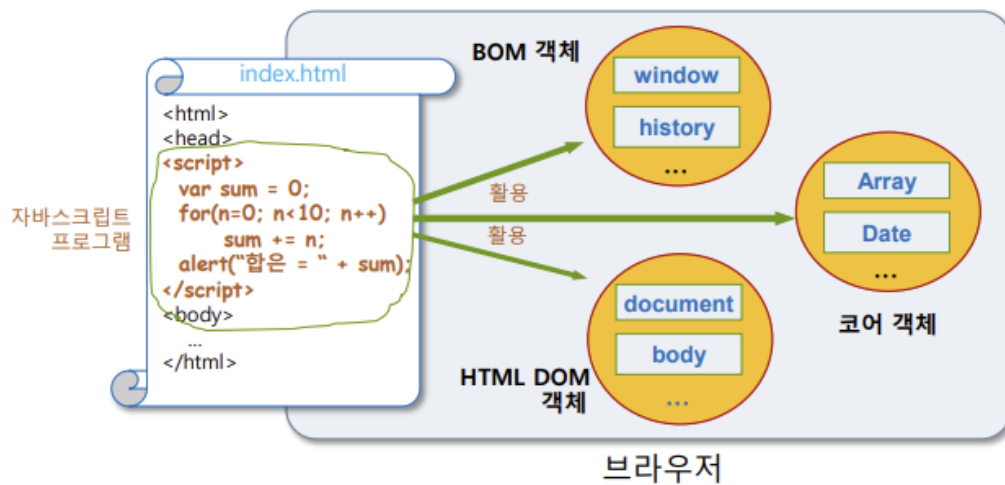


# 8장

## HTML DOM과 Document

- html 페이지와 자바스크립트 객체

: 자바스크립트 코드는 브라우저를 3가지 유형의 객체를 제공 받아 활용할 수 있다.

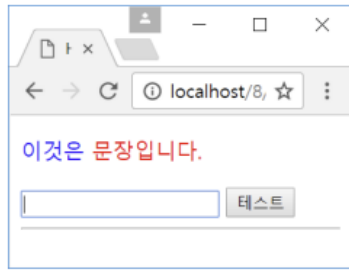


- HTML DOM

-DOM: 웹 페이지에 작성된 HTML 태그 당 객체 생성(HTML 태그가 출력 된 모양이나 콘텐츠를 제어하기 위해!)

-DOM 트리:HTML 태그의 포함 관계에 따라 DOM 객체의 트리 생성

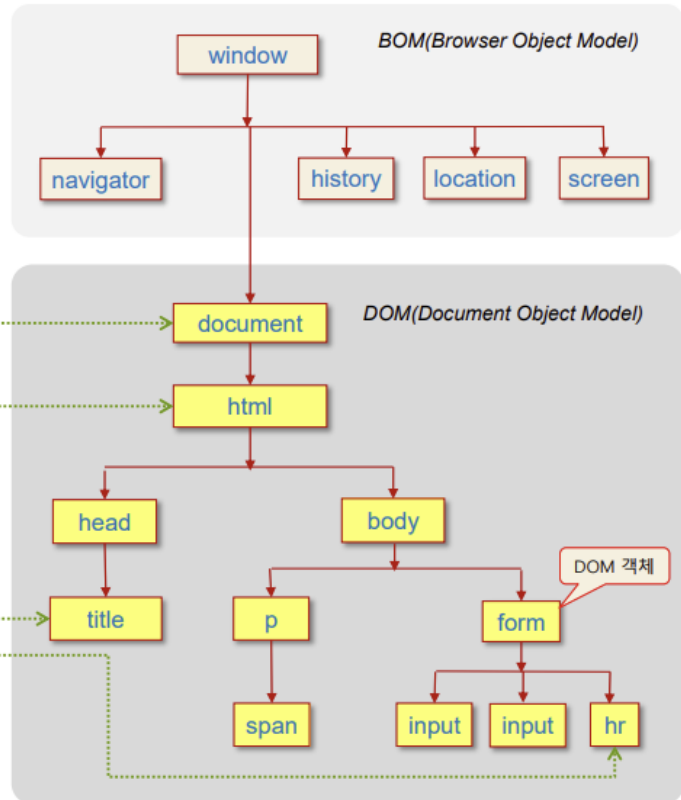
-DOM 객체 : DOM 트리의 한 노드



```

<!DOCTYPE html>
<html>
<head>
  <title>HTML DOM 트리</title>
</head>
<body>
  <p style="color:blue">이것은
    <span style="color:red">문장입니다.
  </span>
</p>
  <form>
    <input type="text">
    <input type="button" value="테스트">
  </form>
  <hr>
</body>
</html>

```



5

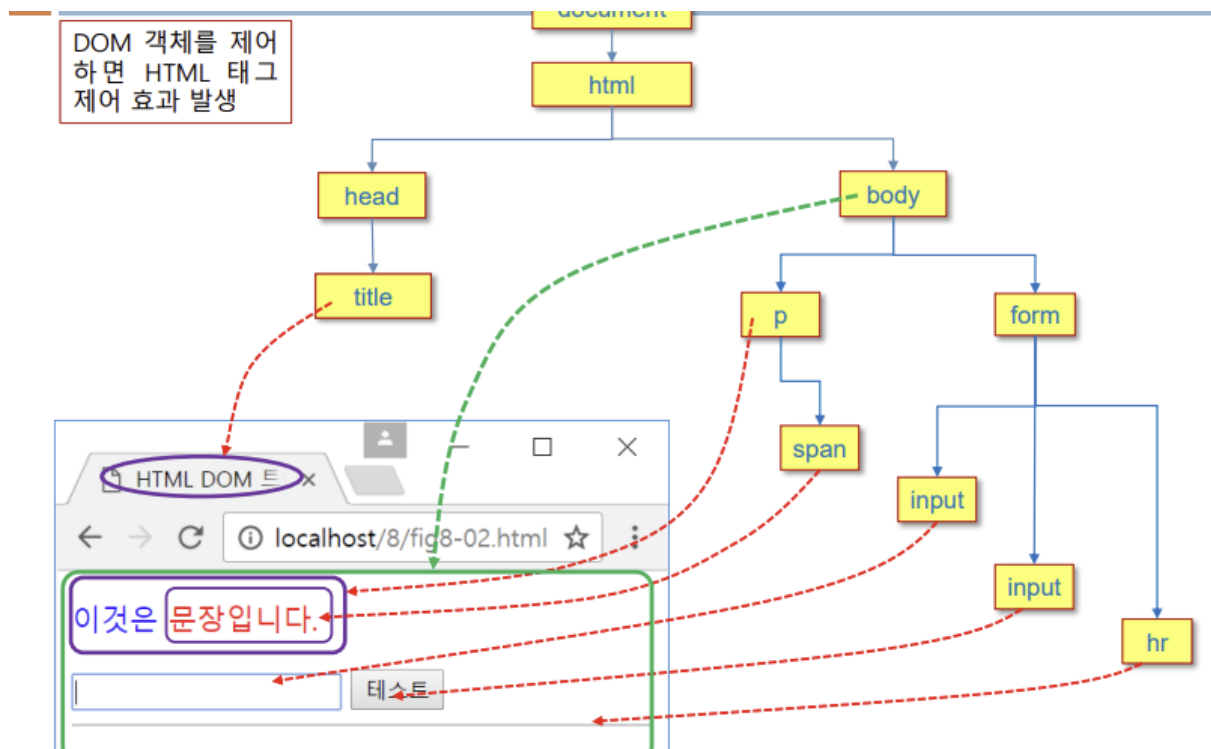
### • DOM 트리의 특징

: DOM 트리의 루트는 document 객체 , DOM 객체의 종류는 HTML 태그 종류만큼 HTML 태그 당 DOM 객체가 하나씩 생성 , HTML 태그의 포함 관계에 따라 DOM 트리에 부모 자식 관계

⇒ 브라우저가 HTML 태그를 화면에 그리는 과정

1. 브라우저가 DOM 트리의 톨 생성
2. 브라우저 HTML 태그를 읽고 DOM트리에 DOM 객체 생성
3. 브라우저는 DOM 객체를 화면에 출력
4. HTML 문서 로딩이 완료되면 DOM 트리 완성
5. DOM 객체 변경 시, 브라우저는 해당 HTML 태그의 출력 모양을 바로 갱신

<DOM 객체와 HTML 페이지의 화면 출력>



- HTML 태그의 요소

⇒ 엘리먼트 이름, 속성, CSS3 스타일, 이벤트 리스너, 콘텐츠 (innerHTML)



- DOM 객체의 구성요소

⇒ 프로퍼티, 메소드, 컬렉션, 이벤트 리스너, CSS3 스타일

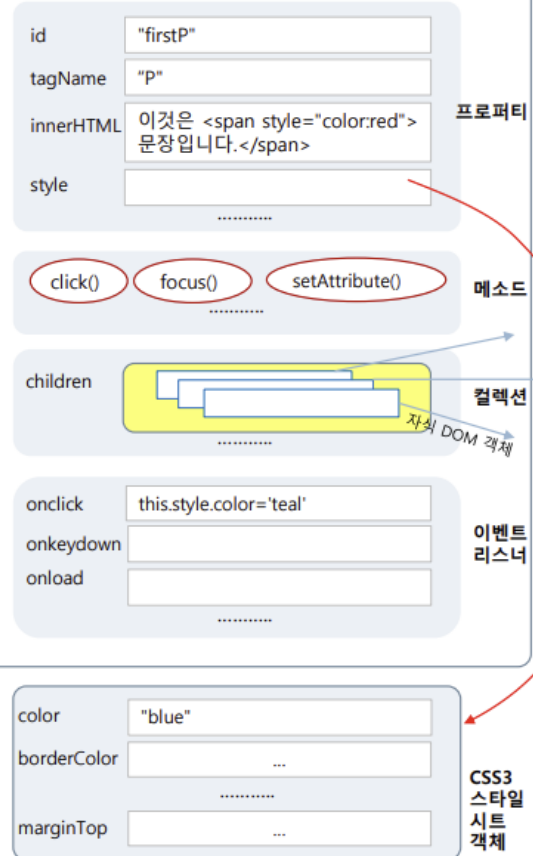
## DOM 객체의 구성

- 프로퍼티(property)
- 메소드(method)
- 컬렉션(collection)
- 이벤트 리스너(event listener)
- CSS3 스타일

## DOM 객체 p

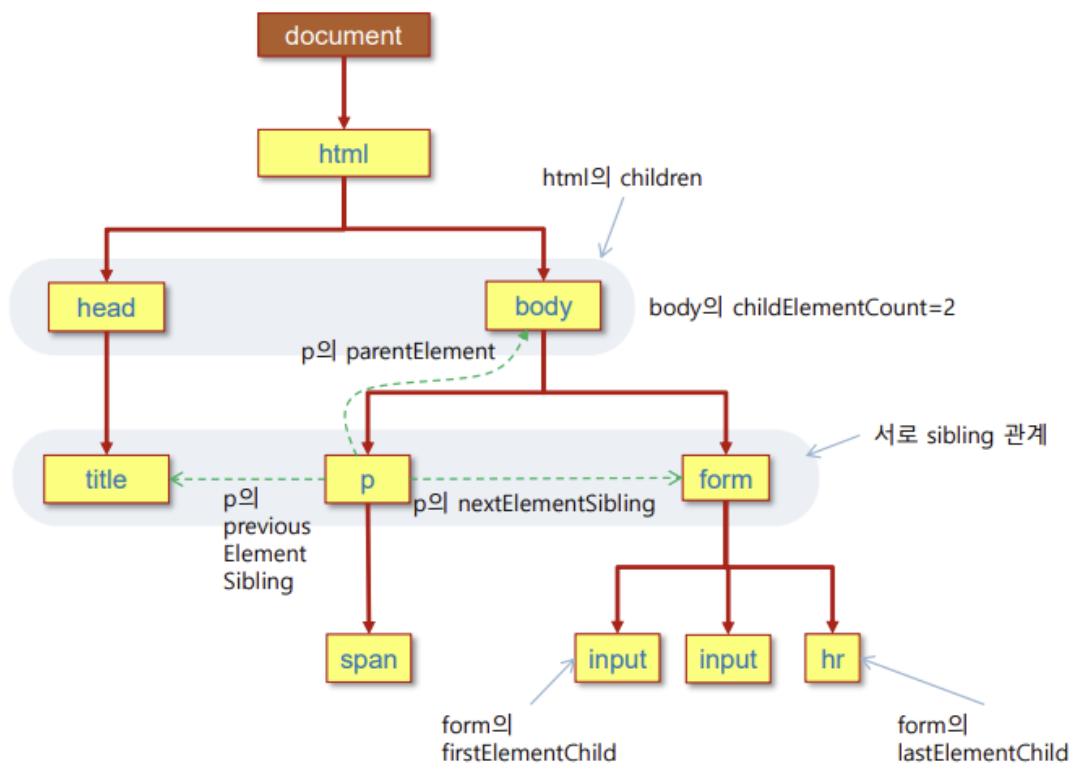
```
<p id="firstP"
  style="color:blue"
  onclick="this.style.color='teal'">
  이것은
  <span style="color:red">
  문장입니다.
  </span>
</p>
```

<p>...</p> 태그



10

## <관계>



- DOM 객체 다루기

⇒ dom 객체 구분 : id 태그 속성

⇒ dom 객체 찾기 : document.getElementById()

⇒ dom 객체의 CSS3 스타일 동적 변경 (

: CSS3 스타일 프로퍼티는 다음과 같이 사용

□ background-color 스타일 프로퍼티 -> backgroundColor

□ font-size 스타일 프로퍼티 -> fontSize

- innerHTML 프로퍼티

: 시작 태그와 종료 태그 사이에 들어있는 HTML 콘텐츠

: innerHTML 프로퍼티 수정 → HTML 태그의 콘텐츠 변경

- this 키워드

: 객체 자신을 가리키는 자바스크립트 키워드

: DOM 객체에서 객체 자신을 가리키는 용도로 사용

- document 객체

: HTML 문서 전체를 대변하는 객체

: DOM 객체를 접근하는 경로의 시작점

: DOM 트리의 최상위 객체

→ document 객체 접근 : window.document 또는 document 이름으로 접근

→ document 객체는 DOM 객체가 아님 ( 연결된 스타일 시트가 없음 )

- DOM 트리에서 DOM 객체 찾기

→ 태그 이름으로 찾기 : document.getElementsByTagName()

→ class 속성으로 찾기 : document.getElementsByClassName()

- document.write() 와 document.writeln()

\*HTML 페이지 로딩 과정

1. 브라우저는 HTML 페이지 로드 전 빈 상태 document 생성
2. 브라우저는 HTML 페이지를 위에서 아래로 해석
3. HTML 태그들을 document 객체에 담아간다(DOM 객체 생성).
4. </html> 태그를 만나면 document 객체를 완성하고 닫는다.

→ write() : document 객체에 담긴 HTML 콘텐츠 마지막에 HTML 태그들을 추가

→ writeln() : HTML 텍스트에 '\n' 을 덧붙여 출력. 한 칸 띄는 효과

- document의 열기와 닫기, open()과 close()

→ document.open() : 현재 브라우저에 출력 된 HTML 콘텐츠를 지우고 새로운 HTML 페이지 시작. 즉 document 객체에 담긴 DOM 트리를 지우고 새로 시작

→ document.close() : 현재 브라우저에 출력 된 HTML 페이지 완성! 더 이상 document.write() 할 수 없음

- 문서의 동적 구성

→ DOM 객체 동적 생성 : document.createElement("태그이름")

→ DOM 트리에 삽입

부모.appendChild(DOM객체);

부모.insertBefore(DOM객체 [, 기준자식]);

→ DOM 객체의 삭제

var removedObj = 부모.removeChild(떼어내고자하는자식객체);

```
var newDIV = document.createElement("div");
newDIV.innerHTML = "새로 생성된 DIV입니다.";
newDIV.setAttribute("id", "myDiv");
newDIV.style.backgroundColor = "yellow";
```

```
<div id="myDiv"
    style="background-color:yellow">
    새로 생성된 DIV입니다.
</div>
```

\* 이 자바스크립트 코드는 사실상 오른쪽의  
<div> 태그 정보를 가진 DOM 객체 생성