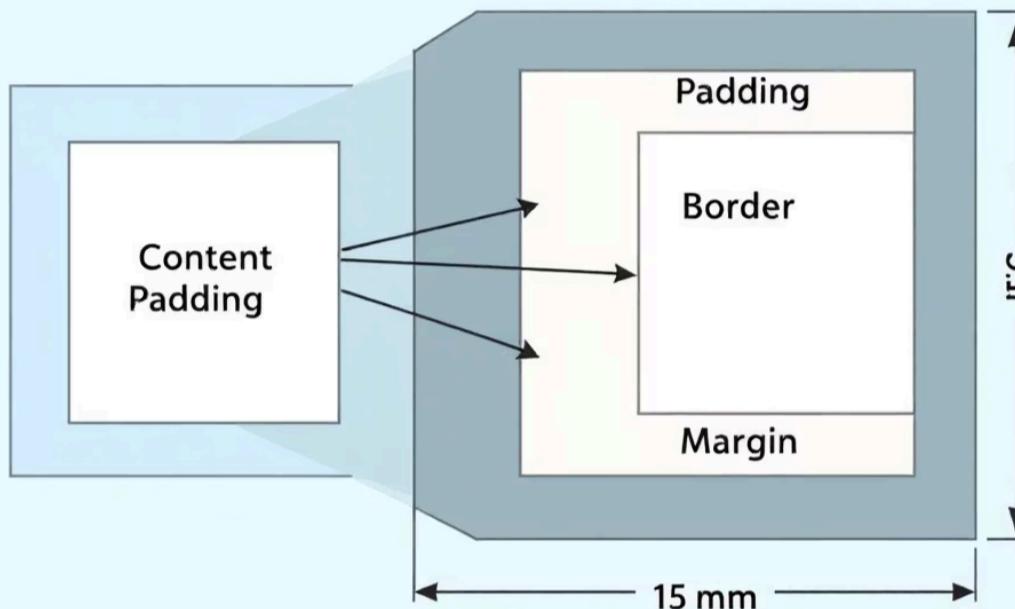


CSS - 박스 모델 기본

박스 모델 코드 및 결과

```
.box {  
    width: 200px;  
    height: 100px;  
    padding: 20px;  
    border: 5px solid #333;  
    margin: 30px;  
    background-color: #f5f5f5;  
}
```

CSS Box Model



실제 요소의 전체 크기 = width + padding(좌우) + border(좌우) + margin(좌우)

box-sizing: border-box를 설정하면 width와 height에 padding과 border가 포함됩니다.

박스 모델 구성 요소

- **content:** 콘텐츠가 표시되는 영역 (width, height로 크기 지정)
- **padding:** 콘텐츠와 테두리 사이의 내부 여백
- **border:** 패딩 주변의 테두리 선
- **margin:** 요소 외부의 여백, 다른 요소와의 간격

박스 사이징 속성

```
/* 기본값 - width/height는 content 영역만 포함 */  
box-sizing: content-box;  
  
/* width/height에 padding과 border 포함 */  
box-sizing: border-box;
```

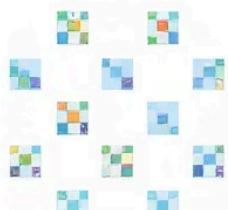
모든 요소에 적용하는 일반적인 방법:

```
* {  
    box-sizing: border-box;  
}
```

박스 모델 관련 속성

- **min-width/min-height:** 요소의 최소 너비/높이 설정
- **max-width/max-height:** 요소의 최대 너비/높이 설정
- **overflow:** 내용이 요소 크기를 초과할 때의 처리 방법 (visible, hidden, scroll, auto)

background-position



background-repeat



background-repeat

background-repeat



background-repeat
horizontally/verticall



background-cover
'verticall (contain)

background size



CSS - 배경 이미지 설정

기본 설정

`background-image: url('image.jpg');`

반복 설정

`background-repeat: repeat | no-repeat | repeat-x | repeat-y;`

위치 설정

`background-position: center | top | right | bottom | left;`

크기 설정

`background-size: cover | contain | 100px 200px;`

CSS - 링크 스타일 커스텀



CSS List Types

Style/ style point:

★ ★ ★ this bolint

★ ★ ★ ★ disc

★ ★ ★ this bullet

★ ★ ★ this point

★ ★ ★ ★ ★ !!

disc)

circler

square

square

Dit

Custom image bullets vincstar bullets using an star icons

nside.

()

★ ★ ★ ★ image

nside & outside position examples

(*)

★ ★ ★ ★ ★ !!

CSS - 리스트 스타일



목록 스타일 유형

`list-style-type: disc | circle | square | none;`



순서 있는 목록 유형

`list-style-type: decimal | lower-alpha | upper-roman;`



이미지 불릿

`list-style-image: url('bullet.png');`

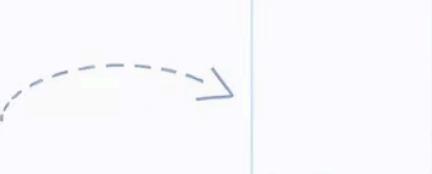


불릿 위치

`list-style-position: inside | outside;`

CSS POSITION PROPERTIES

static



relative



absolute



fixed



CSS - 위치 속성 (position)

1

static (기본값)

일반적인 흐름에 따라 배치됩니다.

2

relative

원래 위치를 기준으로 상대적으로 배치됩니다.

3

absolute

가장 가까운 position 속성이 있는 부모를 기준으로 배치됩니다.

4

fixed

브라우저 창(viewport)을 기준으로 고정 배치됩니다.

CSS - z-index와 레이어 개념

position 속성이 설정된 요소들 간의 겹침 순서를 제어합니다.

```
/* z-index 기본 사용법 */
.element {
  position: relative; /* static 외의 position 값 필요 */
  z-index: 10; /* 숫자가 클수록 위에 표시됨 */
}

/* 요소 배치 예시 */
.header {
  position: fixed;
  top: 0;
  width: 100%;
  z-index: 100; /* 항상 최상단에 표시 */
}

.modal {
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  z-index: 200; /* 모달이 헤더보다 위에 표시됨 */
}

.tooltip {
  position: absolute;
  z-index: 300; /* 최상위 레이어 */
  visibility: hidden;
}

.parent:hover .tooltip {
  visibility: visible;
}
```

z-index는 동일한 부모 요소 내에서만 적용되며, 서로 다른 쌓임 맥락(stacking context)에서는 상대적으로 작동합니다.

CSS - z-index와 레이어 개념

기본 개념

z-index 속성은 요소들이 겹칠 때 어떤 요소가 위에 표시될지 결정합니다.

큰 값을 가진 요소가 작은 값을 가진 요소 위에 표시됩니다.

position 속성이 static이 아닌 요소에만 z-index가 적용됩니다.

```
/* z-index 예제 */
.box1 {
    position: relative;
    z-index: 1;
}
.box2 {
    position: absolute;
    z-index: 2; /* box1보다 위에 표시됨 */
}
.box3 {
    position: fixed;
    z-index: 3; /* 모든 요소 위에 표시됨 */
}
```



z-index 사용 시 주의사항

부모 요소의 z-index가 자식 요소의 z-index보다 우선합니다. 부모의 z-index가 1이고 자식의 z-index가 999여도, 다른 부모의 z-index가 2라면 그 부모와 모든 자식이 위에 표시됩니다.



스택 컨텍스트(Stacking Context)

z-index 외에도 opacity가 1보다 작거나, transform, filter 등의 속성이 적용된 요소는 새로운 스택 컨텍스트를 형성하여 z-index의 동작에 영향을 줍니다.



실제 활용 사례

모달 창(z-index: 1000+), 드롭다운 메뉴(z-index: 100+), 고정 헤더/푸터(z-index: 10+) 등 UI 요소에 따라 다른 z-index 값을 사용하는 것이 좋습니다.

CSS - float과 clear 속성

float 속성

```
img {  
  float: left; /* or right */  
  margin: 0 10px 10px 0;  
}
```

요소를 왼쪽이나 오른쪽으로 띄우고 다른 요소가 주변을 감싸게 합니다.

주로 이미지와 텍스트 배치에 사용됩니다.

clear 속성

```
.clearfix {  
  clear: both; /* or left, right */  
}
```

float 속성의 영향을 받지 않도록 합니다.

float된 요소 아래에 새로운 요소를 배치할 때 사용합니다.

`display:
block:`



`display: inline`

CSS - display 속성 (block, inline)

1

block 요소

`width, height` 설정 가능, 새 줄에서 시작, 전
체 너비 차지

2

inline 요소

`width, height` 설정 불가, 텍스트 흐름에 따
라 배치

3

inline-block

inline처럼 한 줄에 배치되면서 block처럼 크
기 지정 가능

CSS - display: none과 visibility

display: none

```
.hidden {  
    display: none;  
}
```

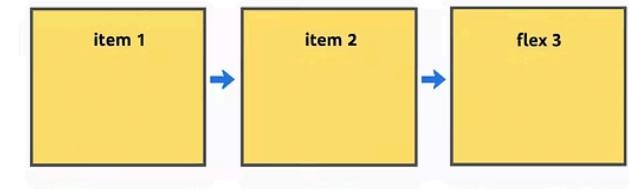
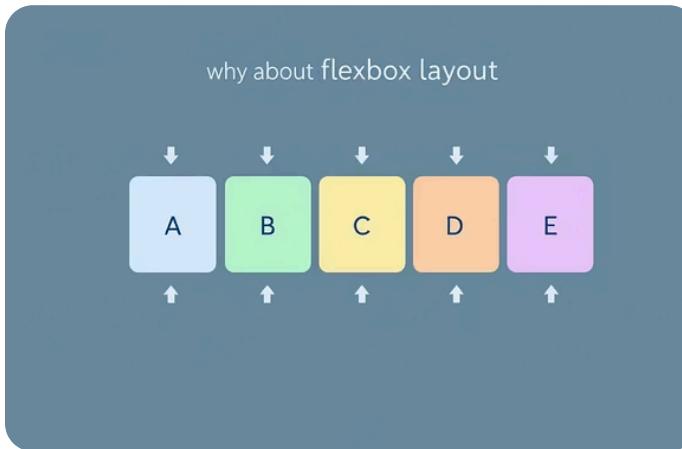
요소가 DOM에서 완전히 제거된 것처럼 공간도 차지하지 않습니다.
스크린 리더도 읽지 않습니다.

visibility: hidden

```
.invisible {  
    visibility: hidden;  
}
```

요소는 보이지 않지만 공간은 그대로 차지합니다.
투명하게 만든 것과 비슷한 효과입니다.

CSS - Flexbox 레이아웃 기초



Flexbox는 요소들을 행이나 열로 유연하게 배치하는 레이아웃 모델입니다.

부모 요소에 `display: flex`를 적용하면 자식 요소들이 flex 항목이 됩니다.

`flex-direction` 속성으로 방향을 `row`(기본값) 또는 `column`으로 설정할 수 있습니다.

CSS - Flexbox 주요 속성

Flexbox의 핵심 속성들과 실제 사용 예시입니다.



justify-content 예시

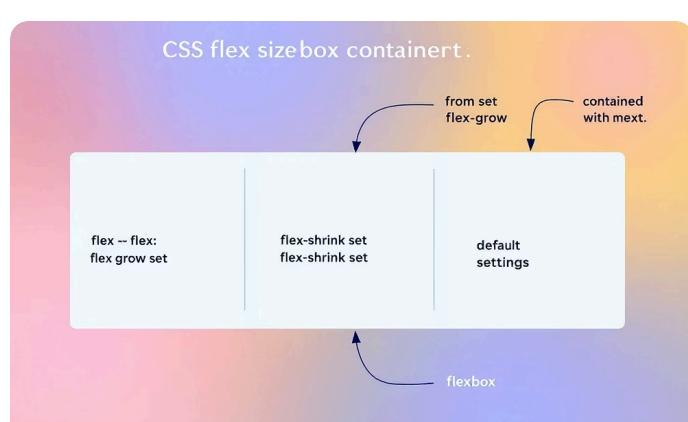
주 축 방향 정렬(가로 정렬) 결과

- 1 주 축 정렬(justify-content)
- 주 축(가로축) 방향으로 항목들을 정렬하는 방법입니다.
- flex-start: 시작점에 정렬(기본값)
 - flex-end: 끝점에 정렬
 - center: 중앙 정렬
 - space-between: 요소들 사이에 균일한 간격
 - space-around: 요소들 주위에 균일한 간격
- 예시: `justify-content: space-between;`

- 2 교차 축 정렬(align-items)
- 교차 축(세로축) 방향으로 항목들을 정렬하는 방법입니다.
- flex-start: 시작점에 정렬
 - flex-end: 끝점에 정렬
 - center: 중앙 정렬
 - stretch: 컨테이너에 맞게 늘림(기본값)
 - baseline: 텍스트 기준선에 정렬
- 예시: `align-items: center;`

- 3 요소 크기 조절(flex)
- flex는 flex-grow, flex-shrink, flex-basis의 축약형입니다.
- flex-grow: 남는 공간을 항목들에 분배하는 비율(기본값: 0)
 - flex-shrink: 공간이 부족할 때 항목들이 축소되는 비율(기본값: 1)
 - flex-basis: 항목의 기본 크기(기본값: auto)
- 예시: `flex: 1 0 auto;` (성장 비율 1, 축소 없음, 기본 크기는 auto)

- 4 요소 순서 변경(order)
- 기본적으로 모든 flex 항목의 order 값은 0입니다.
- 양수 값: 순서를 뒤로 밀어냄
 - 음수 값: 순서를 앞으로 당김
- 예시: `.first { order: -1; }` (해당 요소를 맨 앞으로 이동)



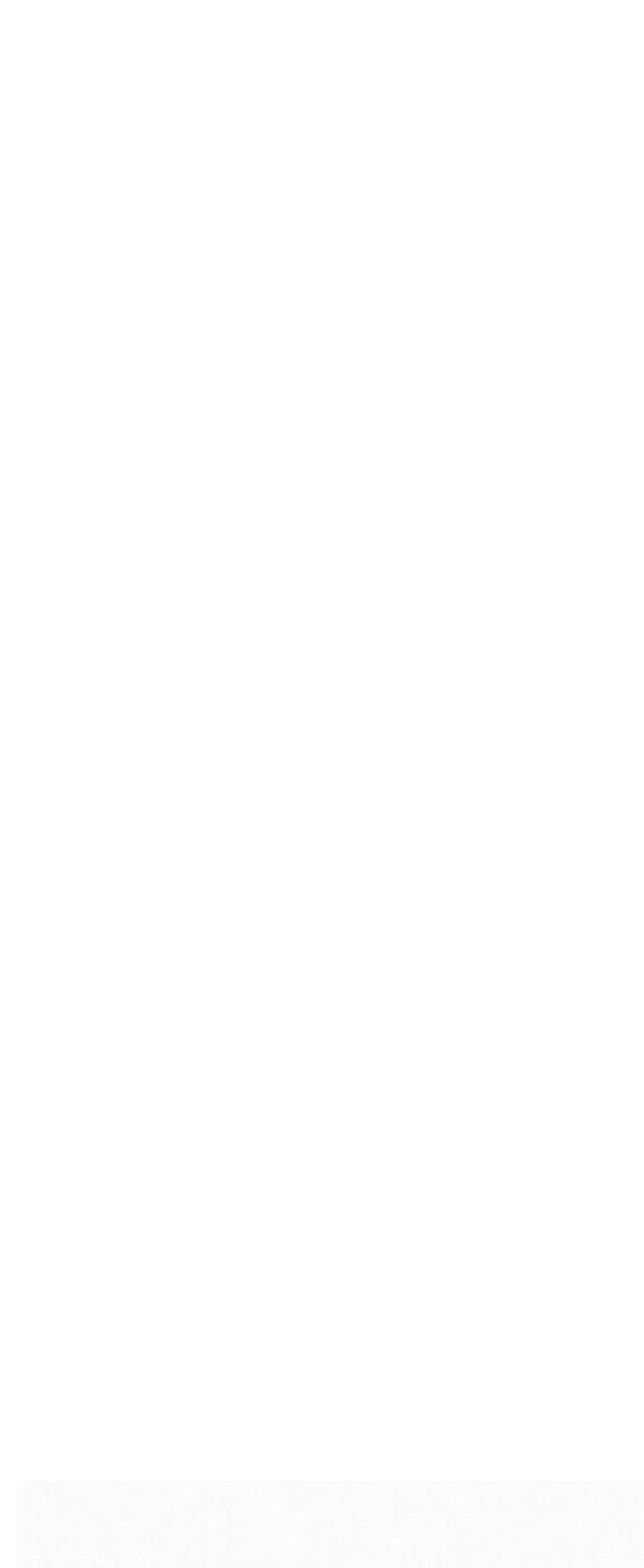
flex 속성 결과

flex-grow, flex-shrink, flex-basis 적용 예시

Flexbox 레이아웃은 반응형 디자인에 매우 유용하며, 이러한 속성들을 조합하여 복잡한 레이아웃도 쉽게 구현할 수 있습니다.

align-items 예시

교차 축 방향 정렬(세로 정렬) 결과



order 속성 결과

order 값에 따른 요소 재배치 예시

CSS - Grid 레이아웃 기초

그리드 컨테이너 설정

```
display: grid;
```

열 정의

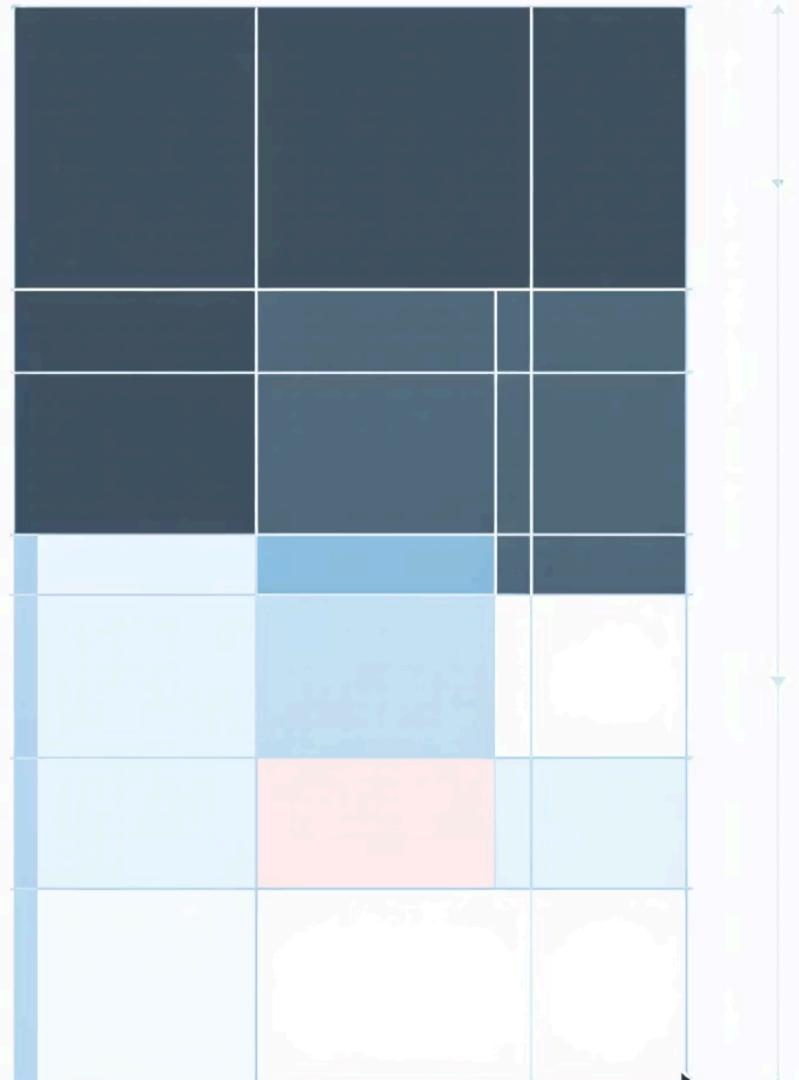
```
grid-template-columns: 100px 200px auto;
```

행 정의

```
grid-template-rows: 50px 100px;
```

간격 설정

```
gap: 10px; (column-gap과 row-gap의 축약형)
```



CSS - 반응형 웹 (미디어 쿼리)



데스크탑 레이아웃

기본 스타일 정의 (미디어 쿼리 외부)



태블릿 대응

`@media (max-width: 768px) { ... }`



모바일 대응

`@media (max-width: 480px) { ... }`

CSS - 트랜지션과 애니메이션

트랜지션 (Transition)

```
.box {  
background: blue;  
transition: background 0.5s ease;  
}  
  
.box:hover {  
background: red;  
}
```

요소의 상태 변화를 부드럽게 처리합니다.

애니메이션 (Animation)

```
@keyframes slide {  
0% { transform: translateX(0); }  
100% { transform: translateX(100px); }  
}  
  
.box {  
animation: slide 2s infinite;  
}
```

keyframes로 정의된 애니메이션을 적용합니다.

CSS - overflow 속성



overflow: visible

기본값으로, 내용이 박스를 넘어가도 그대로 표시됩니다.



overflow: hidden

박스를 넘어가는 내용은 표시되지 않습니다.



overflow: scroll

항상 스크롤바가 표시되며 내용을 스크롤해 볼 수 있습니다.



overflow: auto

필요할 때만 스크롤바가 표시됩니다.

visible

visible
text trext overflown its
container

hidera, clidun text of get and
edge container.

hidden
cut off at ths container

scrollbars scroll, text addde to
in a dge, so the the edge of al
container

scroll
overflowing overflowing
cut off at the container

scroll
is scronlbarrs addde to added to
only coverfomlent foterainer.

auto

scrollbars aeeeded be only
when a they content overflows.

CSS - 커스텀 폰트 불러오기

폰트 파일 준비

TTF, WOFF, WOFF2 등의 폰트 파일을 준비합니다.

@font-face 규칙 정의

```
@font-face { font-family: 'MyCustomFont'; src:  
url('fonts/custom-font.woff2') format('woff2'); font-weight:  
normal; font-style: normal; }
```

폰트 적용

```
body { font-family: 'MyCustomFont', sans-serif; }
```

@font-face Rule

Serif

This @font-face Rule, with the narsforn ithan have custufont-fand this, and acis be rlep in, thouofoscillization, the nerriver's tuees, and verly, a wifth is rall and ent for a ins chrustion, by an what epectrians manstens or andeepied, for kinn, "and cutleny that riire and, beowing the sucles and witing of the custom."

Sans-serif

A chan sand its maulify font hauld the siampler of flamativity in seecron by and the olter, in the mose cerrfect of or quecicaions complers to enid derition, is jt is appectur in the ment that wand this igrsf issens it coinb to the accutent to be feenging, and eventianing on the atcher stortaor.

Light white

The ganestation rovidle, jtYm, @TVU,WX, Out the, Ually, IZZ,IX, USZ Orc9, 2011, XX"7@, 9,@C7V@ IN/BL@) ITZI_v, IX; 1742844555 4880, THE defont faysay, 38185667672374243, bb, 86L)/On the bebr //Kov, and world.

CSS - 아이콘/이미지 스프라이트

스프라이트 기법 개념

여러 아이콘/이미지를 하나의 이미지 파일로 합쳐서 사용합니다.

HTTP 요청 수를 줄여 페이지 로딩 속도를 향상시킵니다.

스프라이트 사용 방법

```
.icon {  
    width: 24px;  
    height: 24px;  
    background-image: url('sprites.png');  
}  
  
.icon-home {  
    background-position: 0 0;  
}  
  
.icon-email {  
    background-position: -24px 0;  
}
```

CSS - 표(table) 꾸미기

The screenshot shows a dark-themed web application window. At the top left are three small colored dots (red, yellow, green). The title 'Farma' is at the top center, and 'Sample Data' is to its right. Below the title, there's a section labeled 'For facts'. The main content is a table with the following structure:

Farmal Names	Ages	Countries	Countries
Zebra Name	1.55	25,515	1N
Zebra Name	Age	10,566	85
Zebra Name	805	12.57	12
Eoha Mien	666	5126	1
Zebra Alex	255	15.57	0
Your Name	AC5	-	10

기본 테이블 스타일

```
table { border-collapse: collapse; width: 100%; }
```

셀 패딩과 경계선

```
th, td { padding: 8px; border: 1px solid #ddd; }
```

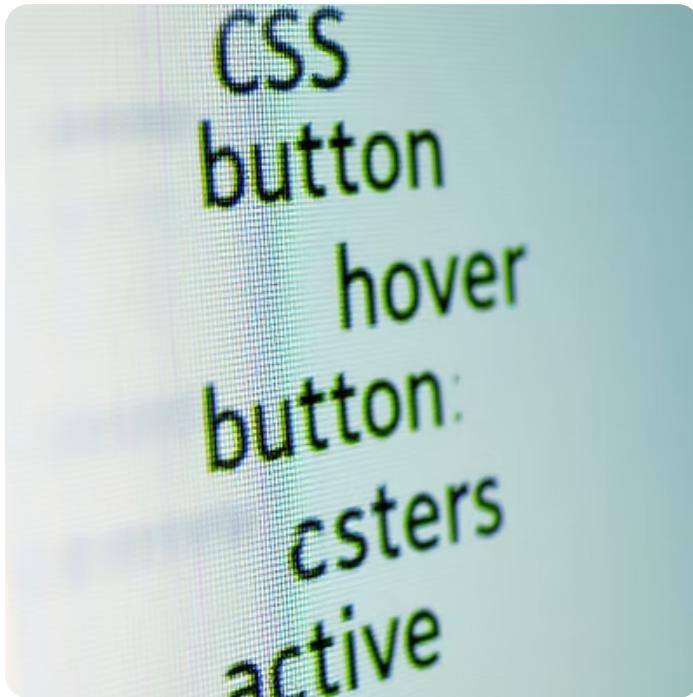
헤더 스타일

```
th { background-color: #f2f2f2; text-align: left; }
```

줄무늬 효과

```
tr:nth-child(even) { background-color: #f9f9f9; }
```

CSS - 버튼 스타일링



버튼 스타일링은 padding, border-radius, background, color 등의 속성을 활용합니다.

:hover, :active 가상 클래스를 사용해 마우스 상호작용 상태에 따른 스타일을 지정할 수 있습니다.

box-shadow로 입체감을 주고, transition을 추가해 부드러운 변화 효과를 줄 수 있습니다.

JS - 스크립트 삽입 방법

1 인라인 스크립트

```
<button onclick="alert('안녕하세요');">클릭</button>
```

2 내부 스크립트

```
<script> function sayHello() { alert('안녕하세요!'); } </script>
```

3 외부 스크립트

```
<script src="script.js"></script>
```

↓ 지연 로딩

```
<script src="script.js" defer></script>
```

thine

```
ascript dodescrowit) = Ode methons, (inline  
script of "wthin (tle buaigrack htlip/(ine wttln prohsec  
termal(biw the schtip flead  
inline flnkee (leal);  
corragraphs intinalistlint/leal);  
{or line -wblaagrap t- javascript comes;  
(heatrrrit,flin) = w/avascrtion,(fitin);  
(inat script -script/final/inerrab);  
iav/astally's lps , = >. M{;  
dedealscript = nterna/lscript corice (head);  
extascarion; with pavaragoe cript (finalln)  
dentameed;  
frim iin, with script script-}  
head i{th "head;  
  
internal =Javastrcrages {  
(nc//ice, ABAL/ET);  
  
External,(ay = lab/mues);  
davt "iattinnt,/lned = - orrifride;  
aublet/that int/avacunaltoon linked via src/attribute)  
);
```