

11. 📖 getElementByClassName – 클래스 기반 다중 선택의 시작

1. 하나가 아닌 여러 개의 요소를 선택하고 싶다면?

지금까지 우리는 DOM에서 특정 `id` 값을 가진 요소를 하나 선택할 수 있는 `getElementById()` 에 대해 배웠습니다.

하지만 실제 웹 페이지를 만들다 보면 하나의 요소만 조작하는 것보다, 같은 성격을 가진 여러 요소를 한꺼번에 선택해서 조작해야 할 때가 훨씬 많습니다.

→ 예: 목록 항목 전체, 버튼 여러 개, 댓글 줄마다 스타일 변경 등

이럴 때 사용하는 메서드가 바로 `getElementsByClassName()` 입니다.

2. 메서드 이름만 봐도 기능이 보인다

`getElementsByClassName()` 이라는 이름을 차례대로 분석해 봅시다:

- **get** → 가져온다
- **Elements** → 여러 개의 요소들
- **ByClassName** → `class` 속성값을 기준으로

→ 즉, 같은 `class` 이름을 공유하는 모든 요소들을 한 번에 선택하는 기능입니다.

3. 예제 HTML: 공통 class를 가진 목록

다음은 HTML 예시입니다:

```
<ul>
  <li class="menu-item">홈</li>
  <li class="menu-item">소개</li>
  <li class="menu-item">문의</li>
</ul>
```

→ 세 개의 `` 요소가 모두 `class="menu-item"` 이라는 공통 클래스를 갖고 있습니다.

자바스크립트에서는 이렇게 선택합니다:

```
const menuItems = document.getElementsByClassName("menu-item");
```

4. 반환값은 HTMLCollection 객체

이 메서드는 일반 배열이 아니라 **HTMLCollection**이라는 유사 배열 객체를 반환합니다.

→ 배열처럼 인덱스로 접근할 수 있고 `.length`도 사용 가능하지만

→ `.forEach()` 같은 배열 메서드는 사용할 수 없습니다.

예를 들어 다음처럼 사용할 수 있습니다:

```
console.log(menuItems.length); // 3
console.log(menuItems[0].innerText); // "홈"
```

5. 반복문으로 모든 요소 조작하기

HTMLCollection은 여러 요소를 담고 있으므로, 보통은 반복문을 통해 하나씩 접근하여 조작합니다:

```
for (let i = 0; i < menuItems.length; i++) {
  menuItems[i].style.color = "red";
}
```

→ 이 코드를 실행하면 모든 메뉴 항목이 **빨간색**으로 표시됩니다.

6. 비유로 이해하기: 집단 호출 방식

이 메서드는 `getElementById()`와는 개념적으로 다릅니다.

비교 항목	설명
<code>getElementById()</code>	"철수 한 명만 불러와!" - 고유한 한 명 선택
<code>getElementsByClassName()</code>	"운동부 전원 불러와!" - 같은 속성 가진 집단 선택

→ 즉, 특정 공통 class를 공유하는 여러 요소를 동시에 불러오는 방식입니다.

7. 실습 예제: `class-test.html` 만들기

다음 코드를 메모장에 저장해 직접 실행해보세요:

```
<!DOCTYPE html>
<html>
<head>
  <title>getElementsByClassName 실습</title>
</head>
<body>
  <ul>
    <li class="menu-item">홈</li>
    <li class="menu-item">소개</li>
    <li class="menu-item">문의</li>
  </ul>
  <script>
    const items = document.getElementsByClassName("menu-item");
    for (let i = 0; i < items.length; i++) {
      items[i].style.fontWeight = "bold";
      items[i].style.color = "blue";
    }
  </script>
</body>
</html>
```

→ 실행 결과:

- 메뉴 항목 3개 모두 글씨가 **진해지고**,
- 색상이 **파란색**으로 변경됩니다.

→ DOM 조작을 통해 여러 요소에 동시에 스타일을 적용하는 실제 사례입니다.

8. 여러 class 중 하나만 일치해도 선택된다

다음 코드를 보세요:

```
<li class="menu-item active">홈</li>
```

이 요소는 `menu-item` 과 `active` 두 가지 class를 가지고 있습니다.

→ 하지만 다음과 같이 호출하면 여전히 선택됩니다:

```
document.getElementsByClassName("menu-item");
```

→ 왜냐하면 **"menu-item"이라는 class를 포함하고 있기** 때문입니다.

→ `class="menu-item active"` 처럼 class가 여러 개여도, **하나만 일치하면 선택됩니다.**

✓ 핵심 요약

- `getElementsByClassName()` 는 같은 class 이름을 가진 모든 요소를 한꺼번에 선택합니다.
- 반환되는 값은 **HTMLCollection**, 즉 유사 배열 객체입니다.
- 반복문을 통해 요소마다 스타일 변경, 텍스트 변경 등이 가능합니다.
- class가 여러 개인 경우에도 **지정한 class를 포함하기만 하면 선택됩니다.**

→ 하나가 아닌 **여러 요소를 다루고 싶을 때 가장 기본적인 선택 도구**입니다.