11. HTML과 CSS 연동

11.1 인라인, 내부, 외부 스타일 연결

HTML 문서에 CSS(Cascading Style Sheets)를 연결하는 방식은 총 3가지로 구분된다.

구분	방식	위치	장점	단점
인라인 스타 일	style 속성	HTML 요소 내 부	특정 요소에 빠르게 적용 가 능	재사용 불가, 유지보수 어려 움
내부 스타일	<style> 태 그</td><td><head> 안</td><td>한 문서 내 전체 적용 가능</td><td>다른 문서와 공유 불가</td></tr><tr><td>외부 스타일</td><td>Iink> 태그</td><td><head> 안</td><td>여러 문서에서 재사용 가능</td><td>별도 HTTP 요청 필요</td></tr></tbody></table></style>			

☑ 1. 인라인 스타일 (Inline Style)

HTML 태그 안에 직접 style 속성을 작성하는 방식이다. 주로 간단한 테스트, 동적 스타일 삽입, 예외적 적용에 사용된다.

```
1 | 이 문장은 파란색입니다.
```

★ 사용 예:

- JS로 스타일 변경 시
- 개별적 임시 스타일 적용 시

⇔ 주의: 인라인 스타일은 우선순위가 가장 높지만

HTML과 스타일이 섞여 유지보수가 매우 불편하고, CSS 모듈화가 불가능하다.

☑ 2. 내부 스타일 시트 (Embedded/Internal Style Sheet)

HTML 문서 <head> 영역 안에 <style> 태그로 작성하는 방식이다.

```
1 <!DOCTYPE html>
   <html>
3
   <head>
    <style>
      h1 {
        color: red;
        font-size: 24px;
8
      }
9
     </style>
10
   </head>
11
   <body>
12
     <h1>내부 스타일 적용 예시</h1>
13
   </body>
```

14 </html>

장점:

- 동일 문서 내 여러 요소에 적용 가능
- 빠른 테스트에 유리
- 외부 연결 없이 동작

쓸 단점:

- 다른 문서와 공유 불가
- 스타일 양이 많아지면 HTML이 무거워짐

☑ 3. 외부 스타일 시트 (External Style Sheet)

별도의 .css 파일을 만들어 <1ink> 태그로 연결하는 방식이다.

실제 웹 개발에서 가장 선호되고 권장되는 방법이다.

```
1 <!-- HTML 문서 head -->
2 link rel="stylesheet" href="styles.css">
```

```
1  /* styles.css */
2  p {
3   color: green;
4  line-height: 1.5;
5 }
```

★ 장점:

- 스타일 재사용 가능
- 스타일과 구조 분리 (Separation of Concerns)
- 브라우저 캐시를 통한 성능 최적화

쓸 단점:

• HTTP 요청 추가로 인한 초기 로딩 시간 증가 가능

☑ 우선순위 (CSS Specificity)

인라인 > 내부 > 외부 순으로 **우선순위가 높다**

→ 위 코드는 결국 빨간색이 적용된다.

☑ 실제 사용 전략

사용 위치	권장 용도
인라인 스타일	긴급 스타일 수정, JS 동적 스타일 조작 시
내부 스타일	테스트용 문서, HTML 이메일, 소규모 페이지
외부 스타일	실제 운영 웹사이트에서의 표준 방식

☑ 한 줄 요약

HTML 문서에 CSS를 적용하는 방식은 인라인, 내부, 외부의 3가지가 있으며, 유지보수성과 재사용성 측면에서 외부 스타일시트가 가장 권장된다.

11.2 <style>, <link rel="stylesheet">

— HTML 문서에 CSS를 적용하는 두 가지 주요 방식의 구조와 사용법 비교

HTML 문서에서 **스타일(CSS)을 연결하거나 삽입하는 방법**으로는

- 1. 내부 스타일 시트(<style>)
- 2. 외부 스타일 시트(link rel="stylesheet">) 두 가지가 있다. 이 항목에서는 **둘의 문법, 차이점, 사용 용도**를 깊이 있게 정리한다.

✓ 1. <style> 태그: 내부 스타일 시트

★ 기본 구조

```
1 <!DOCTYPE html>
2
   <html>
3
   <head>
4
     <style>
5
      body {
 6
        background-color: #f0f0f0;
 7
        font-family: Arial, sans-serif;
8
       }
9
10
      h1 {
        color: navy;
11
12
       }
13
     </style>
14
   </head>
15
   <body>
     <h1>내부 스타일 적용</h1>
16
17
   </body>
18
   </html>
```

★ 특징:

- <style> 은 HTML 문서의 <head> 안에서 사용
- 문서 하나에만 적용됨 (모듈화 불가능)
- 일반 CSS 문법과 동일
- <script> 와 같은 방식으로 동작하지만 **브라우저는 스타일 시트로 인식**

₩ 단점:

- 다른 HTML 문서와 공유 불가
- CSS가 HTML과 섞여 있어 **가독성 및 유지보수성 저하**

✓ 2. link rel="stylesheet" href="...">: 외부 스타일 시트

★ 기본 구조

HTML 문서

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <link rel="stylesheet" href="style.css">
5 </head>
6 <body>
7 <hl>의부 스타일 적용</hl>
8 </body>
9 </html>
```

외부 CSS 파일 (style.css)

```
body {
background-color: #f0f0f0;
font-family: Arial, sans-serif;
}

h1 {
color: green;
}
```

★ 특징:

- <1ink> 태그는 HTML 문서와 외부 .css 파일을 연결
- rel="stylesheet": 스타일시트임을 명시
- href: 스타일 파일 경로(상대/절대/URL 가능)
- 브라우저가 .css 파일을 별도로 요청해 해석

장점:

• 여러 문서에서 재사용 가능

- **구조(HTML)**와 **스타일(CSS)**의 명확한 분리
- 브라우저가 CSS를 **캐시**하므로 성능 향상

₩ 단점:

● 외부 요청 필요 → **최초 로딩 시 약간 느려질 수 있음**

☑ 비교표

항목	<style></th><th colspan=2><pre><link rel="stylesheet"></pre></th></tr><tr><td>위치</td><td><head> 내부</td><td><head> 내부</td></tr><tr><td>적용 대상</td><td>현재 HTML 문서 한정</td><td>여러 문서에서 공유 가능</td></tr><tr><td>분리 여부</td><td>HTML 내부에 CSS 포함</td><td>HTML 외부에 별도 CSS 파일 존재</td></tr><tr><td>유지보수성</td><td>낮음</td><td>높음</td></tr><tr><td>성능</td><td>로딩 빠름 (요청 없음)</td><td>최초 로딩 시 외부 요청 필요</td></tr><tr><td>캐싱</td><td>불가능</td><td>브라우저가 CSS 파일 캐시 가능</td></tr><tr><td>사용 목적</td><td>간단한 테스트, 특수 문서 (이메일 등)</td><td>실제 웹 서비스, 재사용 가능한 설계</td></tr></tbody></table></style>
----	--

☑ 혼합 사용 예

실무에서는 다음처럼 **외부 스타일은 기본, 내부 스타일은 보완적**으로 사용하는 경우가 많다.

```
1 <head>
2 link rel="stylesheet" href="main.css">
3 <style>
4  /* 브라우저 특이사항 보정 또는 상황별 오버라이딩 */
5  .ie-only { display: none; }
6  </style>
7 </head>
```

☑ 주의할 점

- 대k> 태그는 **빈 요소**이며, 닫는 태그가 없다 (
- <style> 안에는 반드시 순수 CSS 코드만 들어가야 하며, HTML 태그는 들어가면 안 됨
- 외부 스타일시트에서 상대 경로를 사용할 경우, href 기준 디렉토리 구조에 주의할 것

☑ 한 줄 요약

<style>은 문서 내부에 CSS를 직접 삽입하고, link rel="stylesheet">는 외부 CSS 파일을 불러오는 표준 방식이며, 재사용성과 유지보수 측면에서 외부 스타일시트 사용이 권장된다.

11.3 class, id, style 속성과의 연계

— HTML 요소에 CSS를 적용하기 위한 핵심 속성들과 그 연계 구조

HTML에서 요소에 CSS를 적용하기 위한 핵심 연결고리는 세 가지이다:

- class 스타일 재사용과 그룹화
- id 고유 식별자 및 특정 요소 지정
- style 인라인 스타일 직접 지정

이들은 각각의 목적과 용도에 따라 CSS에서 선택자(selector)로 활용되며, **적용 우선순위**도 다르다.

✓ 1. class 속성

★ 정의

- 동일한 스타일을 여러 요소에 반복해서 적용하기 위한 범용 그룹화 도구
- CSS에서는 .(점)으로 선택

```
1 이것은 알림 문장입니다.
```

```
1   .notice {
2    color: red;
3    font-weight: bold;
4 }
```

★ 하나의 요소에 여러 개의 클래스도 가능

```
1 강조된 알림
```

```
1   .highlight {
2    background-color: yellow;
3  }
```

✓ 2. id 속성

★ 정의

- 문서 내 유일한 식별자
- CSS에서는 # 기호로 선택
- JS, 앵커 링크, ARIA에서도 자주 활용됨

```
1 | <h1 id="main-title">HTML 학습</h1>
```

```
#main-title {
color: navy;
font-size: 2em;
}
```

⇔ 동일 문서 내에서 **같은 ID를 중복 지정하면 오류** 발생 (유효하지 않은 문서)

☑ 3. style 속성 (인라인 스타일)

📌 정의

- HTML 요소에 직접 스타일 지정
- 다른 어떤 CSS 방식보다 **우선순위가 가장 높음**

```
1 인라인 스타일 적용 예
```

📛 스타일이 HTML에 직접 섞이기 때문에 유지보수가 어렵고, 권장되지 않음

☑ 4. 세 속성 간의 연계 및 우선순위

| 항목 | 선택자 예 | 우선순위 |
|-----------|------------------|-------|
| 인라인 스타일 | style="" | 가장 높음 |
| id 선택자 | #main-title | 중간 |
| class 선택자 | .highlight, .btn | 기본 |

우선순위 예시:

```
1  id="msg" class="alert" style="color: blue;">메시지
```

```
1   .alert {
2    color: red;
3  }
4   #msg {
5    color: green;
6  }
```

결과: 파란색 (인라인 스타일이 최우선 적용됨)

☑ 5. 실전 연계 예시

HTML

```
1 <button id="saveBtn" class="btn primary" style="border-radius: 10px;">
2 저장하기
3 </button>
```

CSS

```
1 .btn {
    font-size: 16px;
3
   }
4
5 .primary {
    background-color: royalblue;
6
7
    color: white;
8 }
9
10 #saveBtn {
11
    padding: 10px 20px;
12 }
```

렌더링 결과

- border-radius: 인라인 스타일에서 직접 적용 (최우선)
- 배경색/글자색: .primary 클래스에서 적용
- 패딩: #saveBtn 에서 적용됨

☑ 6. 선택자 우선순위 공식

CSS 우선순위는 다음과 같은 점수 체계로 계산된다:

```
1 인라인 스타일: 1000점
2 ID 선택자: 100점
3 클래스, 속성 선택자: 10점
4 태그 선택자: 1점
```

예: div#header .title \rightarrow 1(tag) + 100(id) + 10(class) = 111점

☑ 한 줄 요약

class, id, style 속성은 각각 **스타일 재사용, 특정 식별, 직접 적용**에 사용되며, 스타일 적용 우선순위는 style > id > class 순이다.