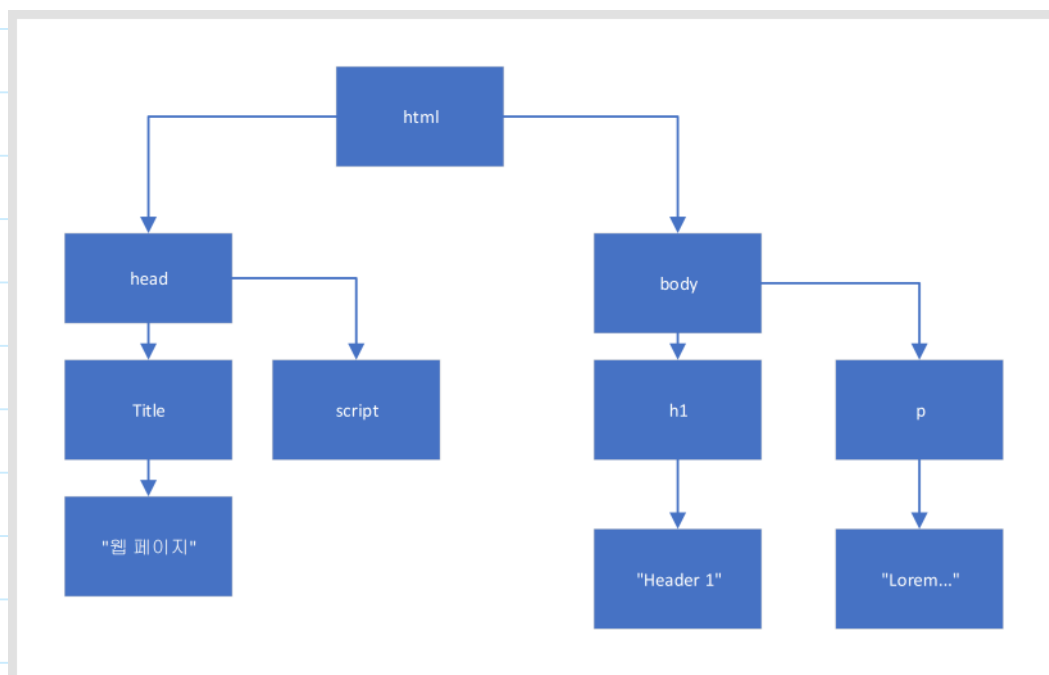


# 관련 용어

2022년 10월 8일 토요일    오후 6:09

- 문서 객체 모델은 넓은 의미로 웹 브라우저가 HTML 페이지를 인식하는 방법
- 좁은 의미로는 document 객체와 관련된 객체의 집합을 나타낸다
- 문서 객체 모델을 사용하면 HTML 페이지에 태그를 추가, 수정, 제거 할 수 있다

```
page2.html x
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>웹 페이지</title>
6
7 <script>
8 </script>
9
10 </head>
11 <body>
12 <h1>Header</h1>
13 <p>Lorem ipsum dolor amet</p>
14
15 </body>
16 </html>
```



- 이때 각 요소를 노드라고 한다
- H1노드와 script태그 처럼 요소를 생성 하는 노드를 요소 노드
- 화면에 출력되는 lorem 등은 텍스트 노드라고 한다



# 문서 객체 선택

2022년 10월 15일 토요일    오후 3:11

## 1. 1개의문서 객체 선택

메소드	설명
Document.querySelector(선택자)	선택자를 사용해 문서 객체를 선택

```
<html>
<head>
  <meta charset="UTF-8">
  <title>웹 페이지</title>

  <script>
    //화면 구성이 완료되면
    window.onload = function (){
      //문서 객체를 선택
      const headers = document.querySelectorAll('h1');
      for(let i = 0; i< headers.length; i++){
        //변수 선언
        const header = headers[i];

        //문서 객체를 조작
        header.style.color= 'orange';
        header.style.background = 'red';
        header.innerHTML = 'Form JavaScript' ;
      }
    };
  </script>
</head>
<body>
  <h1>Header</h1>
  <h1>Header</h1>
  <h1>Header</h1>
</body>
</html>
```

화면 캡처: 2022-10-15 오후 3:33

문서 객체 조작

2022년 10월 15일 토요일 오후 3:37

1. 문서 조작

속성	설명
innerHTML	문서 객체 내부의 글자를 나타낸다

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>문 페이지</title>
  </head>
  <script>
    // 문맥의 구성 완료
    window.onload = function () {
      // h1 태그 내부의 텍스트를 변경
      const header = document.getElementById('header');
      const originalText = header.innerHTML;
      header.innerHTML = '자바스크립트로 변경했어요!<br />';
      header.innerHTML += '원래는' + originalText + '였답니다!';
    };
  </script>
</html>
```

화면 캡처: 2022-10-15 오후 3:45

자바스크립트로 변경했어요!  
원래는Header - 1였답니다!

화면 캡처: 2022-10-15 오후 3:45

innerHTML 속성에 HTML 태그 형식의 문자열을 넣으면 해당 태그를 실제로 생성  
따라서 내부 태그 형식의 문자열을 넣고 싶을 때는  
<를 \$<로 >를 &gt; 이스케이프 문자로 넣어야 한다  
"" it = less than  
"" gt = grater than

[이스케이프 문자]

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>문 페이지</title>
  </head>
  <script>
    // 문맥의 구성 완료
    window.onload = function () {
      // h1 태그 내부의 글자 변경
      const header = document.getElementById('header');
      const originalText = header.innerHTML;
      header.innerHTML = ' <i> i 태그 입니다</i><br />';
      header.innerHTML += ' <i> i 태그 입니다</i>';
    };
  </script>
</html>
```

화면 캡처: 2022-10-15 오후 3:49

i 태그 입니다  
<i>i 태그 입니다</i>

화면 캡처: 2022-10-15 오후 3:49

2. 스타일 조작

- Style 속성 사용
- 스타일시트의 스타일 속성과 자바스크립트의 스타일 속성 차이

스타일 시트의 스타일 속성	자바스크립트의 스타일 속성
Background-color	backgroundColor
Border-radius	borderRadius
Border-bottom	borderBottom

```

<html>
  <head>
    <meta charset="UTF-8">
    <title>문 페이지</title>
  </head>
  <script>
    // 이벤트 연결
    window.onload = function () {
      // 문서 객체 추가

      let output = '';
      for(let i =0; i<256; i++){
        output += `<div></div>`;
      }

      document.body.innerHTML = output;

      // 문서 객체 선택
      const divs = document.querySelectorAll('div');
      for(let i = 0; i<divs.length; i++){
        // 변수 선언
        const div = divs[i];

        // 스타일 적용
        div.style.height = '2px';
        div.style.background = `rgb(${i + 1}, ${i + 1}, ${i + 1})`;
      }
    };
  </script>
</html>
```

화면 캡처: 2022-10-15 오후 4:04



화면 캡처: 2022-10-15 오후 4:04

3. 속성 조작

- 문서 객체의 속성 조작 메소드

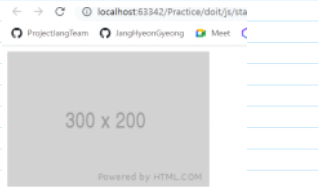
메소드	설명
setAttribute(속성 이름, 속성 값)	속성을 지정
getAttribute(속성 이름)	속성을 추출

- Image.src='rnt.png';
- Alert(image.src)
- 원 표준 정의한 속성 조작

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>문 페이지</title>
  </head>
  <script>
    // 이벤트 연결
    window.onload = function () {
      // 변수 선언
      const image = document.getElementById('image');

      // 속성변경
      image.src = 'https://via.placeholder.com/300x200';
      image.width = 300;
      image.height = 200;
    };
  </script>
</html>
```

화면 캡처: 2022-10-15 오후 4:16



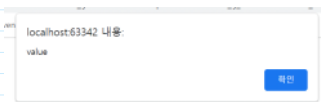
화면 캡처: 2022-10-15 오후 4:16

- 원 표준에서 정의하지 않은 속성 조작

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>문 페이지</title>
  </head>
  <script>
    // 이벤트 연결
    window.onload = function () {
      // 속성 지정
      document.body.setAttribute('data-custon', 'value');

      // 속성 추출
      const dataCuston = document.body.getAttribute('data-custon');
      alert(dataCuston);
    };
  </script>
</html>
```

화면 캡처: 2022-10-15 오후 4:22



화면 캡처: 2022-10-15 오후 4:22

이벤트

2022년 10월 15일 토요일 오후 3:56

- [이벤트]
- 키보드로 키를 입력하거나 마우스 클릭 등과 같은 현상이 프로그램에 영향을 미치는 의미
- 자바스크립트에서는 사용자가 직접 발생시키거나 애플리케이션이 자체적으로 발생시키는 두가지 방법이 있다
- 마우스 이벤트
- 키보드 이벤트
- HTML5 프래임 이벤트
- HTML5 입력 양식 이벤트
- 사용자 인터페이스 이벤트
- 구조 변화 이벤트
- 타지 이벤트

- 이벤트 관련 용어 정리
  - Window.onload = function () {}
    - onload : 이벤트 이름, 이벤트 타입
  - Load : 이벤트 이름, 이벤트 타입
  - 이벤트 속성에 붙은 함수는 이벤트리스너 또는이벤트 핸들러라고 부른다
- 이벤트 모델
  - DOM Level0
    - 인라인 이벤트 모델
    - 고전 이벤트 모델
  - 연결방식이 위와 가장 많이 사용
    - 이벤트를 중복해서 연결하지 못하는 단점 있다.
  - DOM Level2
    - 표준 이벤트 모델
    - 이벤트 중복 연결 가능하지만 복잡하다
  - 위의 단점을 Jquery 라이브러리를 사용해서 해결할 수 있다.

- 2. 인라인 이벤트 모델
- HTML태그 내부에서 이벤트를 연결하는 방법
- 웹 페이지의 가장 기본적인 이벤트 연결 방식으로 HTML 태그 내부에 자바스크립트 코드를 붙여 이벤트 실행

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Event Basic</title>
</head>
<body>
<button onclick="alert('click!')">클릭</button>
</body>
</html>
```

화면 캡처: 2022-10-15 오후 6:30



화면 캡처: 2022-10-15 오후 6:30

화면 캡처: 2022-10-15 오후 6:30

- On 문자열로 시작하는 속성은 모두 이벤트와 관련 있다
- 이벤트 속성

OnBlur	OnFocus	OnBlur	OnFocusOut	OnLoad
OnMove	OnScroll	OnLoad	OnClick	OnClick
OnMouseDown	OnMouseUp	OnMouseMove	OnMouseover	OnMouseout
OnMouseenter	Onmouseleave	OnChange	Onselect	Onsubmit
OnKeyDown	OnKeyPress	OnKeyUp	onerror	

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Inline Basic</title>
</head>
<script>
function ButtonClick(){
alert('click!')
}
</script>
</head>
<body>
<button onclick="ButtonClick()">클릭</button>
</body>
</html>
```



화면 캡처: 2022-10-15 오후 6:31

- 3. 고전 이벤트 모델
- Const image = document.getElementById("image");
- imageWidth = 100;
- imageHeight = 100;

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Inline Basic</title>
</head>
<script>
//이벤트 연결
window.onload = function ButtonClick(){
const button = document.getElementById(button);
}
//이벤트를 연결한다
button.onclick = function (){
alert('click!');
};
</script>
</head>
<body>
<button id = "button">클릭</button>
</body>
</html>
```

화면 캡처: 2022-10-15 오후 6:39

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<meta charset="UTF-8">
<title>Inline Basic</title>
</HEAD>
<BODY>
<button id = "button">클릭</button>
<script>
document.getElementById(button).onclick= function (){
this.innerHTML = this.innerHTML + " * ";
};
</script>
</BODY>
</HTML>
```

화면 캡처: 2022-10-15 오후 6:44



화면 캡처: 2022-10-15 오후 6:45

4. 이벤트 객체

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Inline Basic</title>
</head>
<body>
<button id = "button">클릭</button>
<script>
window.onload = function (event){
alert(event);
};
</script>
</body>
</html>
```

화면 캡처: 2022-10-15 오후 6:50



화면 캡처: 2022-10-15 오후 6:50