

JAVA 웹 개발자 양성과정

Javascript: Web api

2강 - DOM제어와 조작

By SoonGu Hong

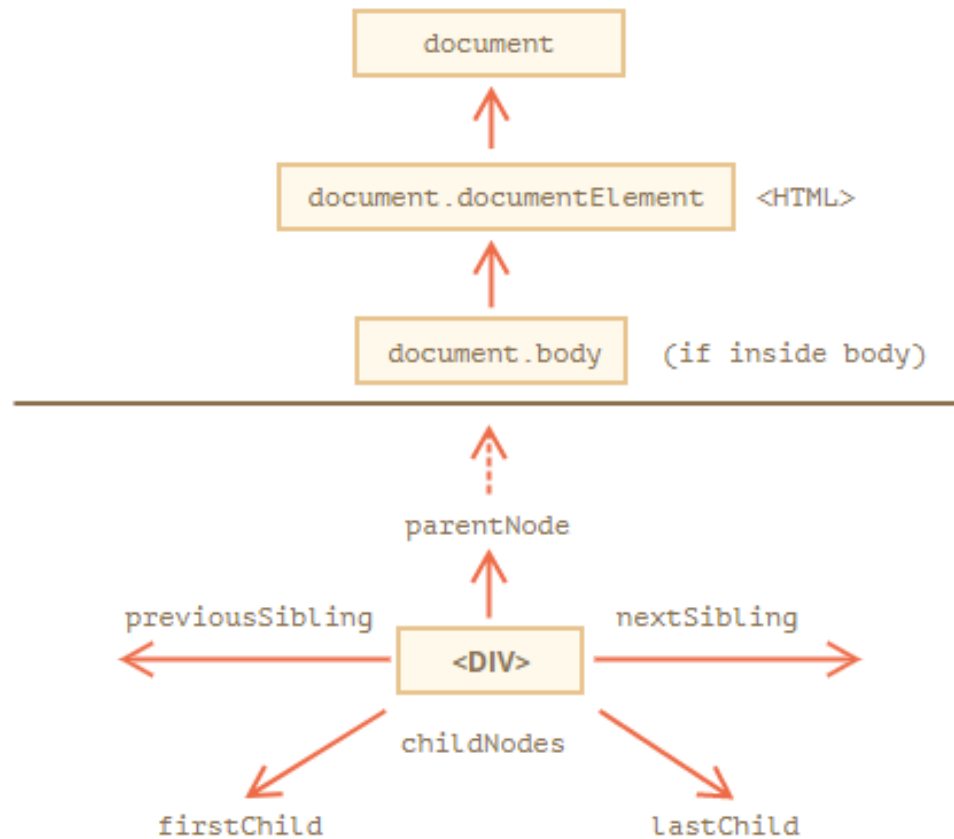
JAVA 웹 개발자 양성과정

Javascript

1. DOM 제어



* 노드 탐색



* 자식 노드 탐색

프로퍼티	설명
childNodes	자식 노드를 모두 탐색하여 NodeList에 담아 반환. 요소 노드 뿐만 아니라 텍스트 노드도 포함함.
children	자식 노드 중 요소 노드만 모두 탐색하여 HTMLCollection객체에 담아 반환. 텍스트 노드가 포함되지 않음.
firstChild	첫 번째 자식 노드를 반환
lastChild	마지막 자식 노드를 반환
firstElementChild	첫 번째 자식 요소 노드를 반환
lastElementChild	마지막 자식 요소 노드를 반환
hasChildNodes()	자식 요소를 가지고 있는지 확인

* 부모, 형제 노드 탐색

프로퍼티	설명
parentNode	부모 노드를 모두 탐색하여 NodeList로 반환.
previousSibling	같은 부모를 가진 형제 노드 중에서 자신의 이전 형제 노드를 반환.
nextSibling	같은 부모를 가진 형제 노드 중에서 자신의 다음 형제 노드를 반환.
previousElementSibling	같은 부모를 가진 형제 노드 중에서 자신의 이전 형제 요소 노드를 반환.
nextElementSibling	같은 부모를 가진 형제 노드 중에서 자신의 다음 형제 요소 노드를 반환.

\$elementNode.textContent

- 요소 노드의 textContent 프로퍼티를 참조하면 요소 내부의 모든 텍스트를 반환합니다.
- 이 때 마크업은 무시됩니다.
- 과거에 사용하던 innerText는 유사한 원리로 작동하나 속도가 느리고 css에 의해 숨겨진 텍스트를 반환하지 않으므로 사용하지 마세요!

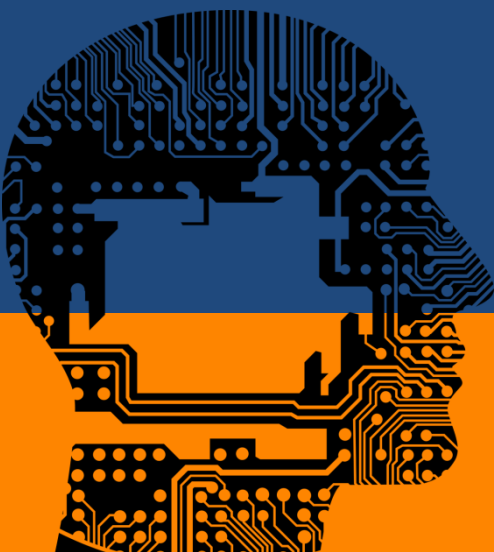
\$elementNode.innerHTML

- innerHTML 프로퍼티는 요소 노드 내부의 HTML 마크업을 취득하거나 변경합니다.
- textContent는 HTML마크업을 무시하고 텍스트만 반환하지만 innerHTML은 마크업이 포함된 문자열을 그대로 반환합니다.

JAVA 웹 개발자 양성과정

Javascript

2. DOM 조작



document.createElement(tagName)

- createElement(tagName) 메서드는 요소 노드를 새롭게 생성하여 반환합니다.
- 매개변수 tagName에는 태그 이름을 나타내는 문자열을 인수로 전달합니다.
- 이 메서드로 생성된 요소 노드는 아무런 자식 노드도 가지지 않습니다. 즉, 텍스트 노드도 없는 상태입니다.
- 해당 메서드는 요소 노드를 생성할 뿐 DOM에 추가하지 않으므로 DOM에 추가하는 사후 처리가 필요합니다.

document.createTextNode(text)

- createTextNode(text) 메서드는 텍스트노드를 생성하여 반환합니다.
- 매개변수 text에는 텍스트 노드의 값으로 사용할 문자열을 인수로 전달합니다.
- createElement(tagName)과 동일하게 노드가 생성만 될 뿐 DOM에 추가되지는 않습니다.

`$Node.appendChild(childNode)`

- `appendChild(childNode)` 메서드는 매개변수에게 전달한 자식 노드를 호출한 부모노드의 마지막 자식으로 추가합니다.
- 자식 노드로 텍스트 노드를 전달하면 텍스트가 추가됩니다.

* 지정한 위치에 노드 삽입

\$Node.insertBefore(newNode,childNode)

- insertBefore(newNode, childNode) 메서드는 첫 번째 인수로 전달받은 노드를 두 번째 인수로 전달받은 노드 앞에 삽입합니다.
- 두 번째 인수로 전달받은 노드는 반드시 insertBefore를 호출한 노드의 자식이어야 합니다. 그렇지 않으면 DOMException 에러가 발생합니다.

appendChild, insertBefore

- 이미 DOM에 존재하는 노드를 대상으로 appendChild() 메서드를 사용하면 현재 위치에서 해당 노드가 제거됨과 동시에 맨 끝으로 이동합니다.
- 또한 이미 DOM에 존재하는 노드를 insertBefore()로 중간으로 이동시킬 수 있습니다.

\$Node.cloneNode(deep)

- cloneNode() 메서드는 노드의 사본을 생성하여 반환합니다.
- 매개 변수 deep에 true를 전달하면 노드를 깊은 복사하여 모든 자손 노드가 포함된 사본을 생성하며, false를 전달하거나 생략하면 얇은 복사하여 자손노드 없이 자신만의 사본을 생성합니다.

\$Node.replaceChild(newChild, oldChild)

- replaceChild() 메서드는 자신을 호출한 노드의 자식 노드를 다른 노드로 교체합니다.
- 없애버릴 자식노드를 oldChild에 인수로 전달하고 그 자리를 새롭게 차지할 자식노드를 newChild에 전달합니다.

`$Node.removeChild(child)`

- `removeChild()` 메서드는 `child` 매개 변수에 인수로 전달한 노드를 DOM에서 제거합니다.
- 전달된 노드는 반드시 호출한 노드의 자식노드여야 합니다.

감사합니다
THANK YOU