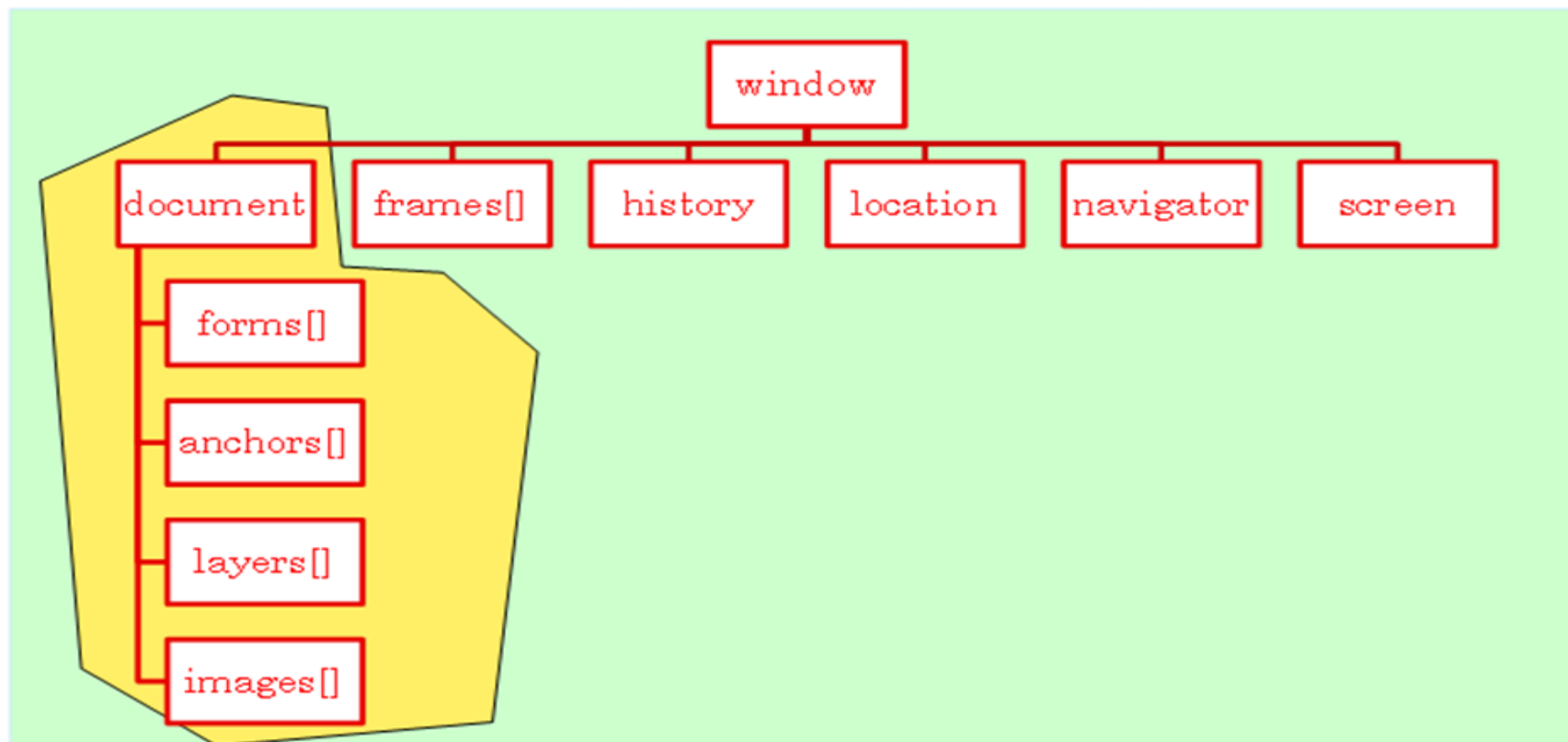


10 BOM과 DOM

DOM/BOM

- HTML 문서를 객체로 표현한 것을 DOM
- 웹브라우저를 객체로 표현한 것을 BOM(Browser Object Model)

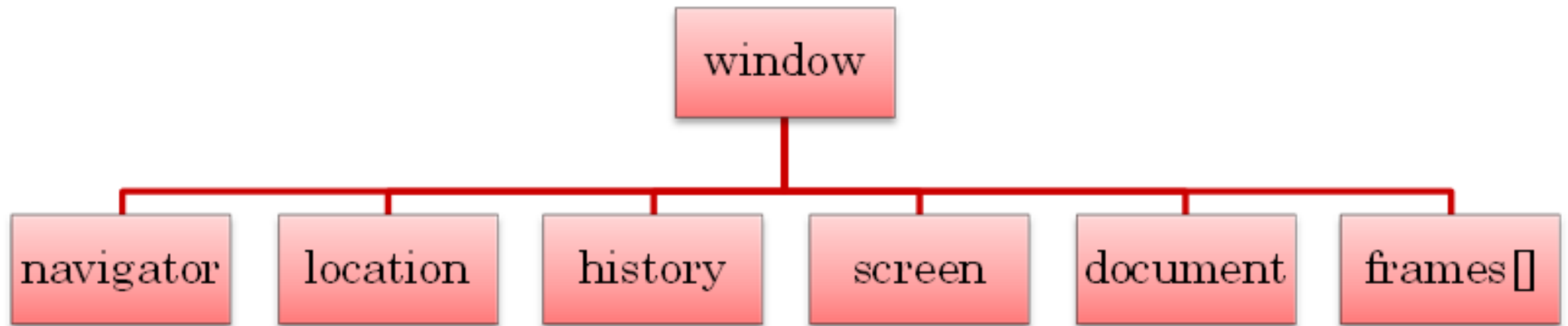
BOM(Browser Object Model)



DOM(Document Object Model)

브라우저 객체 모델(BOM)

- 브라우저 객체 모델(BOM: Browser Object Model): 웹 브라우저가 가지고 있는 모든 객체를 의미
- 최상위 객체는 window이고 그 아래로 navigator, location, history, screen, document, frames 객체가 있다.



Window 객체 (1 / 3)

- 메소드(1/2)

메소드	설명
open()	새로운 창을 연다
close()	열려진 창을 닫는다
alert()	내용을 나타내는 경고 창이 뜬다
confirm()	사용자의 대답을 확인하는 창이 뜬다
prompt()	메시지와 초기값을 나타내고 새로운 값을 입력할 수 있는 창이 뜬다.
setTimeout(fn, millisecond)	주어진 시간이 경과하면 지정된 함수가 호출되어 실행된다
clearTimeout(object)	setTimeout()을 중지 시킨다

Window 객체 (2/3)

- 메소드(2/2)

메소드	설명
setInterval(fn, millisecond)	주어진 시간이 경과할 때마다 지정된 함수가 호출되어 실행된다
clearInterval(id 변수)	setInterval() 메소드를 종료시킨다
moveTo(x, y)	절대적인 위치 x와 y로 이동
moveBy(dx, dy)	상대적인 위치로 x와 y값 만큼 이동
resizeTo(x, y)	x와 y값으로 창의 크기를 재조정
resizeBy(dx, dy)	현재 크기에서 x와 y값 만큼 크기를 재조정.
scrollTo(x, y)	스크롤을 x와 y로 이동
scrollBy(dx, dy)	스크롤을 x와 y값 만큼 이동

Window 객체 (3/3)

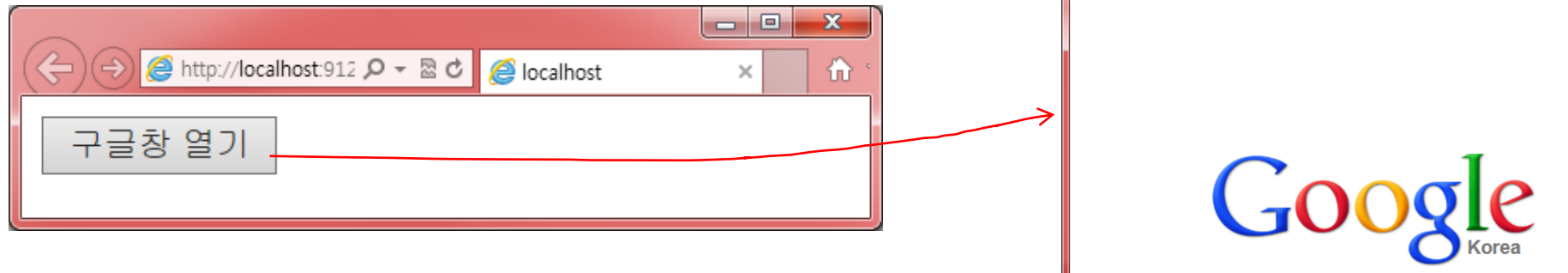
- 속성

속성	설명
opener	open()을 통해 새로운 창을 열었을 때 그 창을 자식창이라 한다면 자식창에서 부모창을 가리킬 때 opener라 한다
innerHeight	ie 5~8 버전에서는 document.documentElement.clientHeight document.body.clientHeight
innerWidth	ie 5~8 버전에서는 document.documentElement.clientWidth document.body.clientWidth

open() 메소드 (1/2)

```
window.open(URL, name, specs);
```

- URL : 오픈할 페이지의 URL
- name : 타겟(target)을 지정하거나 윈도우의 이름
- specs : 여러가지 속성



- 새 창 닫기

```
var pop = window.open(URL, name, specs);  
pop.close();
```

open() 메소드 (2/2)

```
<!DOCTYPE html>
<html>
<head>
<script>
function openPopup() {
    open("http://www.google.co.kr", "_blank", "width=200,height=200");
}
</script>
</head>
<body>
    <input type="button" value="구글창 열기" onclick="openPopup();" >
</body>
</html>
```


setTimeout() 메소드 (1/2)

setTimeout(function, milliseconds)

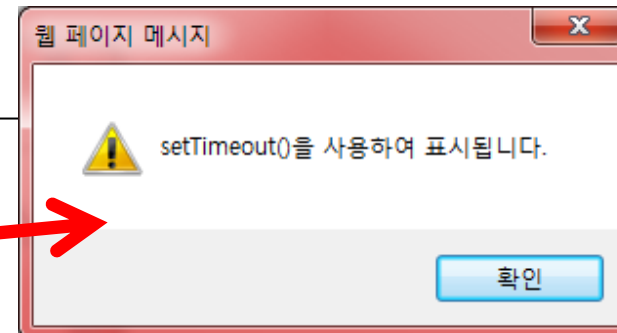
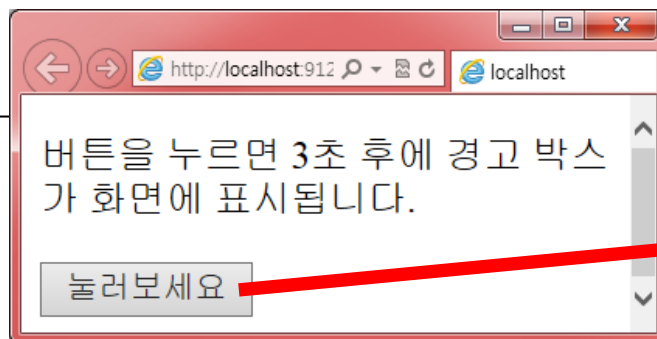
호출되는 함수의 이름
(함수를 직접 정의해도 됨)

함수 호출하기 전에
흘러야 하는 시간



setTimeout() 메소드 (2/2)

```
<!DOCTYPE html>
<html>
<head>
  <script>
    function showAlert() {
      setTimeout(function () {
        alert("setTimeout()을 사용하여 표시됩니다.");
      }, 3000);
    }
  </script>
</head>
<body>
  <p>버튼을 누르면 3초 후에 경고 박스가 화면에 표시됩니다.</p>
  <button onclick="showAlert()">눌러보세요</button>
</body>
</html>
```



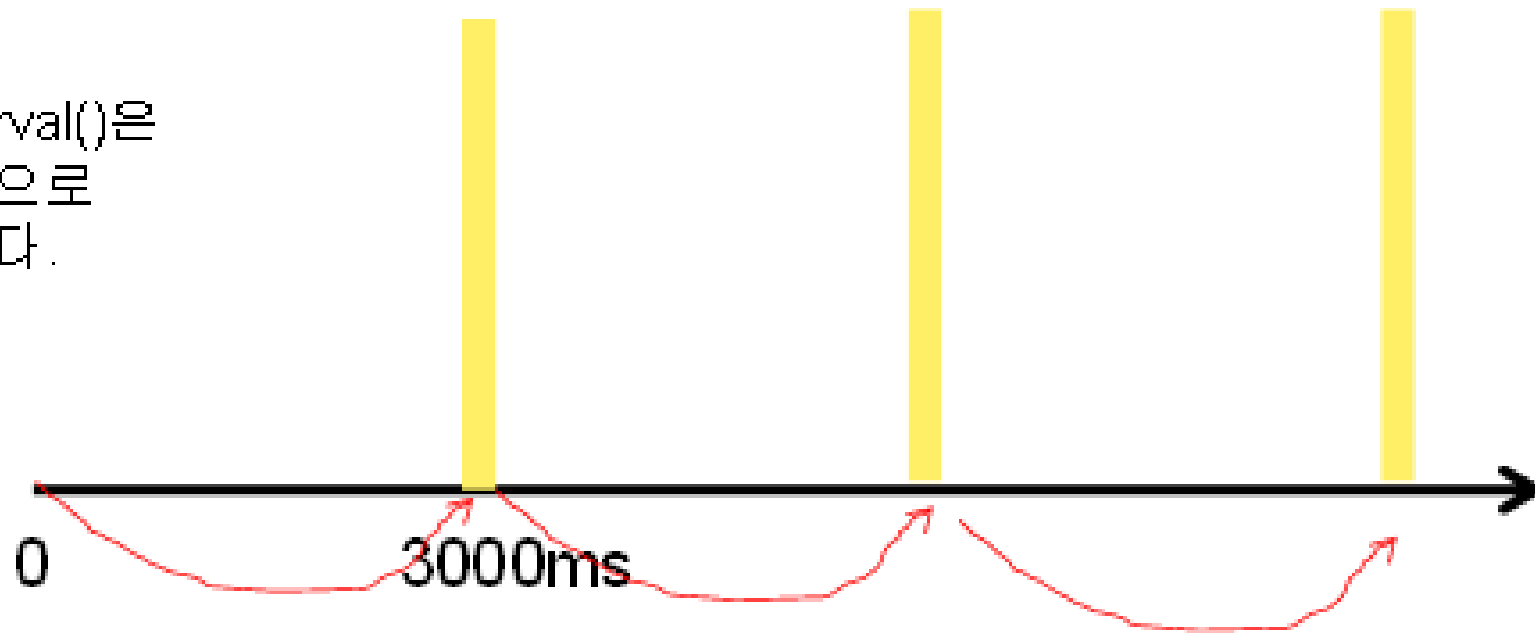
setInterval() 메소드 (1/2)

setInterval(function, milliseconds)

호출되는 함수의 이름
(함수를 직접 정의해도 됨)

함수 호출하기 전에
흘러야 하는 시간

setInterval()은
주기적으로
호출한다.



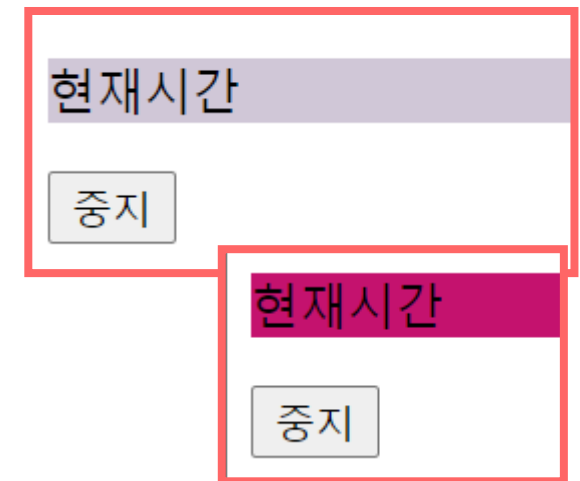
setInterval() 메소드 (2/2)

```
<script>
var tmp = setInterval(changeColor, 3000);

function changeColor() {
  var r = Math.floor(Math.random() * 256);
  var g = Math.floor(Math.random() * 256);
  var b = Math.floor(Math.random() * 256);

  document.getElementById("demo").style.background
    = "rgb(" + r + ", " + g + ", " + b + ")";
}

function stopTextColor() {
  window.clearInterval(tmp);
}
</script>
</head>
<body>
  <p id="demo">배경색</p>
  <button onclick="stopTextColor();">중지</button>
</body>
```



문제

- 1초마다 브라우저 배경색이 랜덤으로 바뀌는 프로그램을 작성하시오.

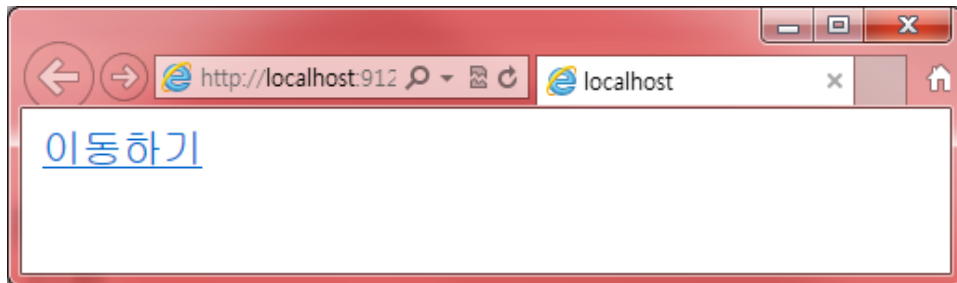
location 객체

속성	설명
hash	URL 중에서 앵커 부분을 반환한다.
host	URL 중에서 hostname과 port를 반환
hostname	URL 중에서 hostname을 반환
href	전체 URL을 반환
pathname	URL 중에서 경로(path)를 반환
port	URL 중에서 port를 반환
protocol	URL 중에서 protocol 부분을 반환
search	URL 중에서 쿼리(query) 부분을 반환

메서드	설명
assign()	새로운 문서를 로드한다.
reload()	현재 문서를 다시 로드한다.
replace()	현재 문서를 새로운 문서로 대체한다. (history초기화)

예제

```
<script>
  function replace() {
    location.replace("http://www.google.com");
  }
</script>
<a href="#" onClick="replace()">이동하기</a>
```

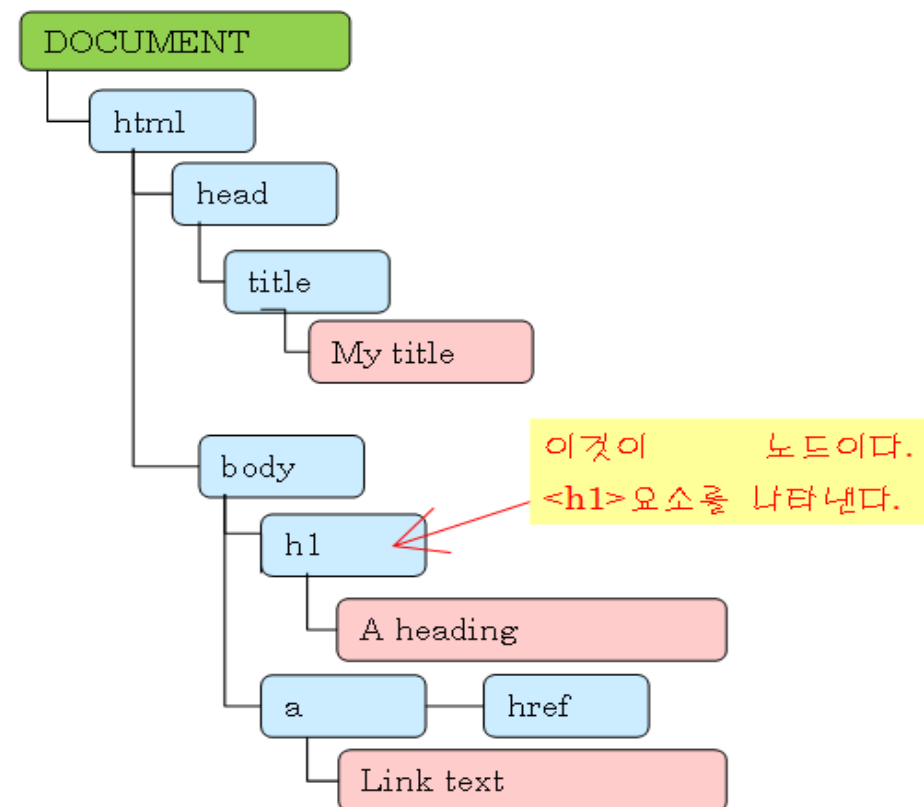


문서 객체 모델(DOM)

- DOM은 W3C (World Wide Web Consortium)의 표준입니다.
- DOM은 문서를 액세스 하기 위한 표준을 정의한다 :
- "W3C 문서 객체 모델 (DOM)은 프로그램 및 스크립트를 동적으로 액세스하여 문서의 콘텐츠, 구조 및 스타일을 갱신 할 수 있도록 플랫폼 및 언어 중립 인터페이스이다."
- DOM은 HTML 문서의 계층적인 구조를 트리(tree)로 표현

```
<html>
  <head>
    <title>My title</title>
  </head>
  <body>
    <h1>A heading</h1>
    <a href="#" >Link text</a>
  </body>
</html>
```

DOM



HTML 요소 찾기

- 동적인 웹페이지를 작성하려면 원하는 요소를 찾아야 한다.
 - 요소 속성 중 id로 찾기

```
<script>  
    document.getElementById("");  
</script>
```

- 요소 속성 중 name으로 찾기

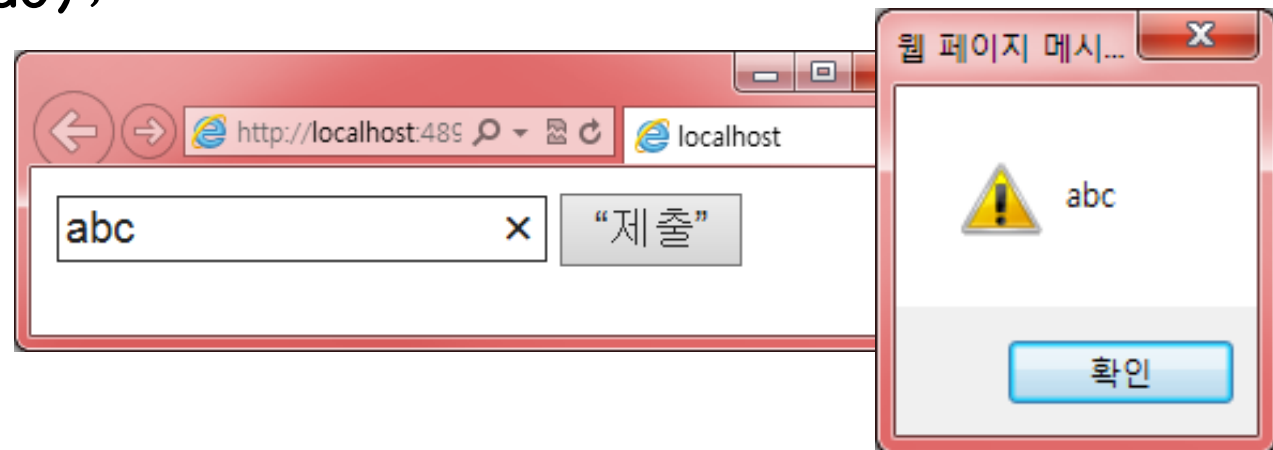
```
<script>  
    document.getElementsByName( "comImg")[0].src;  
</script>
```

- 태그 이름으로 찾기

```
<script>  
    document.getElementsByTagName( "input")[0];  
</script>
```

HTML 요소 찾기

```
<head>
  <script>
    function process() {
      var obj = document.getElementById("target");
      alert(obj);
      alert(obj.value);
    }
  </script>
</head>
<body>
  <form name="myform">
    <input type="text" id="target" name="name">
    <input type="submit" value="제출" onclick="process()">
  </form>
</body>
```



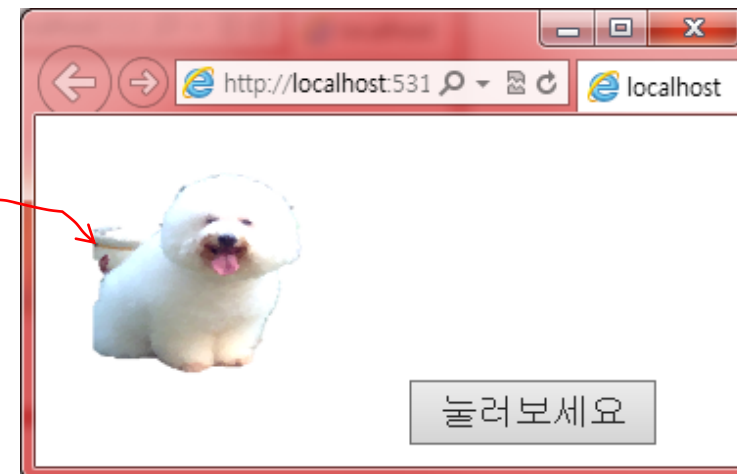
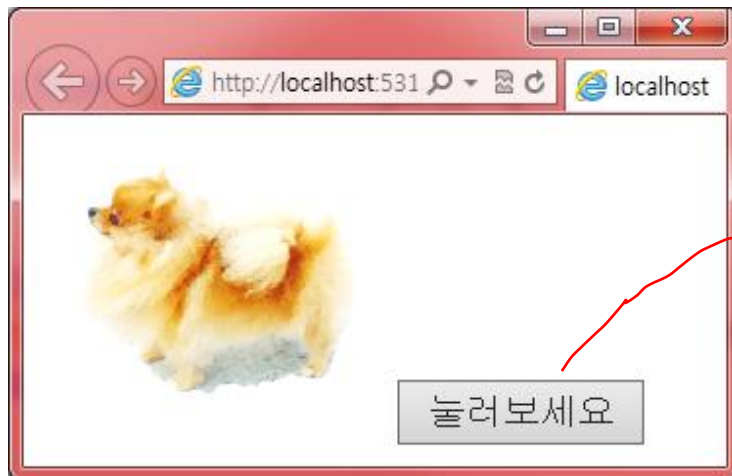
HTML 요소 내용 변경

```
<head>
  <title></title>
  <script>
    function get() {
      var val = document.getElementById("ex").innerHTML;
      alert(val);
    }
    function set(para) {
      document.getElementById("ex").innerHTML = para;
    }

  </script>
</head>
<body>
  <div id="ex">무궁화 꽃이 피었습니다.</div>
  <a href="#" onclick="get()">내용 가져오기</a>
  <br>
  <a href="#" onclick="set( ' 장미꽃이 피었습니다' )">내용 변경하기</a>
</body>
```

HTML 요소 속성 변경

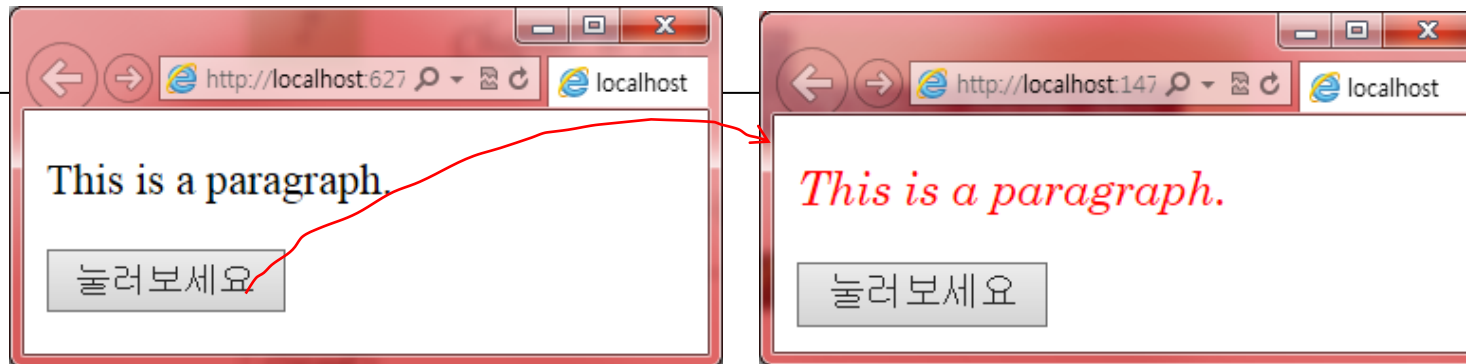
```
<html>
<body>
  
  <script>
    function changeImage() {
      document.getElementById("image").src = "poodle.png";
    }
  </script>
  <input type="button" onclick="changeImage()" value="눌러보세요"/>
</body>
</html>
```



요소의 스타일 변경하기

```
<html>
<body>
<p id="p1">This is a paragraph.</p>
<script>
function changeStyle() {
    document.getElementById("p1").style.color = "red";
    document.getElementById("p1").style.fontFamily = "Century Schoolbook";
    document.getElementById("p1").style.fontStyle = "italic";
}
</script>

<input type="button" onclick="changeStyle()" value="눌러보세요" />
</body>
</html>
```



연습

- 다음 화면과 같이 이미지를 클릭하면 이미지가 스위치가 켜지고 다시 클릭하면 스위치가 꺼지는 프로그램을 작성하십시오.(토글 화면)



DOM 트리 순회

- document : HTML 문서의 루트 노드. DOM의 진입점
- document.documentElement → <html>
- document.head → <head>
- document.body → <body>

속성	내용
parentElement	현재 노드의 부모 요소를 반환
children	모든 자식 요소를 반환
firstElementChild	첫 번째 요소 반환 텍스트는 요소(element)로 취급하지 않음
lastElementChild	마지막 요소 반환 텍스트는 요소(element)로 취급하지 않음
nextElementSibling	현재 노드의 다음 형제 요소 반환 (ie8 이하에서 X)
previousElementSibling	현재 노드의 이전 형제 요소 반환

DOM 트리 순회

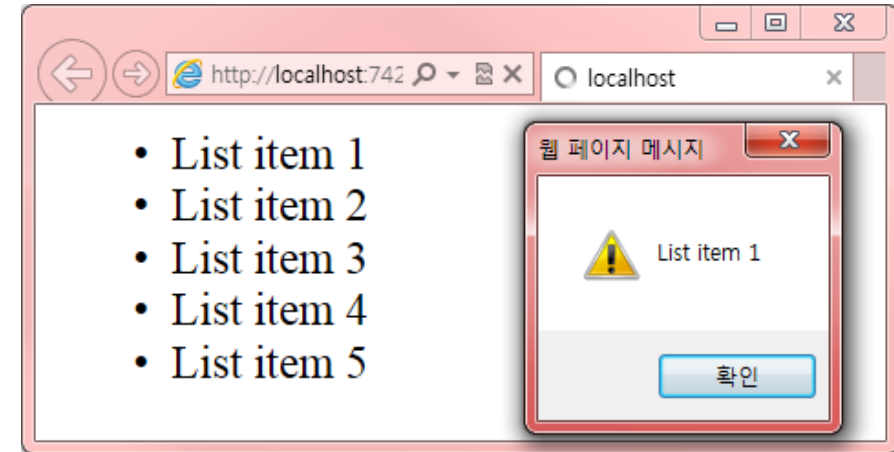
속성	내용
parentNode	현재 노드의 부모 노드를 반환
childNodes	모든 자식 노드를 반환
firstChild	첫 번째 자식 노드 반환 (childNodes[0]와 동일) 텍스트도 하나의 노드로 취급
lastChild	마지막 자식 노드 반환 (childNodes.childNodes.length - 1) 동일)
nextSibling	현재 노드의 다음 형제 노드 반환
previousSibling	현재 노드의 이전 형제 노드 반환

속성	내용
nodeName	노드 이름 반환
nodeValue	노드의 값 반환(텍스트는 항상 Text node 안에 존재). 유사 : data
tagName	요소 이름 반환
textContent	텍스트 반환 (하위 항목의 텍스트도 포함)

DOM - firstChild

```
<html>
<body>
  <ul>
    <li>List item 1</li>
    <li>List item 2</li>
    <li>List item 3</li>
    <li>List item 4</li>
    <li>List item 5</li>
  </ul>
  <script>
    var list = document.getElementsByTagName('ul')[0];
    var allItems = list.getElementsByTagName('li');

    for (var i = 0, length = allItems.length; i < length; i++) {
      alert(allItems[i].firstChild.data);
    }
  </script>
</body>
</html>
```



HTML 요소 생성

- 텍스트 노드를 갖는 요소와 갖지 않는 요소로 구분
- 요소노드와 텍스트 노드를 생성한 후에 텍스트 노드를 요소 노드에 붙임

메소드	설명
<code>createElement(tagName)</code>	태그요소 생성
<code>createTextNode(text)</code>	텍스트 노드 생성
<code>appendChild(node)</code>	새로운 노드를 마지막에 추가
<code>insertBefore(node, element)</code>	새로운 노드를 기존 요소 앞에 추가
<code>removeChild(node)</code>	노드를 삭제한다
<code>remove()</code>	노드삭제

요소 생성 예제1

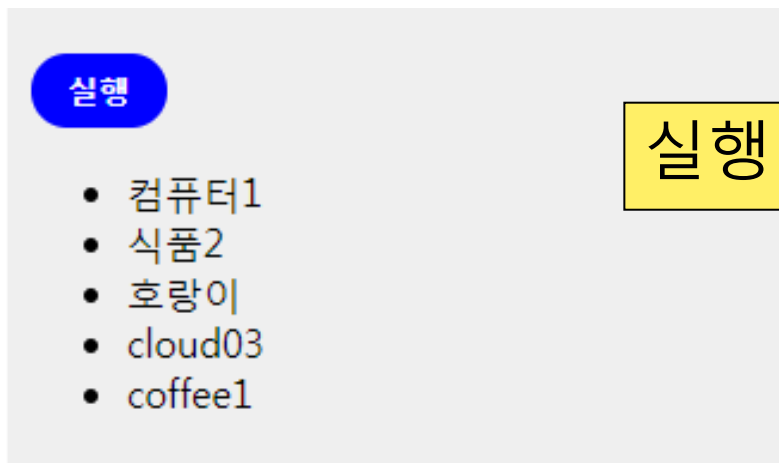
- 기존의 요소(div)에 텍스트 노드 생성해서 추가하기
 - createTextNode(*string*)
 - appendChild(*element*)

```
<script>
  function addtext(t) {
    var node = document.createTextNode(t);
    document.getElementById("target").appendChild(node);
  }
</script>
```

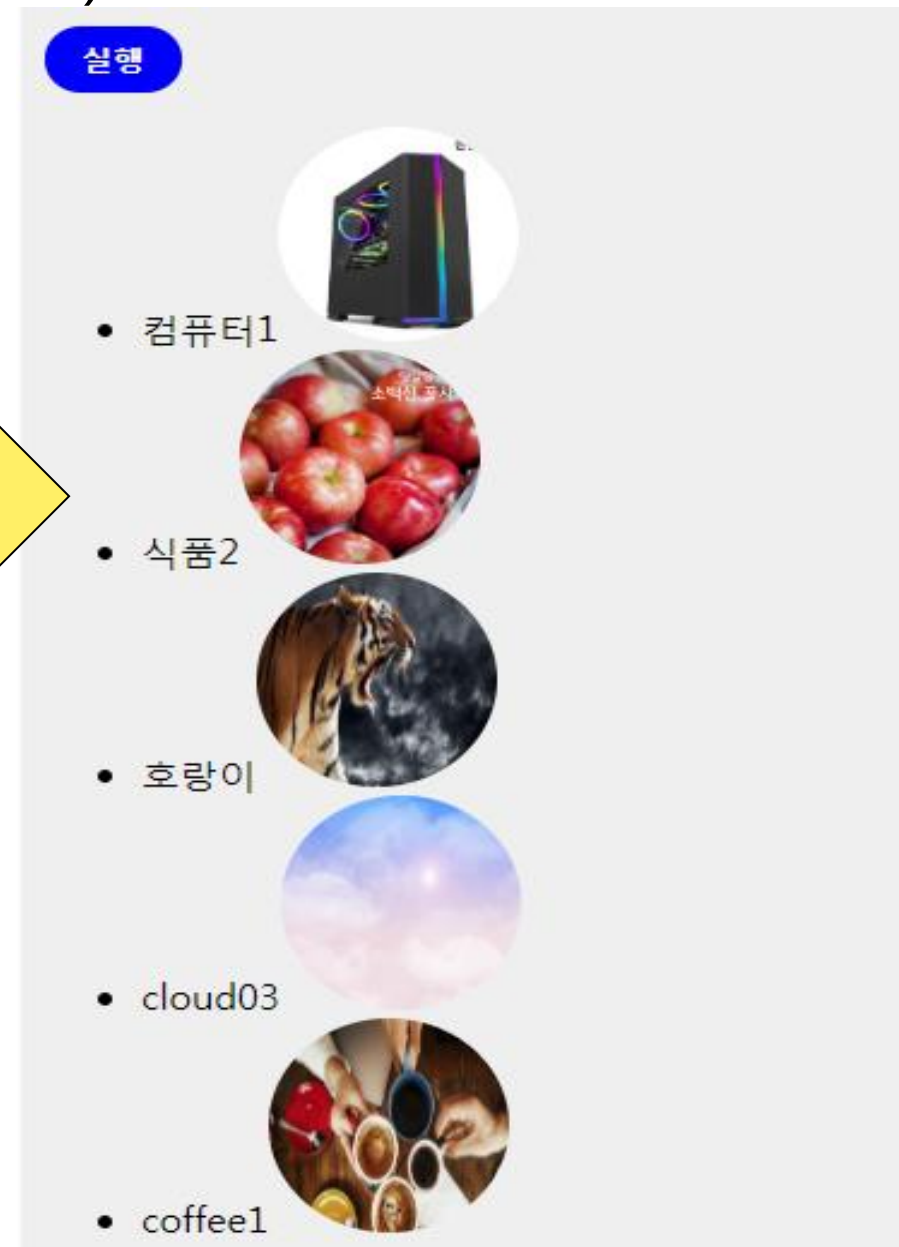
```
<div id="target" style="font: 20px bold; "
  onclick="addtext( ' 무궁화 꽃이 피었습니다.' )">
여기를 클릭하세요.
</div>
```

요소 생성 예제2

- 리스트에 새로운 이미지 태그 만들어 추가하기
 - `document.getElementsByTagName("li")`
 - `firstChild.data`
 - `createElement("img")`
 - `appendChild(e/e)`



실행 버튼 클릭 시



요소 삭제 예제

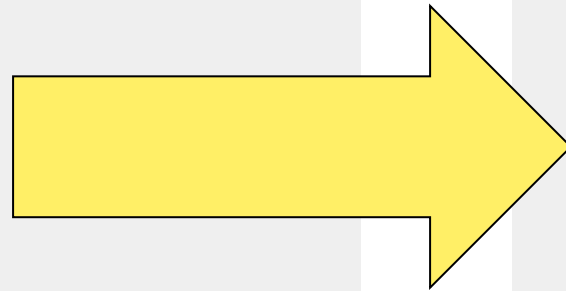
- 요소 삭제하기
 - `removeChild(element)`

```
...  
<script>  
function removeNode() {  
    var parent = document.getElementById("target");  
    var child = document.getElementById("p1");  
    parent.removeChild(child);  
}  
</script>  
<body>  
    <div id="target">  
        <p id="p1">첫번째 단락</p>  
        <p id="p2">두번째 단락</p>  
    </div>  
    <button onclick="removeNode()">누르세요!</button>  
</body>  
</html>
```

연습

- 각 리스트 요소를 클릭하면 해당 이미지 표시

- 컴퓨터1
- 식품2
- 호랑이
- cloud03
- coffee1



- 컴퓨터1
- 식품2
- 호랑이
- cloud03

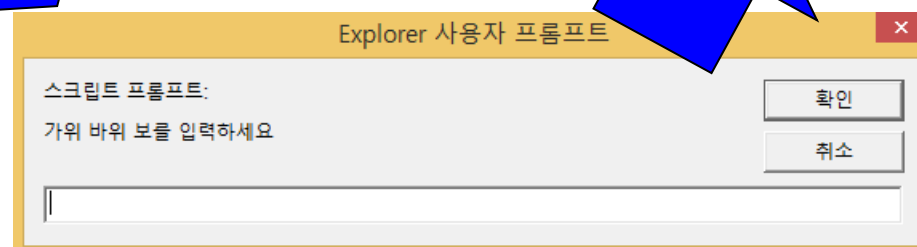
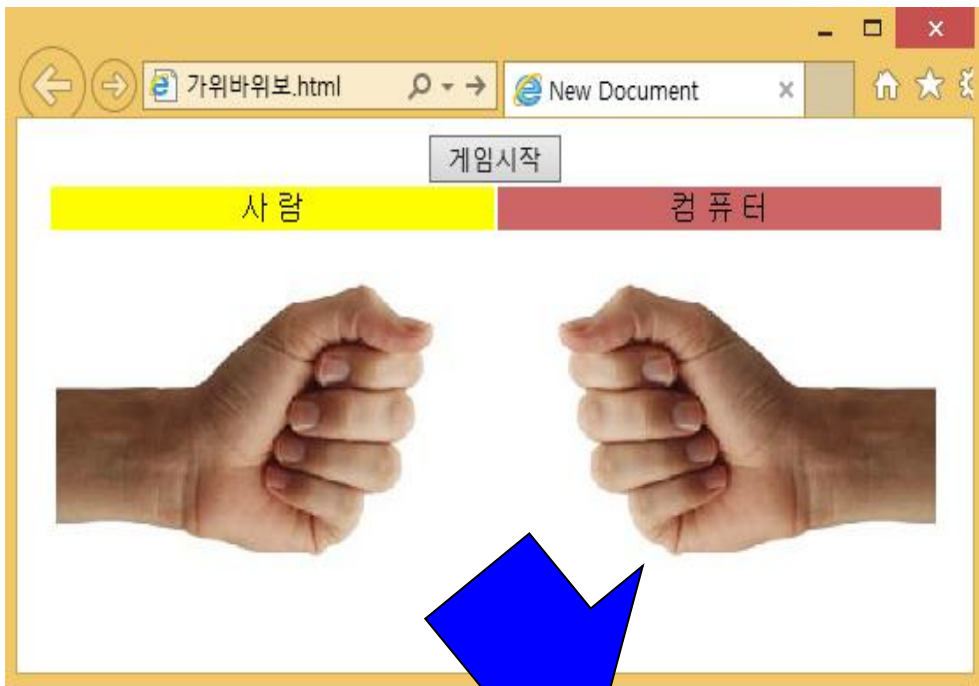


- coffee1

- document.getElementsByTagName("~")
- firstChild.data
- createElement(~)
- appendChild(~)
- parent.removeChild(child)
- remove()

문제

- 이미지를 이용하여 가위 바위 보 게임을 다음과 같이 완성 하시오.
(게임 시작 버튼을 누르면 프롬프트 창이 열리고 가위바위보 중 하나를 입력하면 그 결과를 출력하는 프로그램)



form 문제1

- 다음과 같은 프로그램을 작성하시오. (textarea)

The screenshot shows a web browser window with two tabs: '구구단.html' and 'New Document'. The '구구단.html' tab is active. The page content includes a text input field containing the number '8', followed by the Korean character '단' (dan), and a button labeled '출력' (Output). Below these elements is a text area containing the text '8단' followed by a list of multiplication results for 8 multiplied by numbers 1 through 9.

8 단 출력

8단

8 * 1 = 8
8 * 2 = 16
8 * 3 = 24
8 * 4 = 32
8 * 5 = 40
8 * 6 = 48
8 * 7 = 56
8 * 8 = 64
8 * 9 = 72

form 문제2

- 다음과 같은 입력 양식을 만들고 결과를 출력하는 프로그램을 작성하시오.(취미가 없을 경우는 "취미가 없군요"를 출력)

The screenshot shows a web browser window with two tabs. The active tab is titled "E:\강의자료\HTML\스크립트연습\W38_check_radio문제" and displays a form. The form contains the following elements:

- A text input field labeled "이름:" with the value "셈" entered.
- A radio button group labeled "성별:" with "남" (Male) selected and "여" (Female) unselected.
- A list of hobbies with checkboxes: "취미:" followed by "여행" (checked), "독서" (unchecked), "영화감상" (checked), "바둑" (unchecked), "장기" (unchecked), "등산" (checked), and "낚시" (unchecked).
- A button labeled "결과출력" (Output Result).
- A large text area below the button displaying the output: "셈씨! 당신은 남자이고, 취미는 여행, 영화감상, 등산이군요!" (Mr. Sem! You are a man, and your hobbies are travel, movie appreciation, and hiking!).

form 문제 3

- 다음과 같이 데이터를 교환하는 프로그램을 만드시오.
 - Select객체

select객체 예제

4
5
2
3

>
>>
<
<<

1
6
7

8

추가하기

innerHTML 문제

- <select>와 innerHTML을 이용하여 다음과 같이 만드시오.

