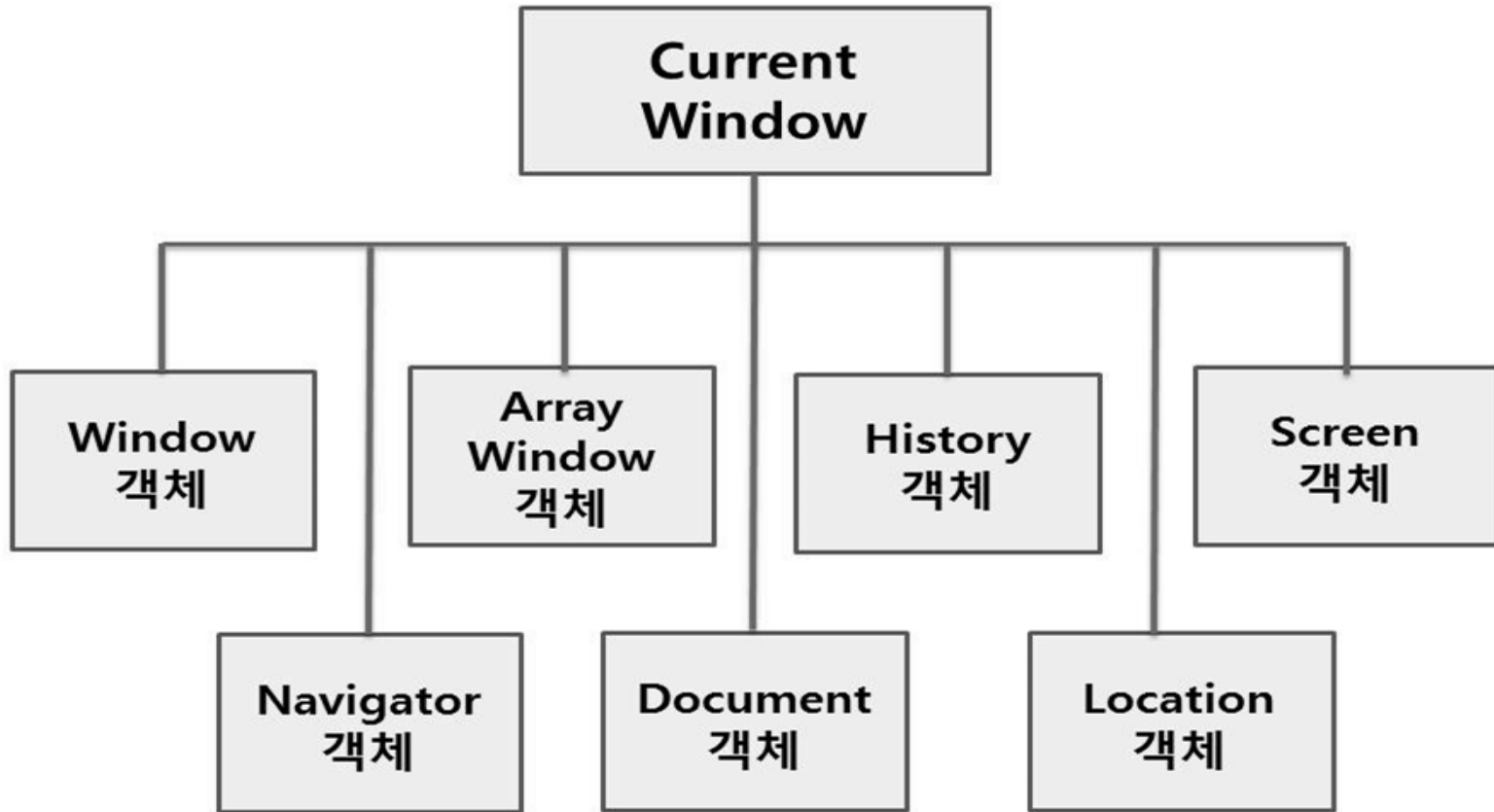


JavaScript 객체

BOM 객체

BOM 객체

- Browser Object Model(BOM)
 - window, navigator, frames[], document, history, location, screen 객체들



BOM 객체

- navigator 객체
 - Browser에 대한 속성과 메소드를 저장

속성	기능 설명
appCodeName	브라우저의 코드명
appName	브라우저명
appVersion	브라우저의 버전
mimeTypes	브라우저에서 제공되는 MINE 형의 배열
platform	브라우저가 사용되는 운영체제
userAgent	브라우저에서 서버로 전송하는 HTTP user-agent 헤더
cookieEnabled	브라우저에서 쿠키를 사용할 수 있는지 여부
geolocation	사용자 위치 정보(Geolocation) 객체
language	브라우저의 사용언어
onLine	브라우저가 온라인 상태인지 여부
product	브라우저의 제품 정보

BOM 객체

- navigator 객체
 - 메서드

메소드	기능 설명
javaEnabled()	java 허용여부
taintEnabled()	javascript 1.2부터 삭제됨, 데이터 수정 가능여부

```
for (var name in navigator){  
    console.log(name + ": " + navigator[name]);  
}
```

```
appCodeName: Mozilla  
appName: Netscape  
appVersion: 5.0 (Windows)  
language: en-US  
mimeTypes: [object MimeTypeArray]  
platform: Win32  
oscpu: Windows NT 6.1; WOW64  
vendor:  
vendorSub:  
product: Gecko  
productSub: 20100101  
plugins: [object PluginArray]  
userAgent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:20.0) Gecko/20100101 Firefox/20.0  
cookieEnabled: true  
onLine: true  
buildID: 20130409194949  
doNotTrack: unspecified  
mozPower: null  
javaEnabled: function javaEnabled() {  
    [native code]  
}  
taintEnabled: function taintEnabled() {  
    [native code]  
}  
vibrate: function vibrate() {  
    [native code]  
}  
addIdleObserver: function addIdleObserver() {  
    [native code]  
}  
removeIdleObserver: function removeIdleObserver() {  
    [native code]  
}
```

```
requestWakeLock: function requestWakeLock() {  
    [native code]  
}  
getDeviceStorage: function getDeviceStorage() {  
    [native code]  
}  
geolocation: [object GeoGeolocation]  
registerContentHandler: function registerContentHandler() {  
    [native code]  
}  
registerProtocolHandler: function registerProtocolHandler() {  
    [native code]  
}  
mozIsLocallyAvailable: function mozIsLocallyAvailable() {  
    [native code]  
}  
battery: [object BatteryManager]  
mozSms: null  
mozGetUserMediaDevices: function mozGetUserMediaDevices() {  
    [native code]  
}  
mozGetUserMedia: function mozGetUserMedia() {  
    [native code]  
}  
mozConnection: [object MozConnection]  
mozCameras: null
```

BOM 객체

```
<html>
  <head>
    <title>Navigator 객체</title>
  </head>
  <body>
    <h2>Browser 속성들</h2>
    <script>
      var BrowserName= navigator.appName;
      var BrowserVersion = navigator.appVersion;
      var BrowserAgent= navigator.userAgent;
      var platform=navigator.platform;
      document.write("<b>Browse의 이름 :</b> " +BrowserName + "<br>");
      document.write("<b>Browser 버전 :</b> " +BrowserVersion + "<br>");
      document.write("<b>Browser agent :</b> " +BrowserAgent + "<br>");
      document.write("<b>Browser 사용 운영체제 :</b> " +platform + "<br>");
    </script>
  </body>
</html>
```

Browser 속성들

Browse의 이름 : Netscape

Browser 버전 : 5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/45.0.2454.99 Safari/537.36

Browser agent : Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/45.0.2454.99 Safari/537.36

Browser 사용 운영체제 : Win32

BOM 객체

- window 객체
 - 웹 브라우저가 제공하는 객체 중에 최상위 객체
 - 객체 계층 구조상 최상위 객체 이므로, 대부분 Window 속성들을 참조할 때 속성 앞에 window를 붙이지 않아도 상관없음
 - window 대신에 self나 this로 표현
 - 다음은 window 객체의 defaultStatus 속성을 참조하는 것으로 동일한 효과

window.defaultStatus
defaultStatus
self.defaultStatus
this.defaultStatus

- 윈도우를 열거나 닫을 때 window 객체를 이용하여 브라우저의 모든 행동을 제어할 수 있음

BOM 객체

- window 객체 속성

속성	기능 설명
status	브라우저의 상태바에 문자열을 설정하거나 읽기
defaultStatus	브라우저의 상태바에 초기 문자열을 설정하거나 읽기
length	현재 Window에 <iframe>엘리먼트의 개수
name	현재 Window의 이름을 설정하거나 읽기
self	현재 Window 자신, window와 같음
window	현재 Window 자신, self와 같음
parent	현재 Window의 부모 Window
opener	현재 Window를 open한 Window
document	현재 Window의 document 객체
frames	현재 Window의 모든 <iframe>엘리먼트 객체
history	현재 Window의 history 객체
location	현재 Window의 location 객체
closed	Window가 닫힌 상태인지 여부
locationbar	로케이션 바
menubar	메뉴 바
personalbar	퍼스널 바
scrollbars	스크롤 바
statusbar	상태 바
toolbar	툴 바

BOM 객체

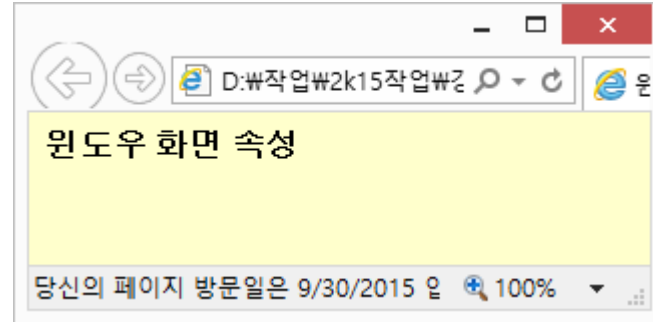
- window 객체 속성

속성	기능 설명
innerHeight	Contents 영역 안쪽 높이(픽셀), IE 지원되지 않음
innerWidth	Contents 영역 안쪽 너비(픽셀), IE 지원되지 않음
outerHeight	툴바, 스크롤바 포함한 Window의 바깥쪽 둘레의 높이, IE 지원되지 않음
outerWidth	툴바, 스크롤바 포함한 Window의 바깥쪽 둘레의 너비, IE 지원되지 않음
pageXOffset	현재 페이지의 좌상단을 기준으로 한 수직 스크롤이 된 위치의 X좌표를 반환, IE 지원되지 않음
pageYOffset	현재 페이지의 좌상단을 기준으로 한 수직 스크롤이 된 위치의 Y좌표를 반환, IE 지원되지 않음
screenX	스크린(모니터)에서 window의 x 좌표를 반환
screenY	스크린(모니터)에서 window의 y 좌표를 반환
screenLeft	screenX와 동일
screenTop	screenY와 동일
scrollX	pageXOffset와 동일
scrollY	pageYOffset와 동일
frameElement	현재 Window에 삽입된 <iframe>엘리먼트 객체

- window 객체 속성

```
<html>
<head>
  <title> 윈도우 속성</title>
  <h4> 윈도우 화면 속성 </h4>
  <SCRIPT>
    function dateinbar() {
      var d = new Date();
      var y = d.getFullYear();
      var m = d.getMonth() + 1;
      var d = d.getDate();
      var t = m + '/' + d + '/' + y + ' ';
      defaultStatus = "당신의 페이지 방문일은 " + t + "입니다.";
    }
  </SCRIPT>
</head>
<BODY BGCOLOR="FFFFcc" onLoad="dateinbar()">

</body>
</html>
```



BOM 객체

- window 객체 메소드

메소드	기능 설명
alert()	메시지와 OK버튼을 가진 경고 대화상자 표시
atob()	base-64 엔코딩된 문자열 복호화
blur()	focus를 다른 곳으로 이동
btoa()	base-64로 문자열을 인코딩
close()	현 Window를 닫음
confirm()	메시지와, 확인, 취소를 선택가능한 대화상자 표시
focus()	다른 곳에서 현재 엘리먼트로 focus를 가져옴
open()	새로운 Browser Window를 열기
print()	화면에 있는 내용을 프린터로 출력
prompt()	입력창이 있는 대화상자 표시
eval()	문자열을 평가하여 결과값을 반환
stop()	로딩 중지. IE 지원 안함

BOM 객체

- window 객체 메소드

메소드	기능 설명
moveBy()	Window를 상대적인 좌표로 이동. 수평방향과 수직방향의 이동량을 픽셀로 지정
moveTo()	Window를 절대적인 좌표로 이동. 창의 왼쪽 상단 모서리를 기준으로 픽셀을 지정
resizeBy()	Window의 크기를 상대적인 좌표로 재설정. 밑변의 모서리를 기준으로 수평방향, 수직방향을 픽셀로 지정
resizeTo()	Window의 크기를 절대적인 좌표로 재설정. 창 크기를 픽셀로 지정
scrollBy()	Window를 상대적인 좌표로 스크롤. 창의 표시 영역의 수평방향과 수직방향에 대해 픽셀로 지정
scrollTo()	Window를 절대적인 좌표를 스크롤. 창의 왼쪽 상단 모서리를 기준으로 픽셀로 지정

BOM 객체

- window 객체 메소드

메소드	기능 설명
captureEvents()	모든 타입의 이벤트를 판단
handleEvent()	이벤트 핸들러를 정함
releaseEvent()	다른 계층의 이벤트로 이벤트를 넘김
routeEvent()	판단한 이벤트와 같은 계층의 이벤트
setTimeout()	일정 간격 함수를 호출하여 수행, millisecond 단위로 지정
clearTimeout()	setTimeout 메소드를 정지
setInterval()	일정시간마다 지정된 처리를 반복
clearInterval()	setInterval 메소드의 정지

BOM 객체

- window 객체 메소드 `window.open(URL, name, specs, replace)`

- URL: 값 설정하지 않으면 about:blank window가 open됨
- name: Window이름 설정
 - _blank: 디폴트, 새 Window open
 - _parent: 부모 프레임에 로드됨
 - _self: 현재 페이지에 URL의 페이지가 로드됨
 - _top: 표시 가능한 frameset들 중 하나에 로드됨
- specs: open될 window 모양 설정

<code>channelmode=yes no 1 0</code>	window를 극장(theater) 모드로 설정. 디폴트: no. IE only
<code>directories=yes no 1 0</code>	없어질 예정. 연결도구모음 표시 여부. 디폴트: yes. IE only
<code>fullscreen=yes no 1 0</code>	window를 full-screen 모드로 설정. 디폴트: no. IE only. 극장모드가 yes일 때 사용가능
<code>height=pixels</code>	window의 높이. 최소값: 100
<code>left=pixels</code>	window의 왼쪽 위치값. 양수값만
<code>location=yes no 1 0</code>	주소입력란 표시여부. Opera only
<code>menubar=yes no 1 0</code>	menu bar 표시여부
<code>resizable=yes no 1 0</code>	window 크기 조절가능 여부. IE only
<code>scrollbars=yes no 1 0</code>	scroll bar의 표시 여부. IE, Firefox & Opera only
<code>status=yes no 1 0</code>	status bar 표시 여부
<code>titlebar=yes no 1 0</code>	title bar 표시여부 . Ignored unless the calling application is an HTML Application or a trusted dialog box
<code>toolbar=yes no 1 0</code>	toolbar 표시여부. IE and Firefox only
<code>top=pixels</code>	window의 윗쪽 위치값. 양수값만
<code>width=pixels</code>	window의 너비. 최소값: 100

- replace: true-history list를 업데이트, false-history list에 URL을 새항목으로 추가

BOM 객체

- window 객체 메소드

```
<html>
<head>   <title> 윈도우 속성</title> </head>
<body>
  <h4> 윈도우 닫기/열기/인쇄 메소드 사용 </h4>
  <SCRIPT>
    var windo=null;
    function inWindow() {
      windo=window.open("예제.html");
    }
    function outWindow() {
      if(windo != null)
        windo.close();
    }
  </SCRIPT>
  <form>
    <input type="button" value="윈도우 open" onClick="inWindow()" />
    <input type="button" value="윈도우 exit" onClick="outWindow()" />
    <input type="button" value="내용 인쇄하기" onClick="window.print()" />
  </form>
</body>
</html>
```

윈도우 닫기/열기/인쇄 메소드 사용

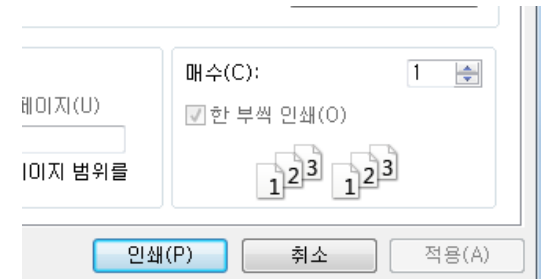
윈도우 open

윈도우 exit

내용 인쇄하기

윈도우 화면 속성

당신의 페이지 방문일은 8/24/2011 입니다.



BOM 객체

- window 객체 메소드

<html>

<head> <title>새로운 윈도우에 이미지 열기/닫기</title>

<script>

function newWindow() {

winObj=window.open("winter.jpg","겨울배경", "width='548px',
height='361px'", "resizable=yes, scrollbars=yes,
location=yes");

winObj.focus(); //winObj.blur();

}

function closeWindow(){ winObj.close(); }

</script>

</head>

<body bgColor="lightblue">

<h2>시골의 겨울배경입니다!!</h2>

<h3>연결되어 있는 겨울 배경 단추를 클릭하세요!!

겨울배경 열기

<p>겨울배경을 닫으실려면 이 단추를 클릭하세요!!

겨울배경 닫기

</h3>

</body>

</html>

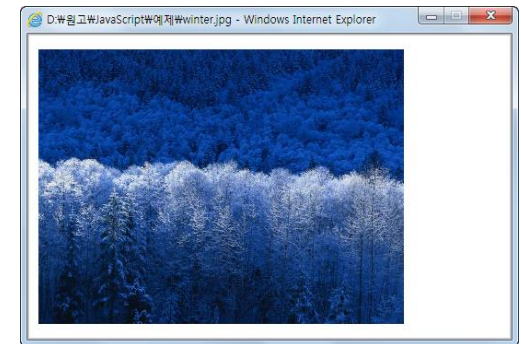
시골의 겨울배경입니다!!

연결되어 있는 겨울 배경 단추를 클릭하세요!!

겨울배경 열기

겨울배경을 닫으실려면 이 단추를 클릭하세요!!

겨울배경 닫기



BOM 객체

- window 객체 메소드

```
<html>
```

```
  <head>    <title>열리는 문서의 이동</title> </head>
```

```
<body bgColor="lightblue">
```

```
  <script>
```

```
    var winObj=null
```

```
    function directions(){
```

```
      winObj=window.open("예제6-39.html","예제639",  
                          "width=200,height=300,resizable=no");
```

```
      winObj.moveTo(0, 0); // 윈도우의 왼쪽 상단으로 이동
```

```
      winObj.focus();
```

```
      parent.window.moveTo(215, 0); // 상단 모서리로 기준 지정
```

```
      parent.window.resizeTo(400,400); // 윈도우 크기 변경
```

```
    }
```

```
    function closeWindow(){    winObj.close();    }
```

```
  </script>
```

```
<h2>열리는 윈도우의 이동!!</h2> 새로운 크기 변경<br />  버튼을 클릭하세요!!
```

```
  <form ><input type="button" value="예제6-39.html" onClick="directions();">
```

```
  <p>문서가 열렸으면 닫기 단추를 누르세요!!<br /> </form>
```

```
<a href="javascript:closeWindow()">문서 닫기</a> </body> </html>
```


BOM 객체

- window 객체 메소드

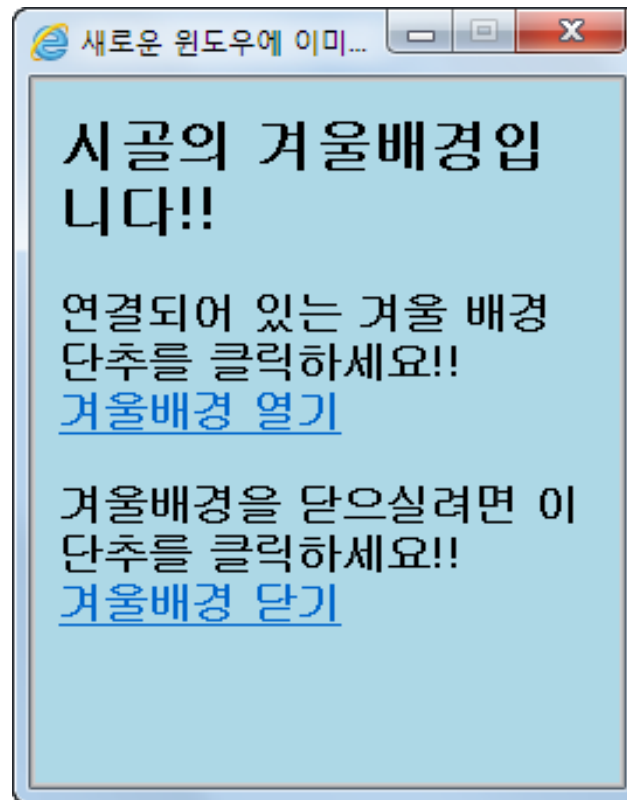
열리는 윈도우의 이동!!

새로운 크기 변경
버튼을 클릭하세요!!

예제6-39.html

문서가 열렸으면 닫기 단추를 누르세요!!

[문서 닫기](#)



BOM 객체

- location 객체
 - 현재 Window의 URL에 대한 정보를 저장
 - location 객체의 속성

속성	기능 설명
hash	url 주소의 anchor part (#)를 설정하거나 읽기
host	url 주소의 호스트 이름과 포트를 설정하거나 읽기
hostname	url 주소의 호스트 이름이나 호스트의 ip 주소 설정 및 읽기
href	문서의 전체 url 주소를 설정하거나 읽기
origin	url 주소의 프로토콜, 호스트이름, 포트번호를 반환
pathname	url 주소에서 경로부분을 설정하거나 읽기
port	포트번호를 설정하거나 읽기
protocol	프로토콜 종류를 설정하거나 읽기
search	url 주소의 쿼리 문자열 부분을 설정하거나 읽기

- location 객체의 메소드

메소드	기능 설명
toString()	location.href를 반환.
assign()	location.href를 설정.
reload()	문서를 새로 고치거나 다시 읽음.
replace()	현재 문서를 URL 문서로 대체.

BOM 객체

- location 객체

```
<!DOCTYPE html>                                location-main.html
<html>
<head>
  <meta charset="utf-8">
  <title>iframe 메인 화면</title>
</head>
<body>
  <iframe src="location.html" width=" 100%" height="130"></iframe>
  <iframe src="empty.html" width=" 100%" height="900"></iframe>
</body> </html>
```

```
<!DOCTYPE html>                                empty.html
<html>
<head>
  <meta charset="utf-8">
  <title>Empty Frame</title>
</head>
<body>
  <h1>이 웹페이지는 iframe 방식으로 표시됩니다.</h1>
</body> </html>
```

BOM 객체

- location 객체

location.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>History Object</title>
</head>
<body bgcolor="lightblue">
  <script>
    function loadPage(urlAddress) { parent.frames[1].location.href = urlAddress; }
  </script>
  <big>
    <form name="form1" id="form1">
      <input type="button" value="영진" onClick="loadPage('http://www.yjc.ac.kr');">
      <input type="button" value="다음"
        onClick="loadPage('http://www.daum.net');">
      <input type="button" value="네이트"
        onClick="loadPage('http://www.nate.com');">
    </form>
  </big>
</body>
</html>
```

BOM 객체

- location 객체



BOM 객체

- history 객체
 - 사용자가 방문했던 URL 리스트 정보를 저장해 두는 객체
 - history 객체의 속성

속성	기능 설명
length	history list에 저장되어 있는 url 주소들의 개수

- history 객체의 메소드

메소드	기능 설명
back()	현재 웹 페이지에서 방문한 이전 페이지로 이동
forward()	현재 웹 페이지에서 방문한 다음 페이지로 이동
go()	상대 숫자를 설정하여 방문한 페이지로 이동

BOM 객체

- history 객체

```
<!DOCTYPE html>                                location-main2.html
<html>
<head>
  <meta charset="utf-8">
  <title>iframe 메인 화면</title>
</head>
<body>
  <iframe src="location2.html" width=" 100%" height="130"></iframe>
  <iframe src="empty.html" width=" 100%" height="900"></iframe>
</body> </html>
```

```
<!DOCTYPE html>                                empty.html
<html>
<head>
  <meta charset="utf-8">
  <title>Empty Frame</title>
</head>
<body>
  <h1>이 웹페이지는 iframe 방식으로 표시됩니다.</h1>
</body> </html>
```

BOM 객체

- location 객체

```
<!DOCTYPE html>                                     location2.html
<html>
<head>
  <meta charset="utf-8">
  <title>History Object</title>
</head>
<body bgcolor="lightblue">
  <script>
    function loadPage(urlAddress) { parent.frames[1].location.href = urlAddress; }
  </script>
  <big>
    <form name="form1" id="form1">
      <input type="button" value="영진" onClick="loadPage('http://www.yjc.ac.kr');">
      <input type="button" value="다음"
        onClick="loadPage('http://www.daum.net');">
      <input type="button" value="네이트"
        onClick="loadPage('http://www.nate.com');">
    </form>
```


BOM 객체

- history 객체

```
<form name="form2" id="form2">
```

```
<input type="button" value="go back" onClick="JavaScript: history.go(-1);">
```

```
<input type="button" value="go forward" onClick="JavaScript: history.go(1);">
```

```
</form>
```

```
</big>
```

```
</body>
```

```
</html>
```

location2.html

