

자바스크립트 (Javascript)

✓ 원리를 알면 IT가 맛있다

JavaScript for Beginners

chapter **07.**

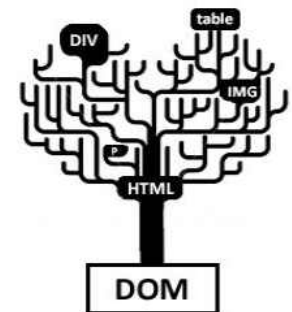
DOM

(Document Object Model)

- DOM 개요
- DOM 핵심 작업
 - 태그 추가/삭제
 - 속성 변경등 동적인 처리 담당.
- DOM 순회방법 및 핵심 API 정리
- Form 객체

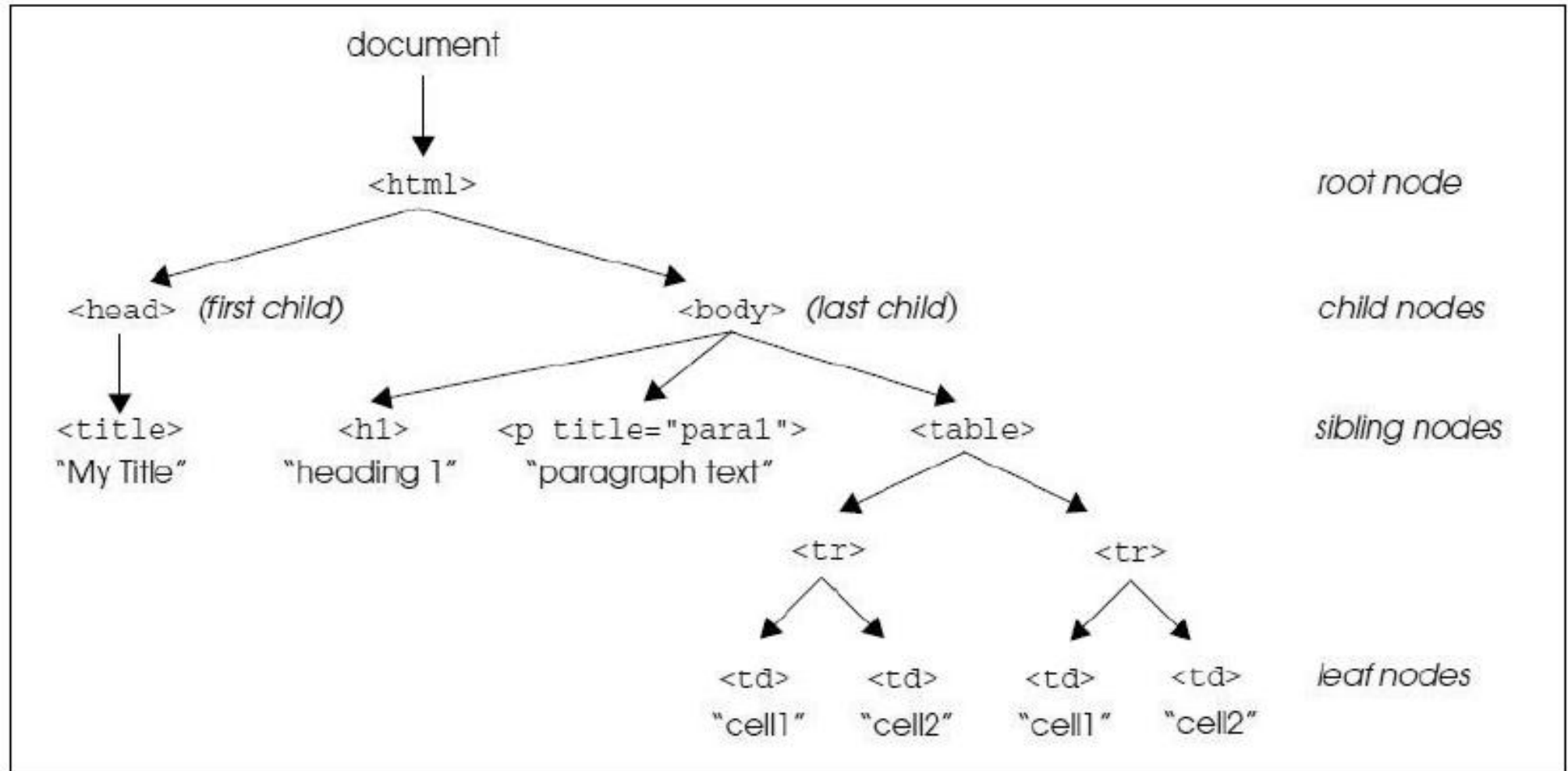
○ DOM 이란?

- 웹 페이지의 **HTML문서 구조를 객체로 표현하기 위한 계층구조**로서 문서를 나타내는 **최상위 객체**는 **document 객체**이다.
- 웹 페이지가 로드될 때, 웹 브라우저는 페이지의 DOM을 생성하고 **트리 구조로 관리**한다.
각각을 **노드(node)**라고 하며 **엘리먼트 노드**, **텍스트 노드**, **속성 노드**가 있다.
- DOM은 플랫폼/언어 **중립적으로** 구조화된 문서를 표현하는 **W3C의 공식 표준안** 이다.
- DOM은 HTML과 XML 문서에 대한 응용 API로서 문서를 **구조적(계층적인 트리구조)**으로 정리하여 문서의 내용과 시각적인 형태를 변경할 수 있도록 지원한다.



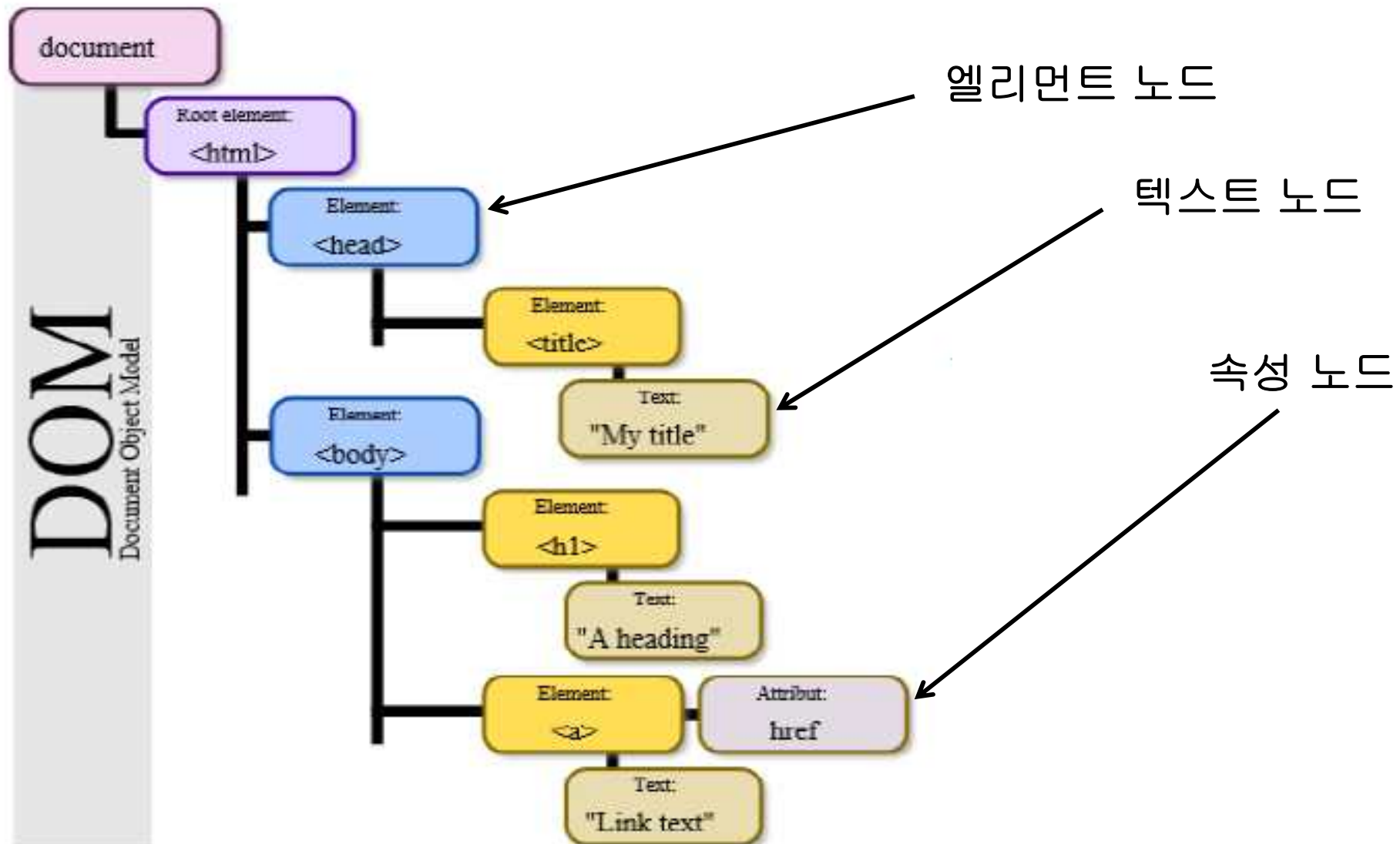
Document Object Model

❑ 1) DOM (Document Object Model) 개요



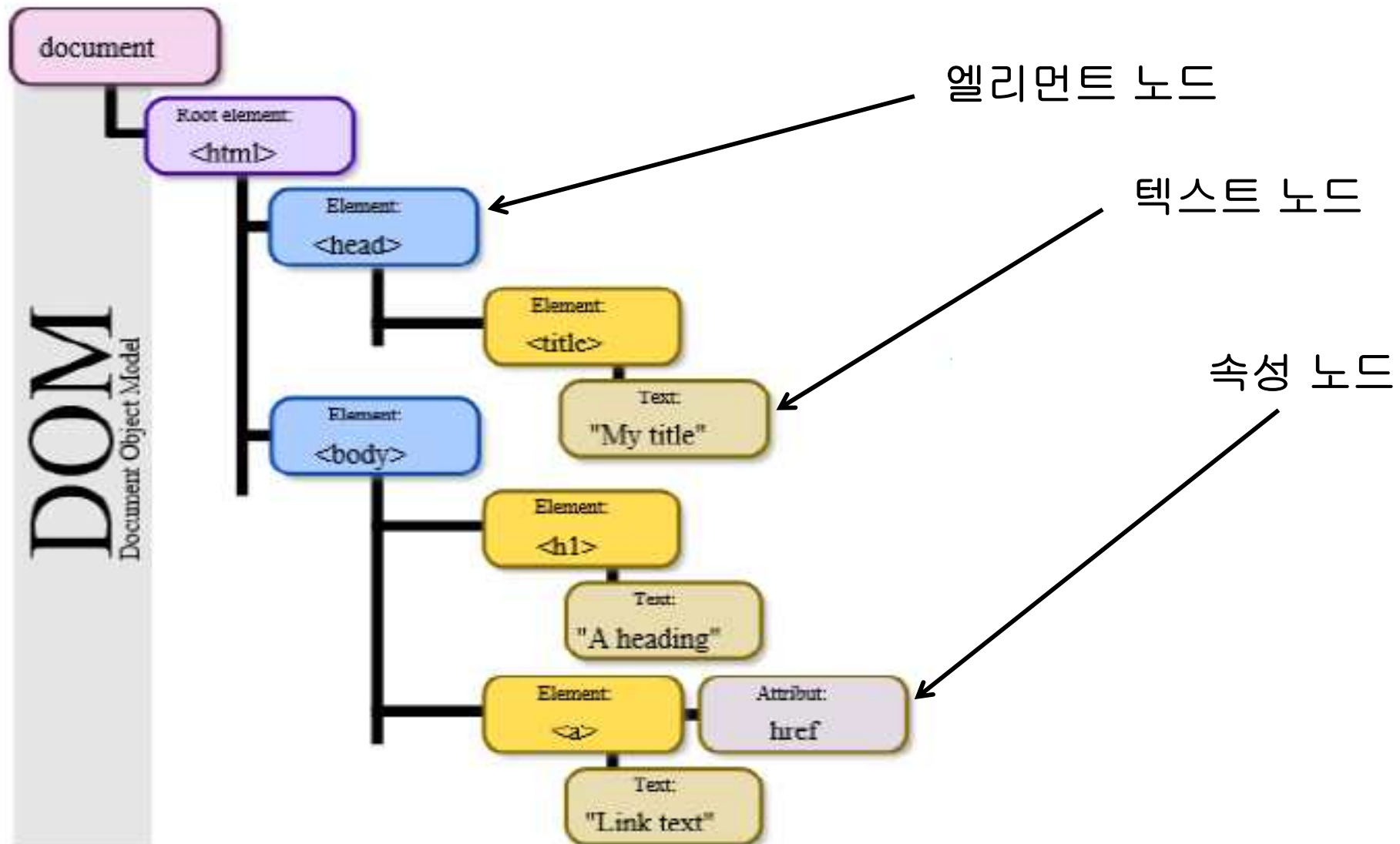
□ 1) DOM (Document Object Model) 개요

○ HTML문서의 DOM 표현방법



□ 1) DOM (Document Object Model) 개요

○ HTML문서의 DOM 표현방법



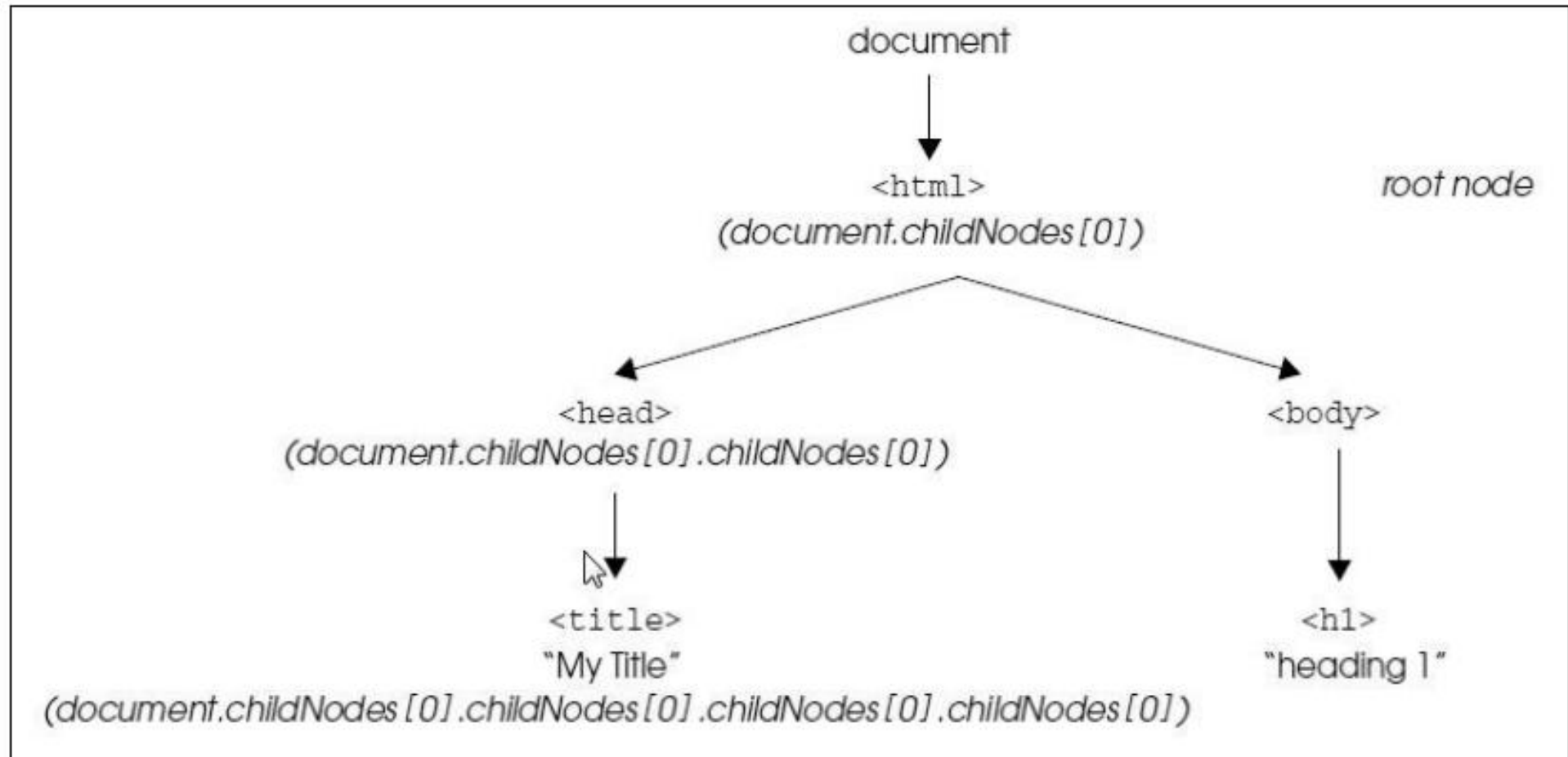
○ 노드(node)속성

속성	기능 설명
firstChild	첫 번째 자식 노드의 요소를 반환
lastChild	마지막 자식 노드의 요소를 반환
nextSibling	현재 자식 노드에서 같은 레벨의 요소 다음 자식 노드를 반환
nodeName	노드의 이름을 반환
nodeType	1은 요소, 2는속성, 3은 내용으로 노드의 형을 반환
nodeValue	내용에 노드 값을 지정
ownerDocument	노드가 소속되어 있는 문서의 최상위 요소를 반환
parentNode	현재 노드가 소속되어 있는 요소를 반환
previousSibling	현재 자식 노드와 같은 레벨 요소의 이전 자식 노드를 반환

○ 노드(node) 메소드

메소드	기능 설명
appendChild(새로운 노드)	자식 노드의 리스트 끝에 새 노드를 추가
cloneNode(옵션)	노드를 복사
hasChildNodes()	노드가 자식이면 true 발생
getAttribute(속성이름)	현재 노드의 속성 값을 반환
hasAttributes()	노드에 속성이 정의되어 있는 경우 Boolean 값을 반환
hasChildNodes()	노드에 자식 노드가 정의되어 있는 경우 Boolean 값을 반환
insertBefore(새로운 노드, 현재 노드)	자식 목록에 새로운 노드를 삽입
removeChild(자식 노드)	자식 목록에 현재 노드를 삭제
setAttributeNode(속성 참조)	현재 노드에 속성을 생성하거나 지정

○ 부모와 자식 노드 관계



○ DOM을 이용한 핵심 작업

- 가. 자바스크립트를 이용한 페이지의 모든 태그 변경 가능
- 나. 자바스크립트를 이용한 페이지의 모든 속성 변경 가능
- 다. 자바스크립트를 이용한 페이지의 모든 CSS 스타일 변경 가능
- 라. 자바스크립트를 이용한 페이지의 모든 이벤트 처리 가능.

○ DOM 핵심 API

- 순회
 - : childNodes , firstChild, lastChild , nextSibling , previousSibling
- 선택
 - : getElementById() , getElementsByTagName() , getElementsByClassName()
querySelector(), querySelectorAll()
- 생성 및 추가/삭제
 - : createElement(), createTextNode(), appendChild(), removeChild()
createAttribute() , setAttribute(), getAttribute()
- 내용 저장 및 조회: innerHTML, innerText

가. 직접 접근 방법

- id 값으로 접근

`document.getElementById(id값) : Node`

- 태그명으로 접근

`document.getElementsByTagName(tag명) : NodeList`

- class 속성으로 접근

`document.getElementsByClassName(class명) : NodeList`

- CSS selector 사용

`document.querySelectorAll(selector); // 일치하는 모든요소`

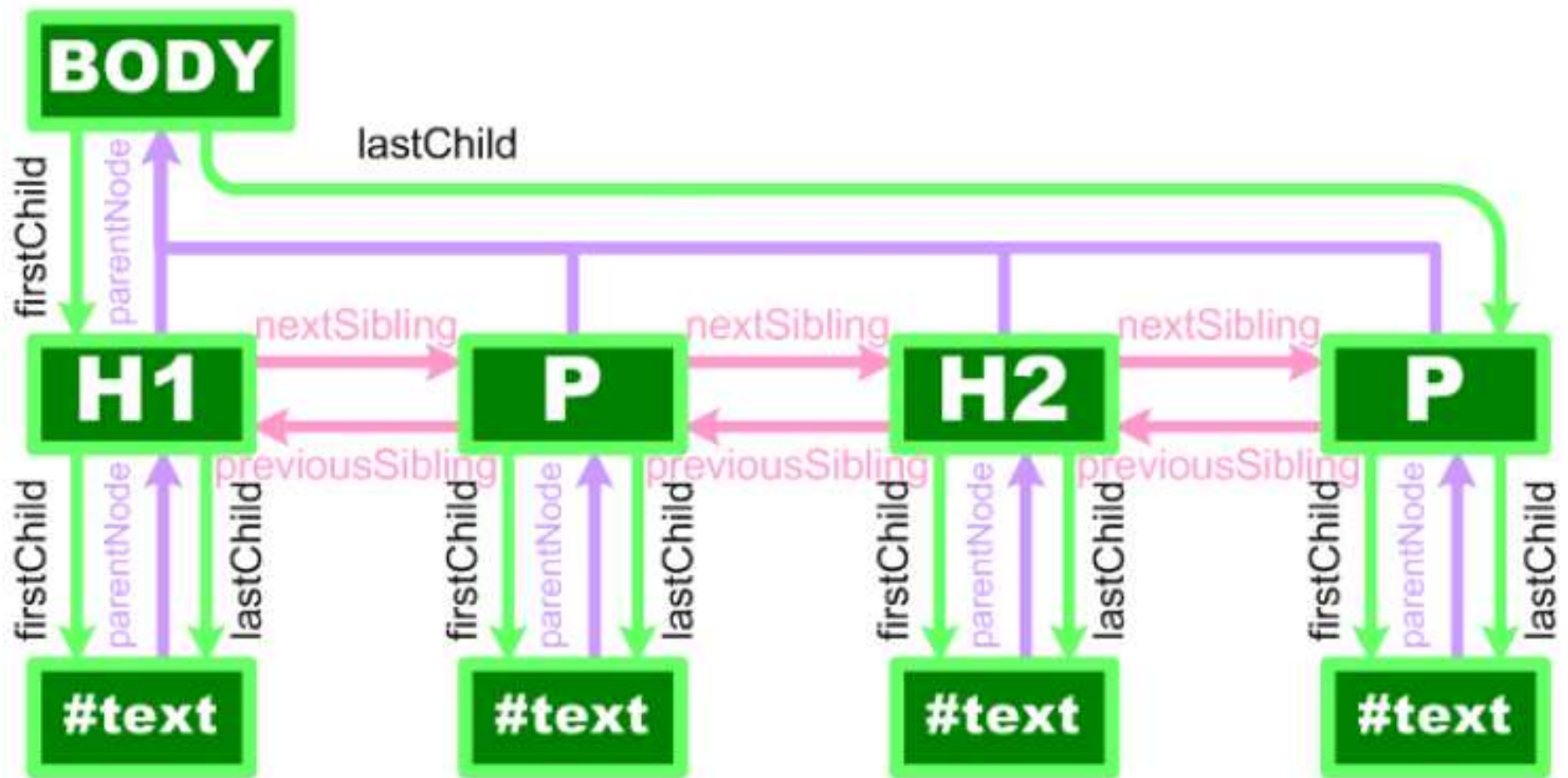
나. 노드 워킹 접근 방법

- 상대적인 위치로 접근하는 방법이다.

예> 첫번째 자식, 형제 , 마지막 자식 등

□ 3) DOM 객체 접근 방법

○ 노드 워킹 접근 방법

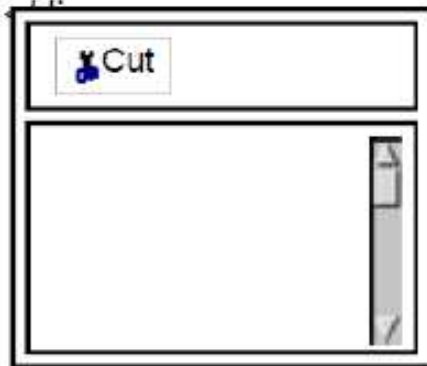


○ DOM을 이용한 HTML 엘리먼트 작성

Declarative HTML vs. Procedural DOM

HTML

```
<div id="main">  
  <div id="toolbar">  
    <button>  
      </img>  
      Cut  
    </button>  
  </div>  
  <textarea id="editor"></textarea>  
</div>
```



Document Object Model (DOM) in Javascript

```
var main = document.createElement("div");  
main.setAttribute("id", "window");  
  
var toolbar = document.createElement("div");  
toolbar.setAttribute("id", "toolbar");  
  
var button = document.createElement("button");  
var img = document.createElement("img");  
img.setAttribute("src", "cut.png");  
button.appendChild(img);  
  
var label = document.createTextNode("Cut");  
button.appendChild(label);  
toolbar.appendChild(button);  
main.appendChild(toolbar);  
  
var editor = document.createElement("textarea");  
editor.setAttribute("id", "editor");  
main.appendChild(editor);
```


□ 5) DOM이 모두 로딩되기를 기다리기

- DOM 작업시 어려움중 하나는 DOM이 완전히 로딩 되기전에 자바스크립트코드가 실행 될 수 있다는 것이다.
- 헤더에 포함된 스크립트와 외부 파일에서 불러온 자바스크립트는 HTML DOM이 실제로 생성되기 전에 실행된다.

따라서 헤더나 외부파일에서 실행된 모든 스크립트는 DOM에 접근할 수 없다.

해결방법>

가. 페이지가 완전히 로딩되기를 기다리기

```
window.onload = init;
```

나. body 태그 맨 끝에 스크립트 추가하기

```
<script type="text/javascript">...</script>  
</body>
```

* jQuery에서는 \$(document).ready(함수) 사용한다.

□ 6] Form 객체

- 폼은 웹의 핵심으로 브라우저(클라이언트)에서 서버에 입력 데이터를 전송하는 역할을 담당한다.
- 자바스크립트에서는 HTML폼에 form객체가 상호작용 할 수 있도록 지원되며 document 객체의 속성으로 설정된다. 접근은 document.forms[] 배열을 통해 접근할 수 있다.

* 폼 이벤트 핸들러는 다음과 같다.

```
<form action="cgi-bin/testform.cgi"
      method="post" onSubmit="return(readySubmit());" >

<form action="http://cgi-bin/testform.cgi"
      method="post" onReset="return resetAll();" />
```

이벤트 핸들러	기능 설명
onSubmit()	사용자가 데이터를 전송했을 때 발생하는 이벤트 핸들러
onReset()	리셋 버튼을 클릭 했을 때 발생하는 이벤트 핸들러

- 폼 컨트롤의 종류

- 모든 폼 컨트롤에 해당하는 객체가 제공된다. 따라서 속성과 메소드를 사용할 수 있다.

예> text 컨트롤에 대한 INPUT TEXT 객체가 제공된다.

입력 형식	속성	기능 설명
button	name, id	사용자의 입력을 위한 버튼을 작성한다.
text	name, id, size, maxlength	사용자의 입력을 위한 텍스트박스를 생성한다. size는 텍스트박스의 길이를 나타낸다. maxlength는 허용되는 문자의 최대 길이를 나타낸다.
textarea	name, id, size, rows, cols	textarea는 다중 라인을 입력할 수 있는 영역을 나타낸다. rows와 cols 박스의 크기를 나타낸다.
password	name, id, value	textbox와 같이 입력시 데이터를 감춘다. *는 입력문자를 대신하여 나타낸다.

입력 형식	속성	기능 설명
checkbox	name, id, value	<input type="checkbox"/> 를 나타내어 체크하면 true로 인식한다. name과 value로 다중으로 지정할 수 있다.
radio	name, id, value	<input type="radio"/> 를 나타내며 여러 개 중에서 단지 한 개만 체크할 수 있다.
select	name, id, option, size, multiple	popup 메뉴와 스크롤 박스를 나타낸다. 단지 한 개만 지정할 수 있다. 다중 속성은 스크롤 박스에서만 가능하다.
file	name, id	서버에 제공되는 파일목록을 나타낸다.
hidden	name, id, value	화면에 나타내는 이름과 value 값을 감추어 준다.
submit	name, id, value	클릭한 경우 서버에 정보를 전송한다.
image	src, name, id, value, align	submit 버튼과 같은 기능을 하며, 문자 대신에 이미지를 나타낸다. 이미지의 파일이름을 src속성으로 나타낸다.
reset	name, id value	폼에서 Reset을 누르면 입력 필드가 지워진다.

- DOM 개요
- DOM 핵심 작업
- DOM 순회방법 및 API



Thank you
