

Chapter 10

문서 객체 모델

웹프로그래밍

AI컴퓨터공학부
정경용 교수

목차

1. 문서 객체 모델의 기본 용어와 개념
2. 문서 객체 선택
3. 문서 객체 조작
4. 이벤트

학습목표

- 문서 객체와 문서 객체 모델의 개념을 이해합니다.
- 문서 객체를 선택하고 조작할 수 있습니다.
- 이벤트 종류를 알아보고 문서 객체에 이벤트를 연결합니다.

Section 01

문서 객체 모델의 기본 용어와 개념

01. 문서 객체 모델의 기본 용어와 개념

1. 문서 객체 모델 기본 용어

- 문서 객체: HTML 태그를 자바스크립트에서 사용할 수 있는 객체로 만든 것.
- 문서 객체를 조작한다는 말은 결국 태그를 조작한다는 의미

코드 10-1 dom_basic.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
  <script>

  </script>
</head>
<body>
  <h1>Header</h1>
  <p>Lorem ipsum dolor amet</p>
</body>
</html>
```

태그 : h1 또는 p

문서 객체 : HTML 태그를
자바스크립트에서 객체로 만든 것

문서 객체 조작 = 태그 조작

01. 문서 객체 모델의 기본 용어와 개념

1. 문서 객체 모델 기본 용어

- 트리의 각 요소가 노드
 - 요소 노드
 - <h1> 태그와 <script> 태그처럼 요소를 생성하는 노드
 - 텍스트 노드 : 화면에 출력되는 문자열

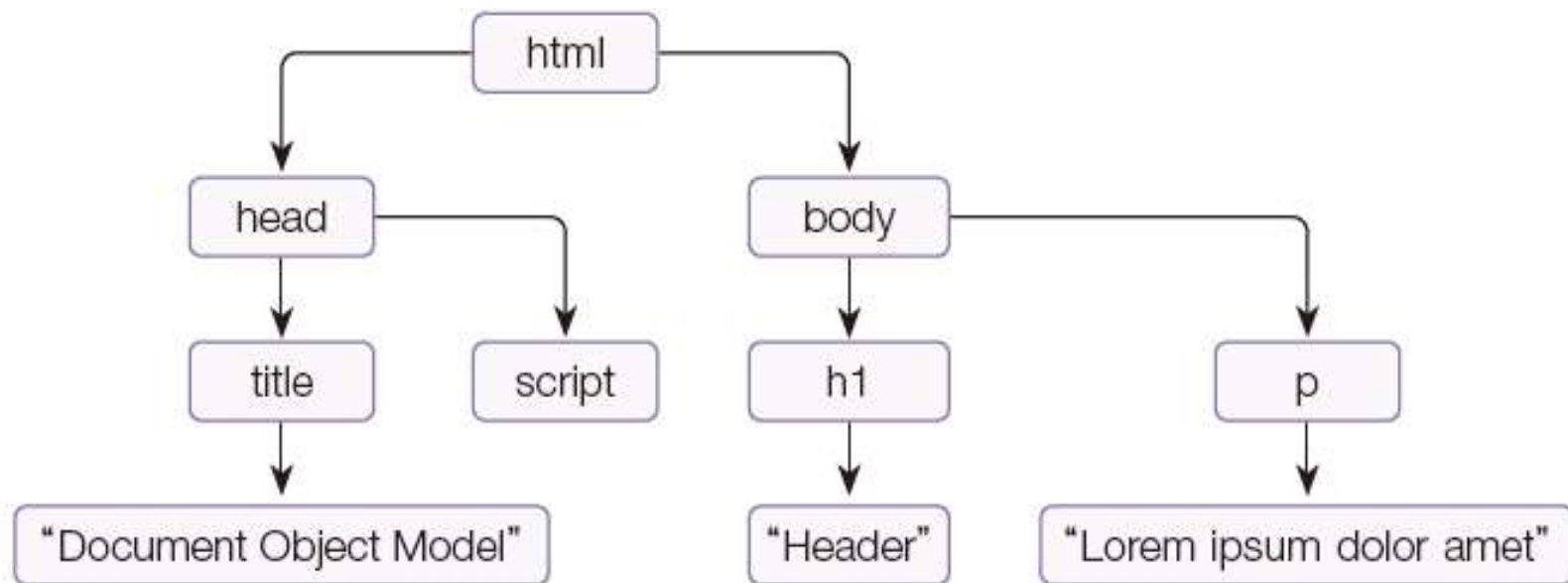


그림 10-1 노드

01. 문서 객체 모델의 기본 용어와 개념

1. 문서 객체 모델 기본 용어

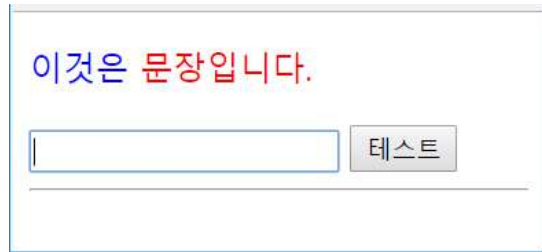
- 텍스트 노드를 갖지 않는 태그

```
<br>  
<hr>  

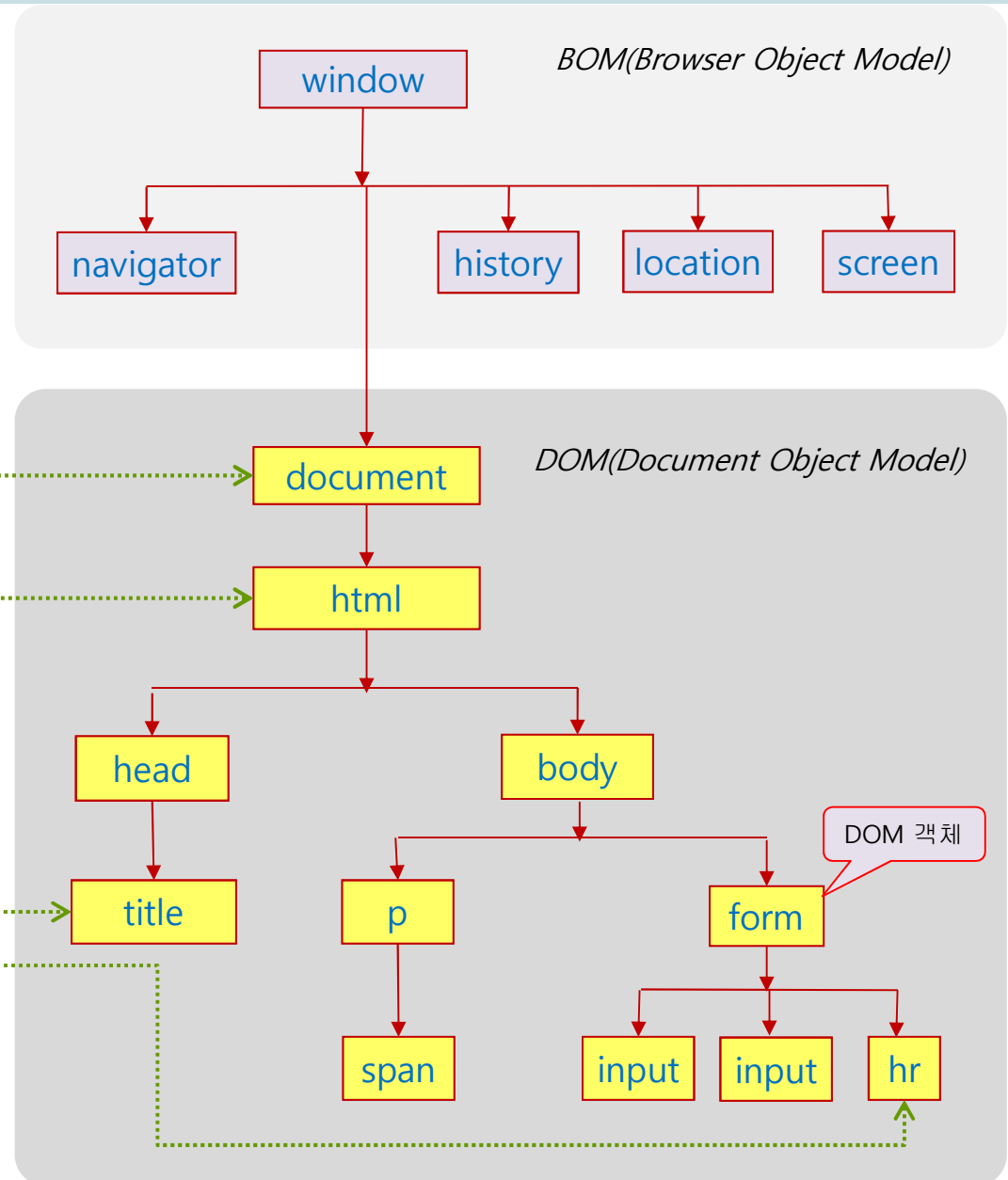
```

- 정적 생성:
 - 웹페이지를 처음 실행할 때 HTML 태그로 적힌 문서 객체를 생성.
- 동적 생성:
 - 페이지를 실행 중에 자바스크립트를 사용해 문서 객체를 생성.
- 객체 모델: Document Object Model, DOM
 - 웹 브라우저가 HTML 파일을 분석하고 출력하는 방식

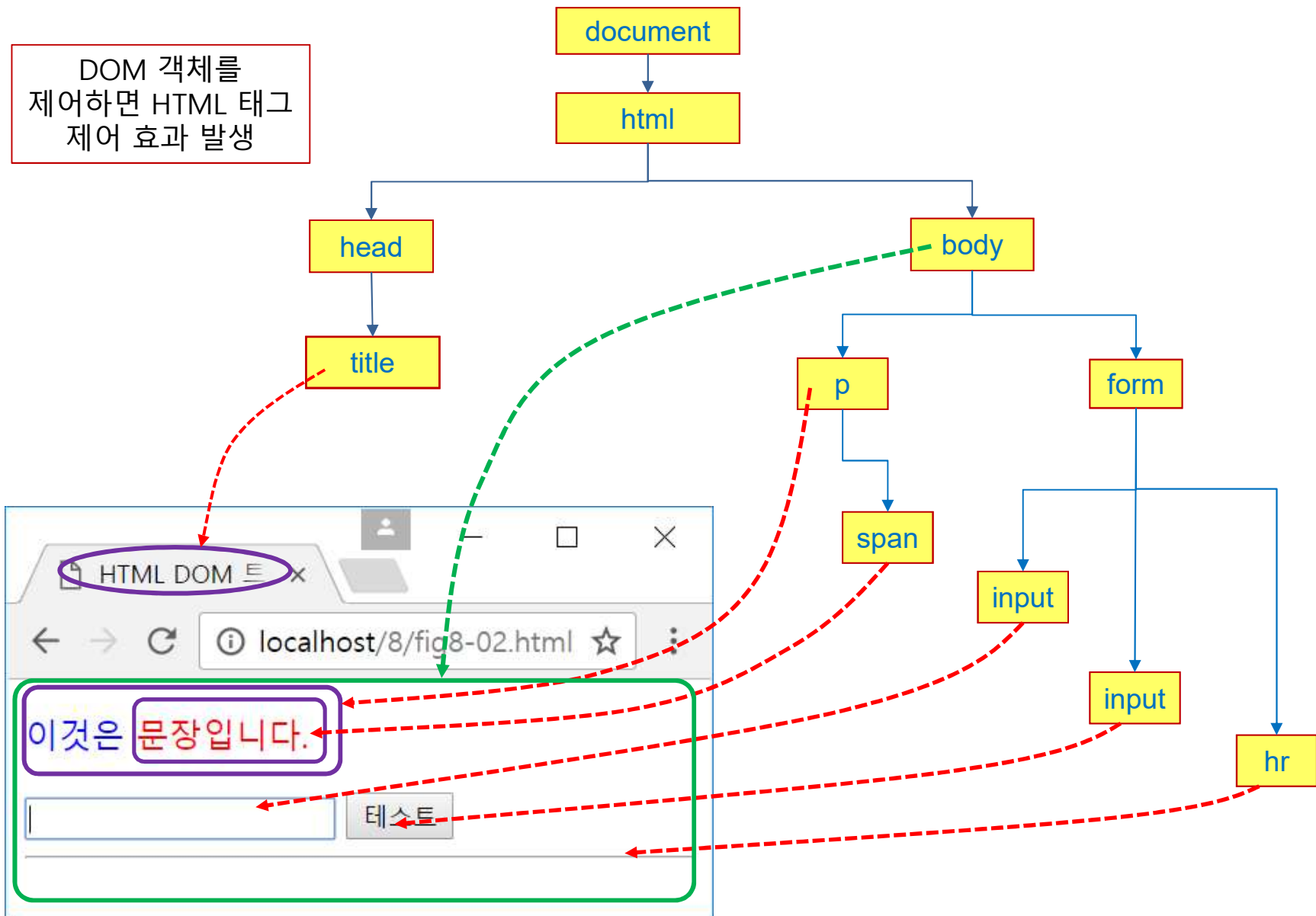
01. 문서 객체 모델의 기본 용어와 개념



```
<!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">
  <hr>
</form>
</body>
</html>
```



01. 문서 객체 모델의 기본 용어와 개념



01. 문서 객체 모델의 기본 용어와 개념

HTML 태그

- 엘리먼트(element)로도 불림
- 다음 5 가지 요소로 구성
 - 엘리먼트 이름
 - 속성
 - CSS3 스타일
 - 이벤트 리스너
 - 콘텐츠(innerHTML)

태그이름
(엘리먼트)

속성

CSS3 스타일

이벤트 리스너

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

콘텐츠(innerHTML)

01. 문서 객체 모델의 기본 용어와 개념

DOM 객체는 5 개의 요소 구성

- 프로퍼티(property)
 - HTML 태그의 속성(attribute) 반영
- 메소드(method)
 - DOM 객체의 멤버 함수로서, HTML 태그 제어 가능
- 컬렉션(collection)
 - 자식 DOM 객체들의 주소를 가지는 등 배열과 비슷한 집합적 정보
- 이벤트 리스너(event listener)
 - HTML 태그에 작성된 이벤트 리스너 반영
 - 약 70여 개의 이벤트 리스너를 가질 수 있음
- CSS3 스타일
 - HTML 태그에 설정된 CSS3 스타일 시트 정보를 반영
 - DOM 객체의 style 프로퍼티를 통해 HTML 태그의 모양 제어 가능

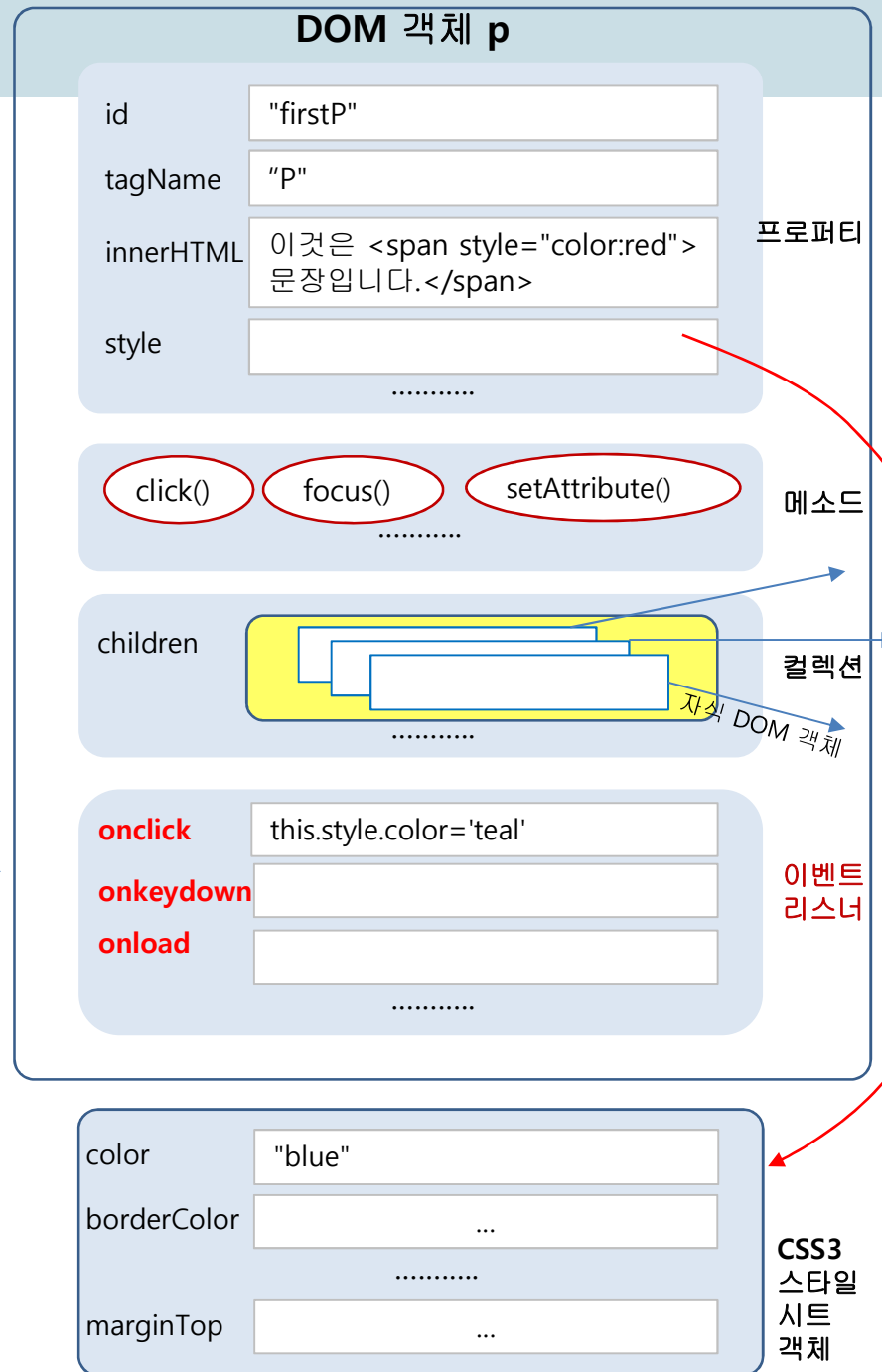
01. 문서 객체 모델

DOM 객체의 구성

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

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

<p>...</p> 태그



01. 문서 객체 모델의 기본 용어와 개념

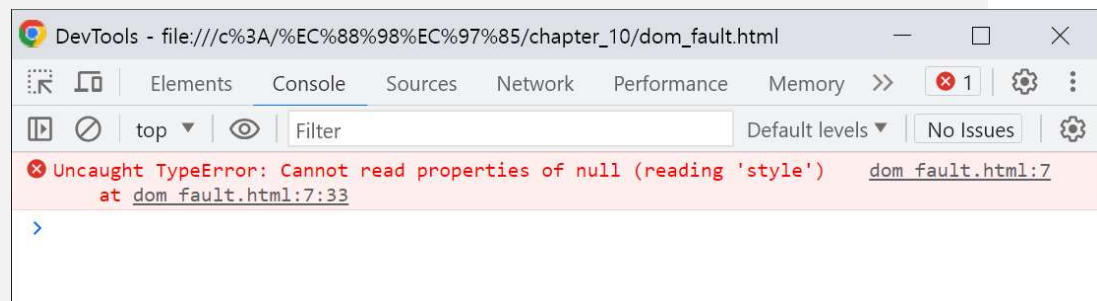
1. 웹 페이지 실행 순서

기본 예제 10-1 실행 순서에 따른 문서 객체 사용 오류

1 문서 객체 사용 오류

코드 10-2 dom_fault.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
  <script>
    // h1 태그의 배경 색상을 변경합니다.
    document.querySelector('h1').style.backgroundColor = 'red';
    // h2 태그의 글자 색상을 변경합니다.
    document.querySelector('h2').style.color = 'red';
  </script>
</head>
<body>
  <h1>Process - 1</h1>
  <h2>Process - 2</h2>
</body>
</html>
```



01. 문서 객체 모델의 기본 용어와 개념

1. 웹 페이지 실행 순서

기본 예제 10-1

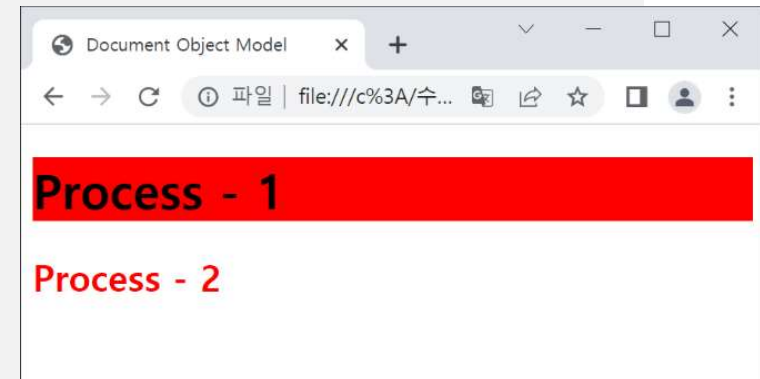
실행 순서에 따른 문서 객체 사용 오류

- 해결 방법 1: script 태그 위치 옮기기

코드 10-3

dom_correct.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
</head>
<body>
  <h1>Process - 1</h1>
  <h2>Process - 2</h2>
  <script>
    // h1 태그의 배경 색상을 변경합니다.
    document.querySelector('h1').style.backgroundColor = 'red';
    // h2 태그의 글자 색상을 변경합니다.
    document.querySelector('h2').style.color = 'red';
  </script>
</body>
</html>
```



01. 문서 객체 모델의 기본 용어와 개념

1. 웹 페이지 실행 순서

기본 예제 10-1

실행 순서에 따른 문서 객체 사용 오류

NOTE script 태그 위치

코드 10-4

note_scriptFirst.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
  <script src="jquery.js"></script>
  <script src="jquery.masonry.js"></script>
  <script src="jquery.rint.js"></script>
  <script src="jquery.rint.graphics.js"></script>
  <script src="jquery.rint.util.js"></script>
  <script>
    window.onload = function () { };
  </script>
</head>
<body>
  <h1>Header - 1</h1>
  <h2>Header - 2</h2>
</body>
</html>
```

01. 문서 객체 모델의 기본 용어와 개념

1. 웹 페이지 실행 순서

기본 예제 10-1 / 실행 순서에 따른 문서 객체 사용 오류

- 해결 방법 2: 이벤트 활용하기

코드 10-5 dom_event.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
  <script>
    window.onload = function () {
      // h1 태그의 배경 색상을 변경합니다.
      document.querySelector('h1').style.backgroundColor = 'red';
      // h2 태그의 글자 색상을 변경합니다.
      document.querySelector('h2').style.color = 'red';
    };
  </script>
</head>
<body>
  <h1>Process - 1</h1>
  <h2>Process - 2</h2>
</body>
</html>
```


Section 02

문서 객체 선택

02. 문서 객체 선택

- 문서 객체 선택: HTML 태그를 자바스크립트에서 문서 객체로 변환
- '문서 객체를 선택한다'.
 - 이미 존재하는 HTML 태그를 자바스크립트에서 문서 객체로 변환하는 것

표 10-1 문서 객체를 선택하는 메서드

구분	메서드	설명
1개 선택	document.getElementById(아이디)	아이디로 1개 선택
	document.querySelector(선택자)	선택자로 1개 선택
여러 개 선택	document.getElementsByName(이름)	name 속성으로 여러 개 선택
	document.getElementsByClassName(클래스)	class 속성으로 여러 개 선택
	document.querySelectorAll(선택자)	선택자로 여러 개 선택

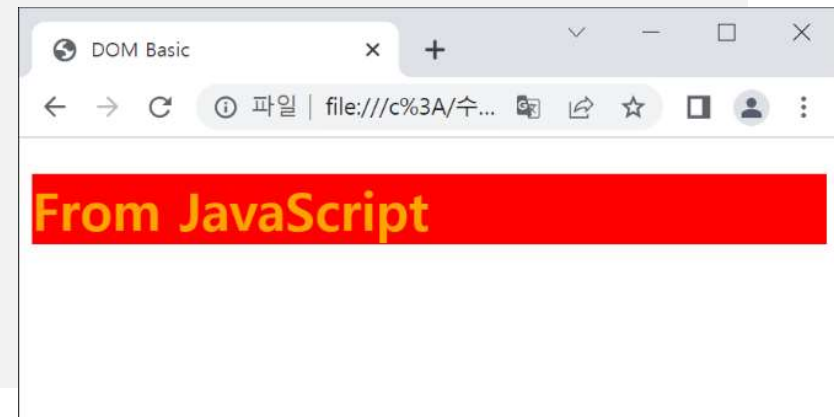
02. 문서 객체 선택

기본 예제 10-2 문서 객체 선택

1 getElementById() 메서드를 사용해 문서 객체 1개 선택하기

코드 10-6 select_id.html

```
...  
<script>  
  // 이벤트를 연결합니다.  
  window.onload = function () {  
    // 문서 객체를 선택합니다.  
    let header = document.getElementById('header');  
    // 문서 객체를 조작합니다.  
    header.style.color = 'orange';  
    header.style.background = 'red';  
    header.innerHTML = 'From JavaScript';  
  };  
</script>  
</head>  
<body>  
  <h1 id="header">Header</h1>  
</body>  
</html>
```



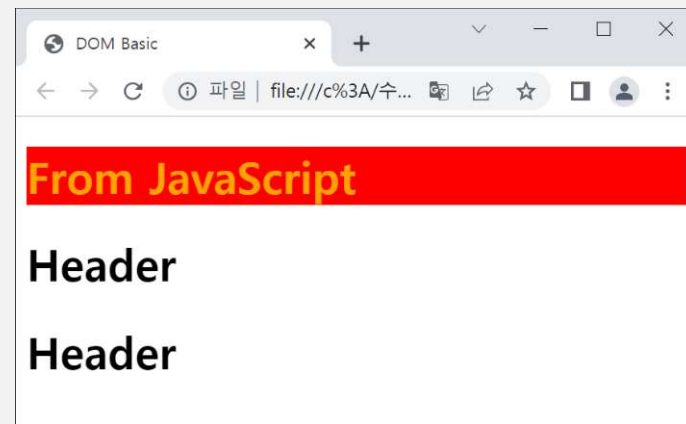
02. 문서 객체 선택

기본 예제 10-2 문서 객체 선택

2 `querySelector()` 메서드를 사용해 문서 객체 1개 선택하기

코드 10-7 select_query.html

```
...  
<script>  
  // 이벤트를 연결합니다.  
  window.onload = function () {  
    // 문서 객체를 선택합니다.  
    let header = document.querySelector('h1');  
  
    // 문서 객체를 조작합니다.  
    header.style.color = 'orange';  
    header.style.background = 'red';  
    header.innerHTML = 'From JavaScript';  
  };  
</script>  
</head>  
<body>  
  <h1>Header</h1>  
  <h1>Header</h1>  
  <h1>Header</h1>  
</body>  
</html>
```



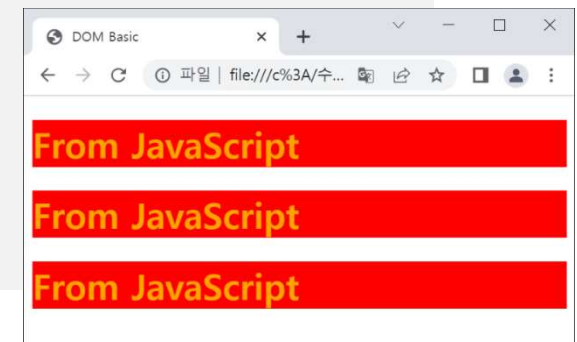
02. 문서 객체 선택

기본 예제 10-2 문서 객체 선택

3 querySelectorAll() 메서드를 사용해 문서 객체 여러 개 선택하기

코드 10-8 select_all.html

```
...  
<script>  
  // 이벤트를 연결합니다.  
  window.onload = function () {  
    // 문서 객체를 선택합니다.  
    let headers = document.querySelectorAll('h1');  
  
    for (let i = 0; i < headers.length; i++) {  
      // 변수를 선언합니다.  
      let header = headers[i];  
      // 문서 객체를 조작합니다.  
      header.style.color = 'orange';  
      header.style.background = 'red';  
      header.innerHTML = 'From JavaScript';  
    }  
  };  
</script>  
...
```



Section 03

문서 객체 조작

03. 문서 객체 조작

- 최신 웹 페이지 소스 코드의 HTML 태그에 내용이 없는 경우가 있음.

[유튜브 소스 코드 예 1]

```
<div class="video-skeleton">
  <div class="video-details">
    <div class="thumbnail skeleton-bg-color"></div>
    <div class="details">
      <div class="video-title text-shell skeleton-bg-color"></div>
      <div class="video-meta text-shell skeleton-bg-color"></div>
    </div>
  </div>
</div>
```

03. 문서 객체 조작

[유튜브 소스 코드 예 2]

```
<div id="info-container">
  <div id="primary-info" class="skeleton-light-border-bottom">
    <div id="title" class="text-shell skeleton-bg-color"></div>
    <div id="info">
      <div id="count" class="text-shell skeleton-bg-color"></div>
      <div class="flex-1"></div>
      <div id="menu">
        <div class="menu-button skeleton-bg-color"></div>
        <div class="menu-button skeleton-bg-color"></div>
        <div class="menu-button skeleton-bg-color"></div>
        <div class="menu-button skeleton-bg-color"></div>
        <div class="menu-button skeleton-bg-color"></div>
      </div>
    </div>
  </div>
</div>
```


03. 문서 객체 조작

```
        </div>
    </div>
    <div id="secondary-info" class="skeleton-light-border-bottom">
        <div id="top-row">
            <div id="video-owner" class="flex-1">
                <div id="channel-icon" class="skeleton-bg-color"></div>
                <div id="upload-info" class="flex-1">
                    <div id="owner-name" class="text-shell skeleton-bg-color">
                        </div>
                    <div id="published-date" class="text-shell skeleton-bg-color">
                        </div>
                </div>
            </div>
            <div id="subscribe-button" class="skeleton-bg-color"></div>
        </div>
    </div>
</div>
```

- 유튜브는 SPA 사이트 Single Page Application

03. 문서 객체 조작

1. 글자 조작

- 글자 속성
 - 문서 객체 내부에 있는 글자를 조작할 때 사용

표 10-2 글자 속성

속성	설명
textContent	문서 객체 내부 글자를 순수 텍스트 형식으로 가져오도록 변경
innerHTML	문서 객체 내부 글자의 HTML 태그를 반영해 가져오도록 변경

03. 문서 객체 조작

1. 글자 조작

기본 예제 10-3 내부 글자 변경

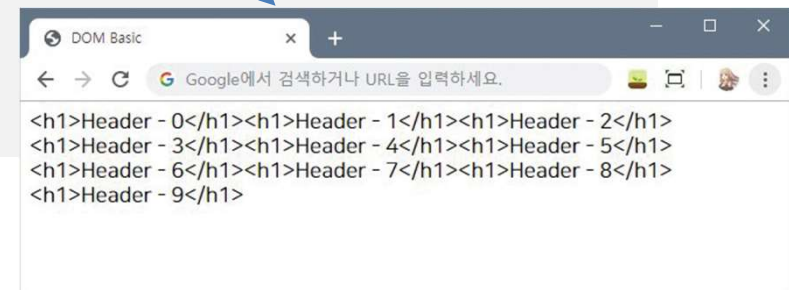
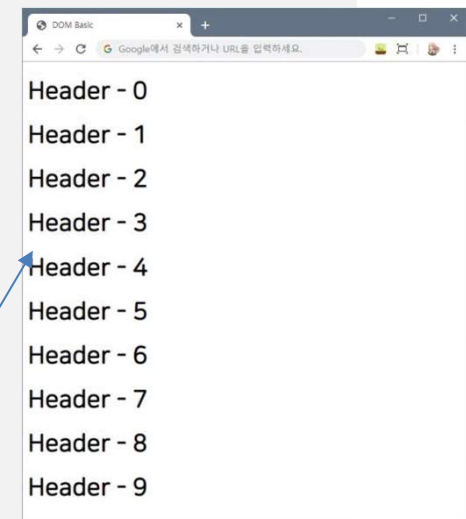
코드 10-9 control_text.html

...

```
<script>
  // 이벤트를 연결합니다.
  window.onload = function () {
    // 변수를 선언합니다.
    let output = '';
    for (let i = 0; i < 10; i++) {
      output += '<h1>Header - ' + i + '</h1>';
    }

    // 문서 객체 내부의 글자를 변경합니다.
    document.body.textContent = output; ①
    // innerHTML 속성을 사용합니다.
    document.body.innerHTML = output; ②
  };
</script>
```

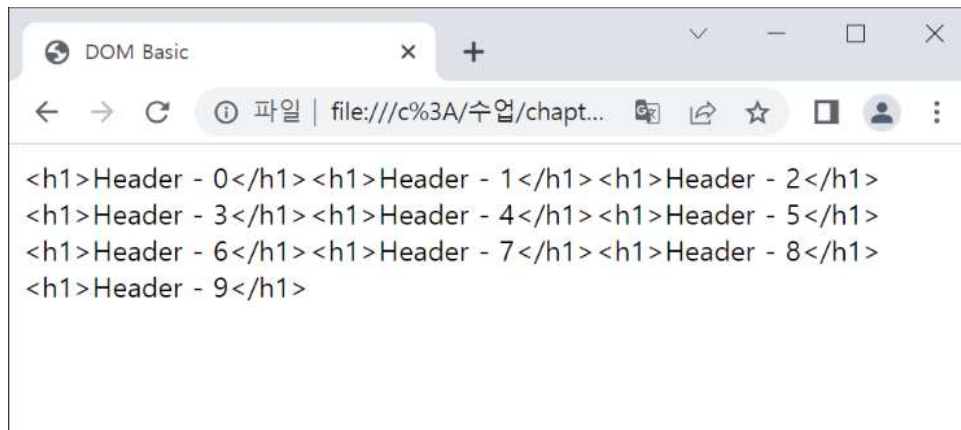
...



03. 문서 객체 조작

1. 글자 조작

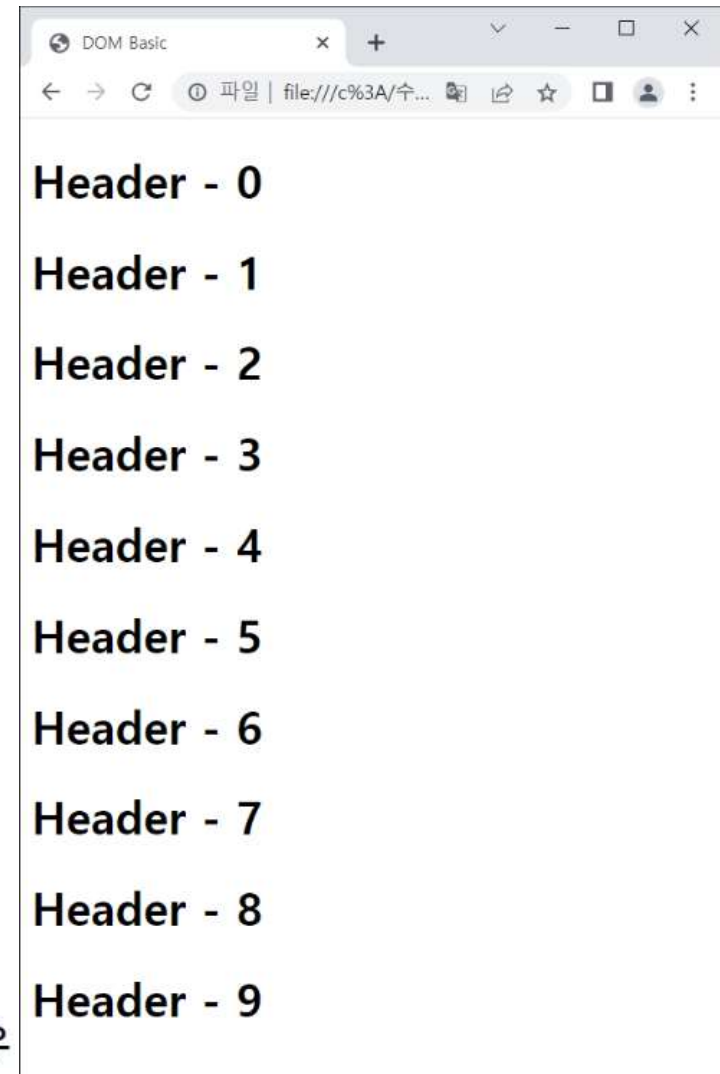
기본 예제 10-3 내부 글자 변경



```
<h1>Header - 0</h1><h1>Header - 1</h1><h1>Header - 2</h1>  
<h1>Header - 3</h1><h1>Header - 4</h1><h1>Header - 5</h1>  
<h1>Header - 6</h1><h1>Header - 7</h1><h1>Header - 8</h1>  
<h1>Header - 9</h1>
```

❶ textContent를 사용한 경우

❷ innerHTML을 사용한 경우



03. 문서 객체 조작: DOM 객체의 구조 출력 : p 객체 사례 - 실습1

```
<!DOCTYPE html>
<html>
<head><title>HTML DOM 트리</title></head>
<body>
<h3>DOM 객체 p의 프로퍼티, 스타일, 이벤트 리스너</h3>
<hr>
<p id="firstP"
  style="color:blue; background:yellow"
  onclick="this.style.color='teal'">
  이것은 <span style="color:red">문장입니다.
</span>
</p>
<script>
  var p = document.getElementById("firstP");
  var text = "p.id = " + p.id + "\n";
  text += "p.tagName = " + p.tagName + "\n";
  text += "p.innerHTML = " + p.innerHTML + "\n";
  text += "p.style.color = " + p.style.color + "\n";
  text += "p.onclick = " + p.onclick + "\n";
  text += "p.childElementCount = " + p.childElementCount + "\n";
  text += "너비 = " + p.offsetWidth + "\n";
  text += "높이 = " + p.offsetHeight + "\n";
  alert(text);
</script>
</body>
</html>
```

id가 firstP인 태그의 DOM 찾기

DOM 객체 p의 프로퍼티,
스타일, 이벤트 리스너

이것은 문장입니다.

localhost 내용:

```
p.id = firstP
p.tagName = P
p.innerHTML = 이것은
□<span style="color:red">문장입니다.</span>

p.style.color = blue
p.onclick = function onclick(event) {
  this.style.color='teal'
}
p.childElementCount = 1
너비 = 234
높이 = 21
```

확인

03. 문서 객체 조작: DOM 객체의 구조 출력 : p 객체 사례 - 실습2

```
<!DOCTYPE html>
<html> <head> <title>CSS 스타일 동적 변경</title>
<script>
function change() {
  var span = document.getElementById("mySpan");
  span.style.color = "green"; // 글자 색 green
  span.style.fontSize = "30px"; // 글자 크기는 30픽셀
  span.style.display = "block"; // 블록 박스로 변경
  span.style.width = "6em"; // 박스의 폭. 6 글자 크기
  span.style.border = "3px dotted magenta"; // 3픽셀 점선 magenta 테두리
  span.style.margin = "20px"; // 상하좌우 여백 20px
}
</script>
</head>
<body>
<h3>CSS 스타일 동적 변경</h3>
<hr>
<p style="color:blue" >이것은
  <span id="mySpan" style="color:red">문장입니다.</span>
</p>
<input type="button" value="스타일변경" onclick="change()">
</body>
</html>
```

버튼을 클릭하면
change() 함수 호출.
스타일 변경

CSS 스타일 동적 변경

이것은 문장입니다.

스타일변경

CSS 스타일 동적 변경

이것은

문장입니다.

스타일변경

인라인 박스가
블록 박스로 변경

03. 문서 객체 조작: innerHTML을 이용하여 HTML 콘텐츠 동적 변경-실습3

innerHTML 프로퍼티

- 시작 태그와 종료 태그 사이에 들어 있는 HTML 콘텐츠

```
<p id="firstP" style="color:blue">  
  여기에<span style="color:red">  
  클릭하세요.</span>  
</p>
```

innerHTML

- innerHTML 프로퍼티 수정 -> HTML 태그의 콘텐츠 변경

```
var p = document.getElementById("firstP");  
p.innerHTML = "나의 <img src='puppy.jpg'>강아지입니다.";
```



```
<p id="firstP" style="color:blue">  
  나의 <img src='puppy.jpg'>  
  강아지입니다.  
</p>
```

03. 문서 객체 조작:innerHTML을 이용하여 HTML 콘텐츠 동적 변경-실습3

innerHTML을 이용하여 HTML 콘텐츠 동적 변경

```
<!DOCTYPE html>
<html>
<head>
<title>innerHTML 활용 </title>
<script>
function change() {
    var p = document.getElementById("firstP");
    p.innerHTML= "나의 <img src='puppy.png'> 강아지";
}
</script>
</head>
<body>
<h3>innerHTML 활용 : 아래 글자에 클릭하면
예쁜 강아지가 보입니다.</h3>
<hr>
<p id="firstP" style="color:blue"
    onclick="change()">
    여기에 <span style="color:red">클릭하세요</span>
</p>
</body>
</html>
```

innerHTML 활용 : 아래 글자에 클릭하면 예쁜 강아지가 보입니다.

여기에 클릭하세요

마우스 클릭하면 아래와 같이 변경됩니다.

innerHTML 활용 : 아래 글자에 클릭하면 예쁜 강아지가 보입니다.



나의 강아지

03. 문서 객체 조작

2. 스타일 조작

- 자바스크립트로 CSS 속성 값을 추가·제거·변경 가능
- 1. 자바스크립트는 특수 문자 '-'을 식별자에 사용할 수 없으므로 오류 출력

```
❶ var header = document.getElementById('header');  
    header.style.background-color = 'red';
```

- 2. - 로 연결된 단어의 첫 글자를 대문자로 변경

```
❷ var header = document.getElementById('header');  
    header.style.backgroundColor = 'red';
```

스타일시트의 스타일 속성	자바스크립트의 스타일 식별자
background-image	backgroundImage
background-color	backgroundColor
box-sizing	boxSizing
list-style	listStyle

03. 문서 객체 조작

2. 스타일 조작

- div 태그 256개가 위쪽에서 아래쪽으로 그레이디언트를 형성

기본 예제 10-4 / 스타일 조작

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM Basic</title>
  <script>
    // 이벤트를 연결합니다.
    window.onload = function () {
      // 문서 객체를 추가합니다.
      var output = '';
      for (var i = 0; i < 256; i++) {
        output += '<div></div>';
      }
      document.body.innerHTML = output;
    }
  </script>
</head>
</html>
```

03. 문서 객체 조작

2. 스타일 조작

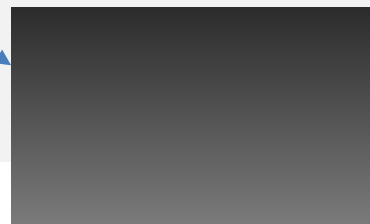
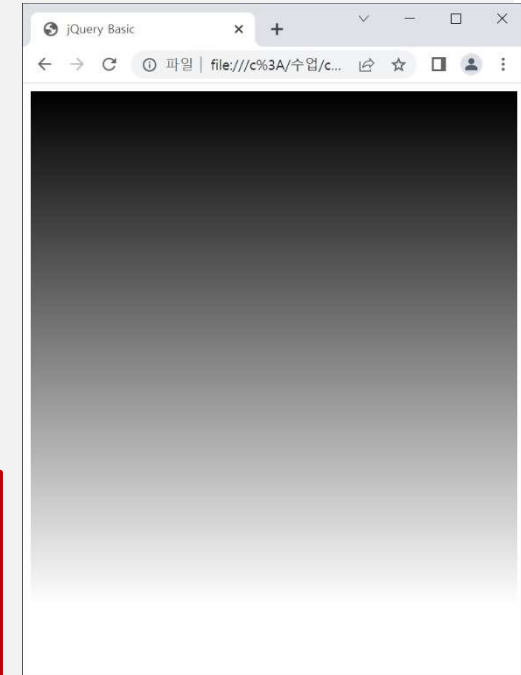
기본 예제 10-4

스타일 조작

...

```
<script>
  // 이벤트를 연결합니다.
  window.onload = function () {
    // 문서 객체를 추가합니다.
    let output = '';
    for (let i = 0; i < 256; i++) {
      output += '<div></div>';
    }
    document.body.innerHTML = output;
    // 문서 객체를 선택합니다.
    let divs = document.querySelectorAll('div');
    for (let i = 0; i < divs.length; i++) {
      // 변수를 선언합니다.
      let div = divs[i];
      // 스타일을 적용합니다.
      div.style.height = '2px';
      div.style.background = 'rgb(' + i + ',' + i + ',' + i + ')';
    }
  };
</script>
```

...



```
... <body> == $0
<div style="height: 2px; background: rgb(0, 0, 0);"></div>
<div style="height: 2px; background: rgb(1, 1, 1);"></div>
<div style="height: 2px; background: rgb(2, 2, 2);"></div>
<div style="height: 2px; background: rgb(3, 3, 3);"></div>
<div style="height: 2px; background: rgb(4, 4, 4);"></div>
<div style="height: 2px; background: rgb(5, 5, 5);"></div>
```

03. 문서 객체 조작

3. 속성 조작

표 10-4 문서 객체의 속성 조작 메서드

메서드	설명
setAttribute(속성 이름, 속성 값)	속성 지정
getAttribute(속성 이름)	속성 추출

- setAttribute(), getAttribute() 메서드
 - 웹 표준에서 지정하지 않은 속성에 접근할 때 사용
- 웹 브라우저 제조사가 구현하지 않았다면 접근하지 못할 수 있음.
- 'data-'로 시작하는 사용자 지정 속성은 반드시 속성 조작 메서드로 접근.

```
<body>
  <div data-role="page">
    <div data-role="header"></div>
    <div data-role="content"></div>
    <div data-role="footer"></div>
  </div>
</body>
```

문자열 'data-'로 시작하는 사용자 지정 속성은 반드시 속성 조작 메소드를 사용해서 접근해야 함

그림 10-2 jQuery Mobile 코드

03. 문서 객체 조작

3. 속성 조작

기본 예제 10-5 / 문서 객체 속성 조작

1 img 태그 속성 조작하기

코드 10-11 control_attribute.html

...

```
<script>
  // 이벤트를 연결합니다.
  window.onload = function () {
    // 변수를 선언합니다.
    let image = document.getElementById('image');

    // 속성을 변경합니다.
    image.src = 'http://via.placeholder.com/300x200';
    image.width = 300;
    image.height = 200;
  };
</script>
```

```
</head>
<body>
  <img id="image">
</body>
</html>
```



300 x 200

Elements Console Sources Network Performance Memory Appl

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body> == $0
    
  </body>
</html>
```

03. 문서 객체 조작

3. 속성 조작

기본 예제 10-5 문서 객체 속성 조작

2 body 태그 속성 조작하기

코드 10-12 control_body.html

```
...  
<script>  
  // 이벤트를 연결합니다.  
  window.onload = function () {  
    // 속성을 지정합니다.  
    document.body.setAttribute('data-custom', 'value');  
  
    // 속성을 추출합니다.  
    let dataCustom = document.body.getAttribute('data-custom');  
    alert(dataCustom);  
  };  
</script>  
...
```

```
<!DOCTYPE html>  
▼<html>  
  ▶<head>...</head>  
  <body data-custom="value"></body>  
</html>
```

검사로 속성 조작 확인

이 페이지 내용:

value

확인

03. 문서 객체 조작

3. 속성 조작

기본 예제 10-5 문서 객체 속성 조작

```
<!DOCTYPE html>
<html>
  <head> ... </head>
  <body data-custom="value"> </body> == $0
</html>
```

그림 10-3 검사로 속성 조작 확인

코드를 실행하고 검사를 사용해서 살펴보면 body 태그에 data-custom 속성이 추가되어 있음

03. 문서 객체 조작

3. 속성 조작

기본 예제 10-5 문서 객체 속성 조작

코드 10-13 dom_clock.html

Wed Jun 14 2023 10:55:47 GMT+0900 (한국 표준시)

...

```
<script>
  // 이벤트를 연결합니다.
  window.onload = function () {
    // 문서 객체를 선택합니다.
    let clock = document.getElementById('clock');
    // 1초마다 함수를 실행합니다.
    setInterval(function () {
      let now = new Date();
      clock.innerHTML = now.toString();
    }, 1000);
  };
</script>
```

```
</head>
<body>
  <h1 id="clock"></h1>
</body>
</html>
```

- 1초 단위로 현재 시간을 표시하는 시계 만들기
- setInterval() 함수를 사용해 1초마다 clock 문서 객체의 innerHTML 속성을 현재 시간으로 변경

// setInterval() 매개변수로 '실행될 함수'와 '시간'을 받습니다. 이때 시간은 밀리초로 설정해 줍니다. 1000 밀리초는 1초, 60000 밀리초는 1분입니다.

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body> == $0
    <h1 id="clock">Tue Jun 02 2020 16:21:56 GMT+0900 (대한민국 표준시)</h1>
  </body>
</html>
```


03. 문서 객체 조작

- [실습] 문서 객체를 사용한 시계: **setInterval() + clearInterval()** 함수
 - setInterval()함수를 변수에 대입하고, clearInterval()함수를 통해 중지가 가능한 1초 단위로 현재 시간을 표시하는 시계 만들기

```
<script>
    // 이벤트를 연결합니다.
    window.onload = function () {
        // 변수를 선언합니다.
        var clock = document.getElementById('clock');
        var cnt = 0;
        // 1초마다 함수를 실행합니다.
        var timer = setInterval(function () {
            cnt++;
            var now = new Date();
            clock.innerHTML = now.toString();

            if(cnt == 5)
                clearInterval(timer);
        }, 1000);
    }; // timer라는 setInterval()을 생성한 뒤 clearInterval()을 통해 종료
</script>
```

Section 04

이벤트

04. 이벤트

■ 이벤트 용어 정리

- 이벤트 속성 (event property) - **onload**
- 이벤트 이름(event name) 또는 이벤트 타입(event type) - on을 제외한 **load**
- 이벤트 리스너(event listener) 또는 이벤트 핸들러(event handler)
 - 이벤트 속성에 넣는 함수

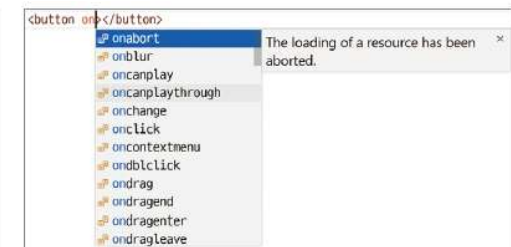
```
window.onload = function () { };
```

■ **이벤트**: 키보드 누르기, 마우스 클릭 등 어떤 현상이 프로그램에 영향을 미침.

- 마우스 이벤트
- 키보드 이벤트
- HTML 프레임 이벤트
- HTML 입력 양식 이벤트
- 사용자 인터페이스 이벤트
- 구조 변화 이벤트
- 터치 이벤트

```
onblur onfocus onfocusin onfocusout  
onload onresize onscroll onunload  
onclick ondblclick onmousedown  
onmouseup onmousemove onmouseover  
onmouseout onmouseenter  
onmouseleave onchange onselect  
onsubmit onkeydown onkeypress  
onkeyup onerror
```

그림 10-4 이벤트 속성 종류와 통합 개발 환경의 속성 자동 완성



03. 문서 객체 조작

1. 이벤트 연결

- 이벤트 모델: 문서 객체에 이벤트를 연결하는 방식.

표 10-5 이벤트 모델 종류(이벤트 연결 방식)

구분	종류
DOM 레벨 0	<ul style="list-style-type: none">• 인라인 이벤트 모델• 고전 이벤트 모델
DOM 레벨 2	<ul style="list-style-type: none">• 마이크로소프트 인터넷 익스플로러 이벤트 모델• 표준 이벤트 모델

- 인라인 이벤트 모델
 - HTML 태그 내부에 자바스크립트 코드를 넣어 이벤트를 연결하는 방식

03. 문서 객체 조작

1. 이벤트 연결

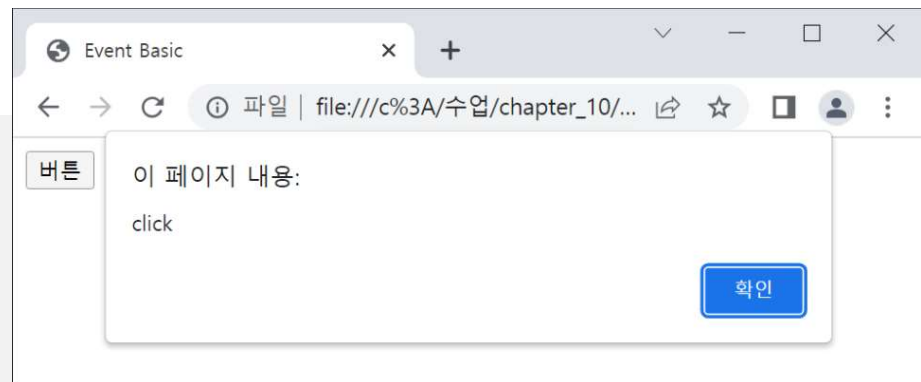
인라인 이벤트 모델 기본 예제 10-7 이벤트 연결

- HTML 태그 내부에 자바스크립트 코드를 넣어 이벤트를 연결하는 방식.

1 인라인 이벤트 모델 사용하기

코드 10-14 event_inline.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Event Basic</title>
</head>
<body>
    <button onclick="alert('click')">버튼</button>
</body>
</html>
```



03. 문서 객체 조작

1. 이벤트 연결

기본 예제 10-7 / 이벤트 연결

2 script 태그에 인라인 이벤트 모델 사용하기

코드 10-15 event_inlineWithScript.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Inline Event</title>
  <script>
    function buttonClick() {
      alert('click');
    }
  </script>
</head>
<body>
  <button onclick="buttonClick()">버튼</button>
</body>
</html>
```

03. 문서 객체 조작

1. 이벤트 연결

고전 이벤트 모델 - 과거에 표준으로 정의되어 많이 사용하던 이벤트 모델

기본 예제 10-8

고전 이벤트 모델

코드 10-16 event_tradition.html

```
...  
<script>  
  // 이벤트를 연결합니다.  
  window.onload = function () {  
    // 문서를 객체를 선택합니다.  
    let button = document.getElementById('button');  
    function () {  
      alert('click');  
      // 이벤트를 연결합니다.  
      button.onclick = function () {  
        alert('click');  
      };  
    };  
  };  
</script>  
</head>  
<body>  
  <button id="button">버튼</button>  
...
```

이벤트 리스너

03. 문서 객체 조작

1. 이벤트 연결

NOTE 이벤트 발생 객체

코드 10-17 note_eventOrigin.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Traditional Event</title>
</head>
<body>
    <button id="button">버튼 - </button>
    <script>
        // 이벤트를 연결합니다.
        document.getElementById('button').onclick = function () {
            this.textContent = this.textContent + '★';
        };
    </script>
</body>
</html>
```

버튼 - ★★★★★★★★★★

03. 문서 객체 조작

2. 이벤트 사용

- 이벤트 객체를 사용하면 이벤트와 관련한 정보를 알아낼 수 있음.

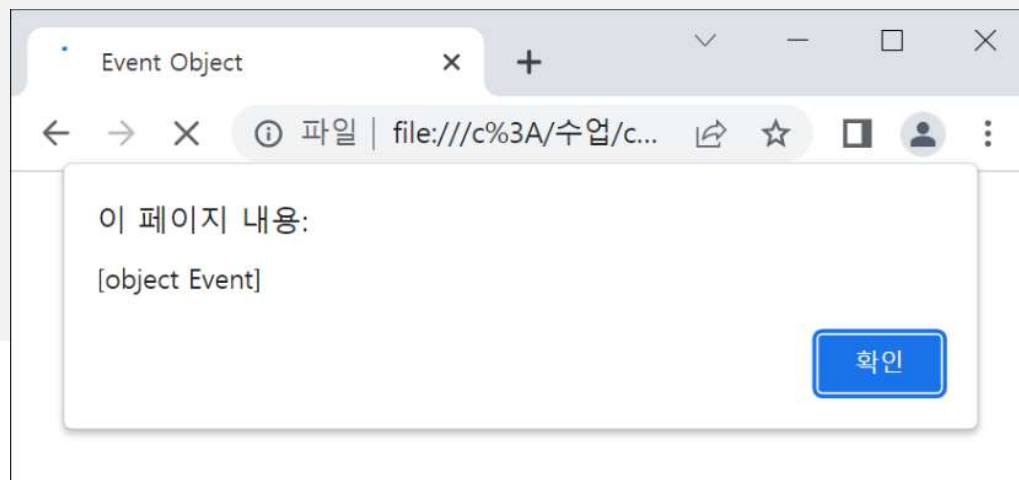
기본 예제 10-9 이벤트 정보

코드 10-18 event_object.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Event Object</title>
  <script>
    window.onload = function (event) {
      alert(event);
    };
  </script>
</head>
<body>

</body>
</html>
```

이벤트 객체



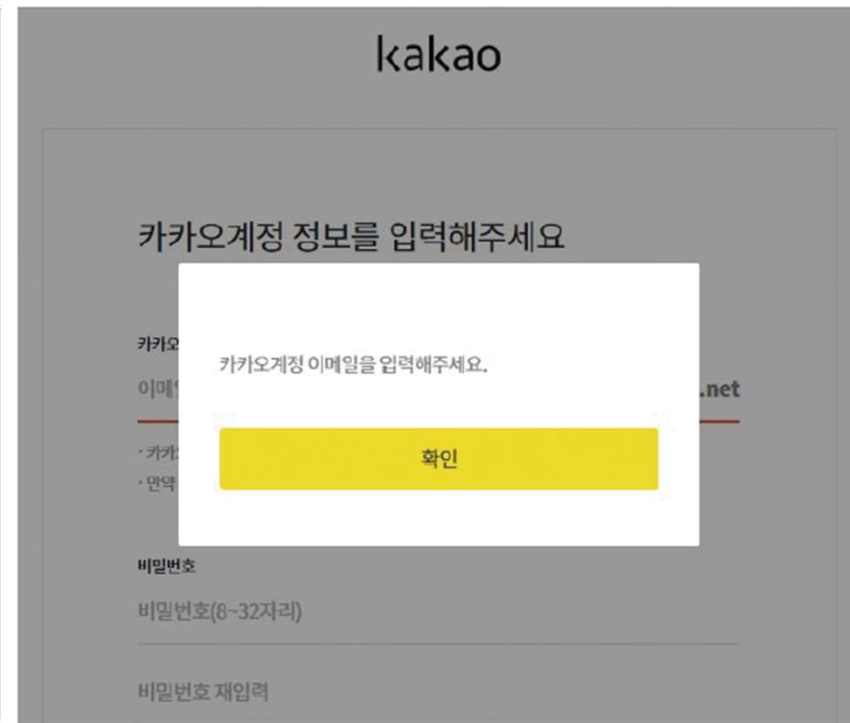
03. 문서 객체 조작

2. 이벤트 사용

- 기본 이벤트: 특정 태그가 가진 기본적인 이벤트.



The image shows the Naver login interface. At the top is the 'NAVER' logo in green. Below it are three tabs: 'ID 로그인' (selected), '일회용 번호', and 'QR코드'. The 'ID 로그인' tab contains a form with an '아이디' (ID) input field, a password input field with a toggle for visibility, and a '로그인 상태 유지' (Keep login state) checkbox. To the right of the password field is an 'IP보안' (IP Security) toggle. Below the password field is a red-bordered box containing the text '아이디를 입력해주세요.' (Please enter your ID). At the bottom is a large green '로그인' (Login) button.



The image shows the Kakao login interface. At the top is the 'kakao' logo. Below it is a title '카카오계정 정보를 입력해주세요' (Please enter your Kakao account information). There are two input fields: '이메일' (Email) and '비밀번호' (Password). The '이메일' field has a placeholder text '카카오계정 이메일을 입력해주세요.' (Please enter your Kakao account email). Below the email field is a yellow '확인' (Confirm) button. The '비밀번호' field has a placeholder text '비밀번호(8~32자리)' (Password (8~32 characters)). Below the password field is a placeholder text '비밀번호 재입력' (Re-enter password).

그림 10-5 기본 이벤트 제거를 사용한 유효성 검사

03. 문서 객체 조작

2. 이벤트 사용

기본 예제 10-10

기본 이벤트 제거

코드 10-19 event_default.html

```
...  
<script>  
  // 이벤트를 연결합니다.  
  window.onload = function () {  
    // 문서 객체를 선택합니다.  
    let button = document.getElementById('button');  
  
    // 이벤트를 연결합니다.  
    button.onclick = function () {  
      // 기본 이벤트를 제거합니다.  
      return false;  
    };  
  };  
</script>  
</head>  
<body>  
  <a id="button" href="http://hanbit.co.kr">버튼</a>  
</body>  
</html>
```



hanbit.co.kr