

UI 프로그래밍

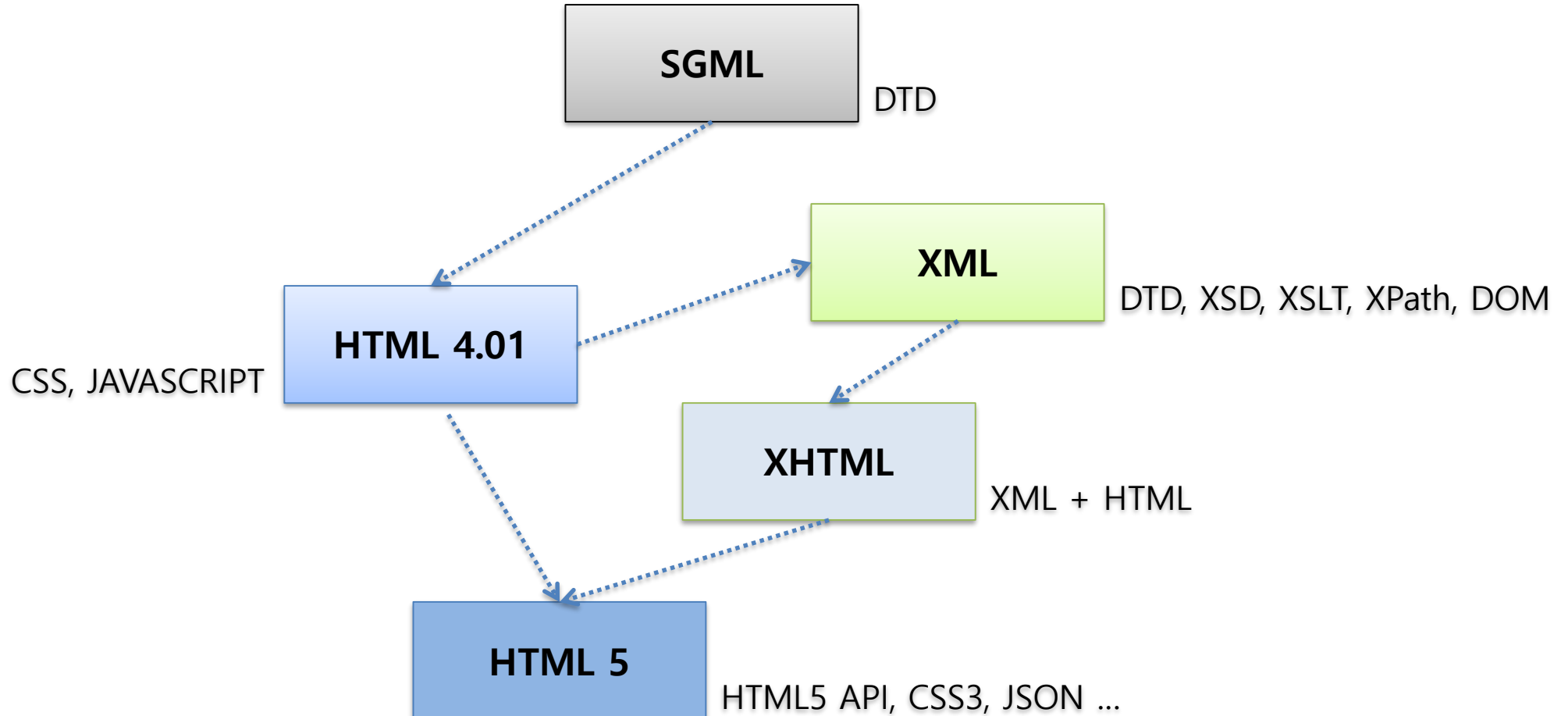
전혜영

OOPSW.jeonhye0@gmail.com

1. HTML5

History

❖ ML(Markup Language)



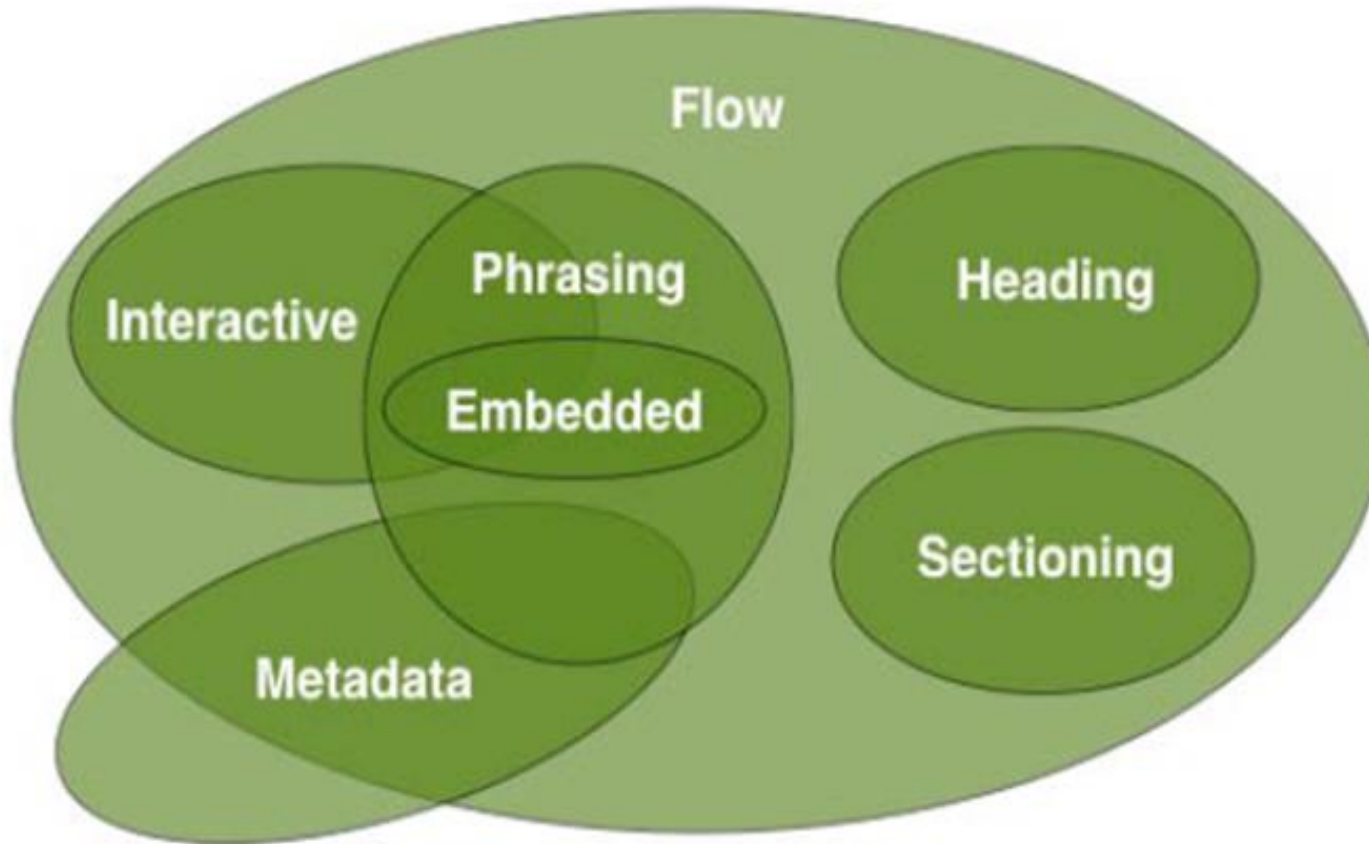
History

❖ HTML5

- 2006.06 : 웹 하이퍼텍스트 워킹그룹(WHATWG) 출범
- Web Form 2.0, Web Applications 1.0
- 2007.03 : 새로운 HTML 워킹 그룹 생성
- 2009.10 : W3C에서 XHTML 전환 실패를 인정함
- 2011.05 : HTML5 최종 초안 (Last Call Working Draft)
- 2012.02 : HTML5 후보 표준안 (Candidate Recommendation) - 2개 이상의 브라우저에서 테스트 완료
- 2014.01 : HTML5 제안 표준안 (Proposed Recommendation) - 브라우저 업체의 피드백 반영
- 2014.2분기 : HTML5 최종 표준안 (Recommendation) 예정

HTML5 개요

❖ HTML5 분류 기준



<Fig. HTML5 분류 기준 출처 : <http://www.whatwg.org> >

HTML5 개요

❖ DOCTYPE

- HTML5

```
<!DOCTYPE html>
```

- HTML 4.01

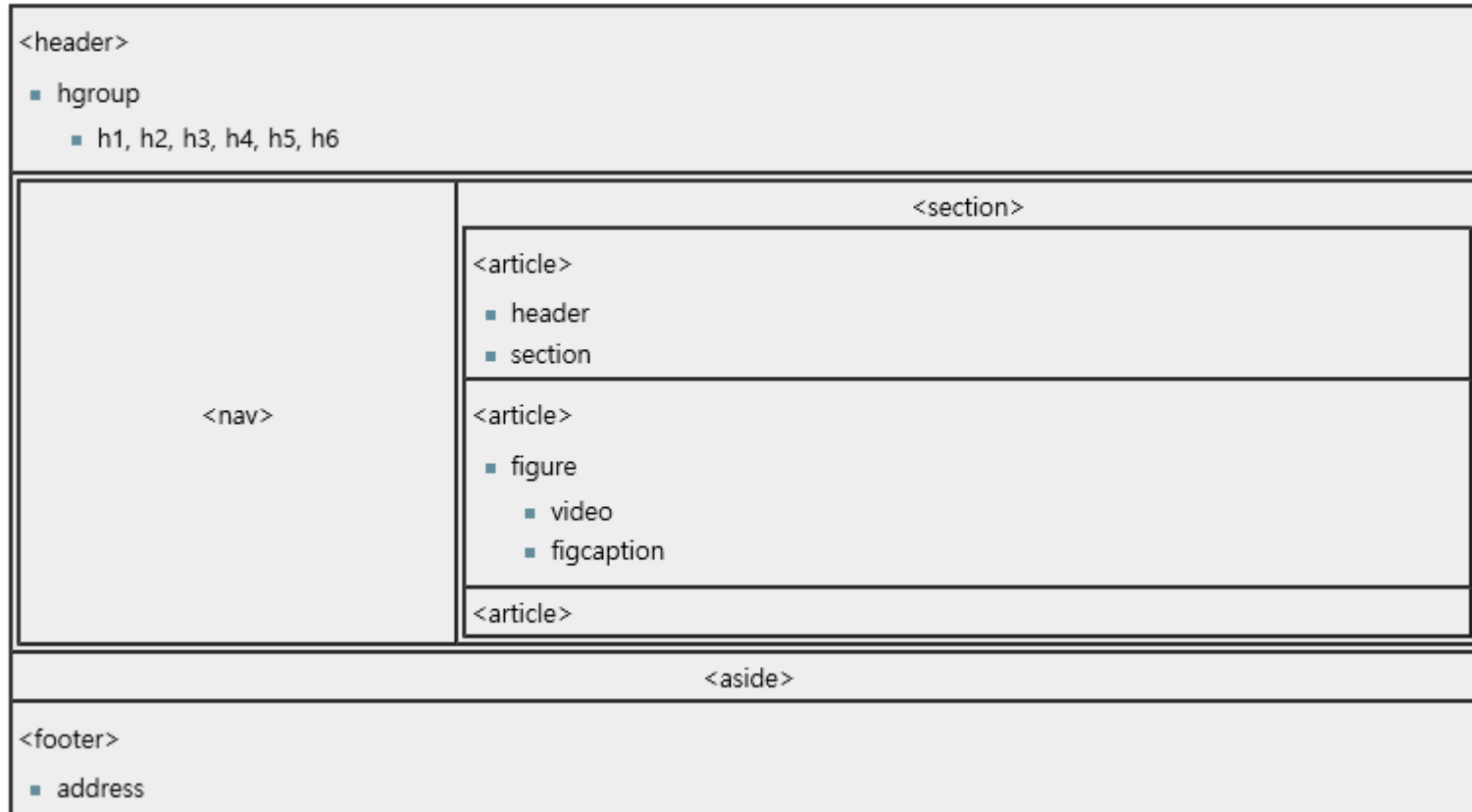
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

- XHTML 1.0

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

HTML5 개요

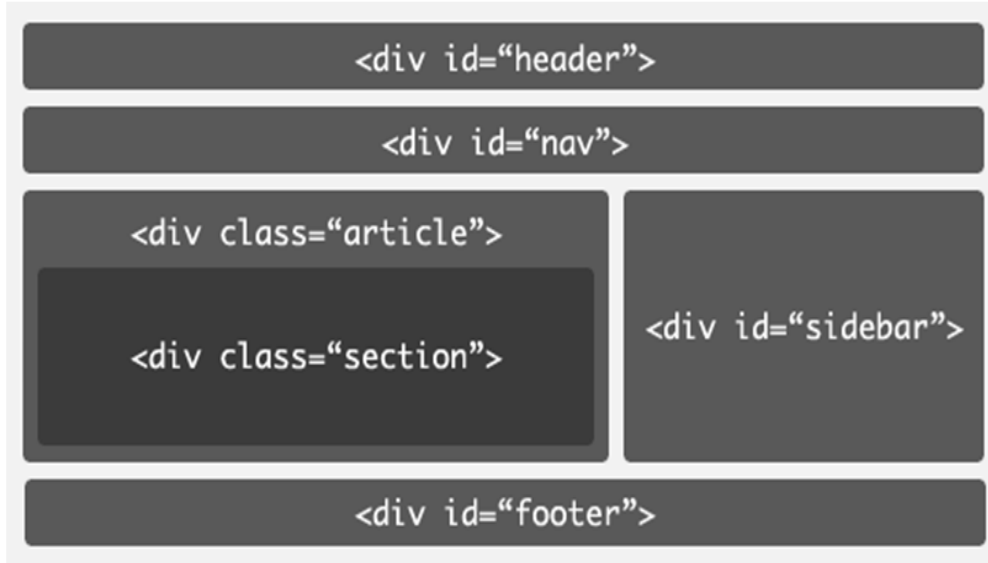
❖ HTML5 Semantic Elements



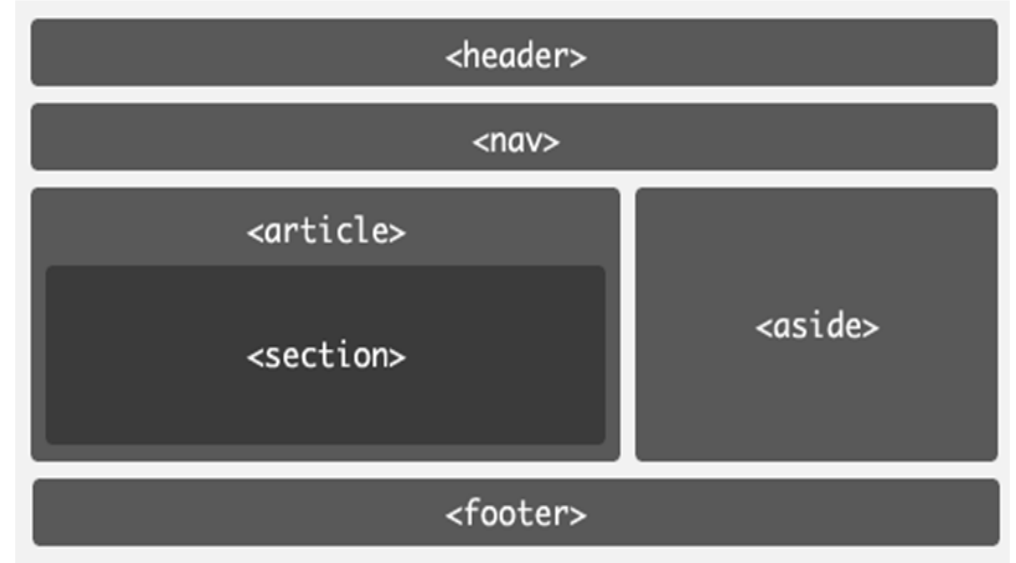
<Fig. HTML5 분류 기준 출처 : <http://www.whatwg.org> >

HTML5 개요

❖ HTML4 VS HTML5



HTML4의 `<div>`를 이용한 화면 디자인



HTML5의 New Element를 이용한 화면 디자인

HTML5 개요

❖ New Elements

▪ Semantic/Structural

- header, hgroup, nav, section, main, article, aside, footer, details, summary, figure, figcaption, mark, time, bdi, wbr, dialog, command, meter, progress, ruby, rt, rp

▪ Form

- datalist, keygen, output

▪ Media

- audio, video, source, track, embed

▪ Graphic

- canvas
- MathML, svg

HTML5 개요

❖ Update Elements

Element	내용
a	href 속성값이 없어도 널(#)링크
address	실제 우편물 주소를 표시
b	텍스트를 진하게하고 제품 소개서안의 제품명, 요약 문서 안의 키워드처럼 중요하지는 않지만 진하게 표시하려고 할때 사용
hr	가로줄 표시에서 단락 단위로 주제 바꾸려고 할 때 사용
i	텍스트 이탤릭체 표시뿐 아니라 중요한 정보가 들어 있지 않은 주변 콘텐츠로 부터 텍스트를 따로 표시할 때 사용
menu	실제 문서 메뉴 정보를 제공하는데 사용
small	세부 주석이나 법적 인쇄 문서에서 작은 인쇄 정보를 표시
strong	단순히 진하게 표시하는 것보다 더 중요한 정보를 표시할 때 사용

HTML5 개요

❖ Global Attributes

- contenteditable
- contextmenu
- data-*
- draggable
- dropzone
- hidden
- spellcheck
- translate

HTML5 개요

❖ New Input Types

- color
- date
- datetime
- datetime-local
- email
- month
- number
- range
- search
- tel
- time
- url
- week

HTML5 개요

❖ Update Attributes

Attribute	대상 Element	내용
media	a, area	링크할 미디어 지정
hreflang	area	링크하는 대상 언어 지정
rel	area	area가 포함된 문서와 링크 대상r간의 관계
target	base	href 속성에서 지정한 링크의 target 지정
value	li	첫번째 항목의 값 지정
charset	meta	문자 인코딩 선언
autofocus	input, select, textarea, button	입력 필드에 포커스
placeholder	input, textarea	입력 필드에 힌트 내용 표시
required	input(hidden, image, submit 제외), textarea	필수 입력 필드 지정
type	menu	메뉴 종류 선택(toolbar, context)
label	menu	메뉴명 지정
scoped	style	해당 요소의 부모 요소와 자식 요소 스타일 적용대상
reversed	ol	목록 역순 표시
start	ol	첫번째 항목 시작값으로 지정

HTML5 개요

❖ New Event Attributes

▪ New Window Events

- onafterprint, onbeforeprint, onbeforeunload,
- onerror, onhaschange, onmessage, onoffline, ononline,
- onpagehide, onpageshow, onpopstate, onredo,
- onresize, onstorage , onundo

▪ New Form Events

- oncontextmenu,
- onformchange,
- onforminput, oninput,
- oninvalid

HTML5 개요

❖ New Event Attributes

▪ New Mouse Events

- ondrag, ondragend, ondragenter, ondragleave, ondragstart,
- ondrop, onmousewheel, onscroll

▪ New Media Events

- oncanplay, oncanplaythrough, ondurationchange, onemptied, onended, onerror
- onloadeddata, onloadedmetadata, onloadstart, onpause, onplay , onplaying
- onprogress, onratechange, onreadystatechange, onseeked
- onseeking, onstalled, onsuspend, ontimeupdate, onvolumechange
- onwaiting

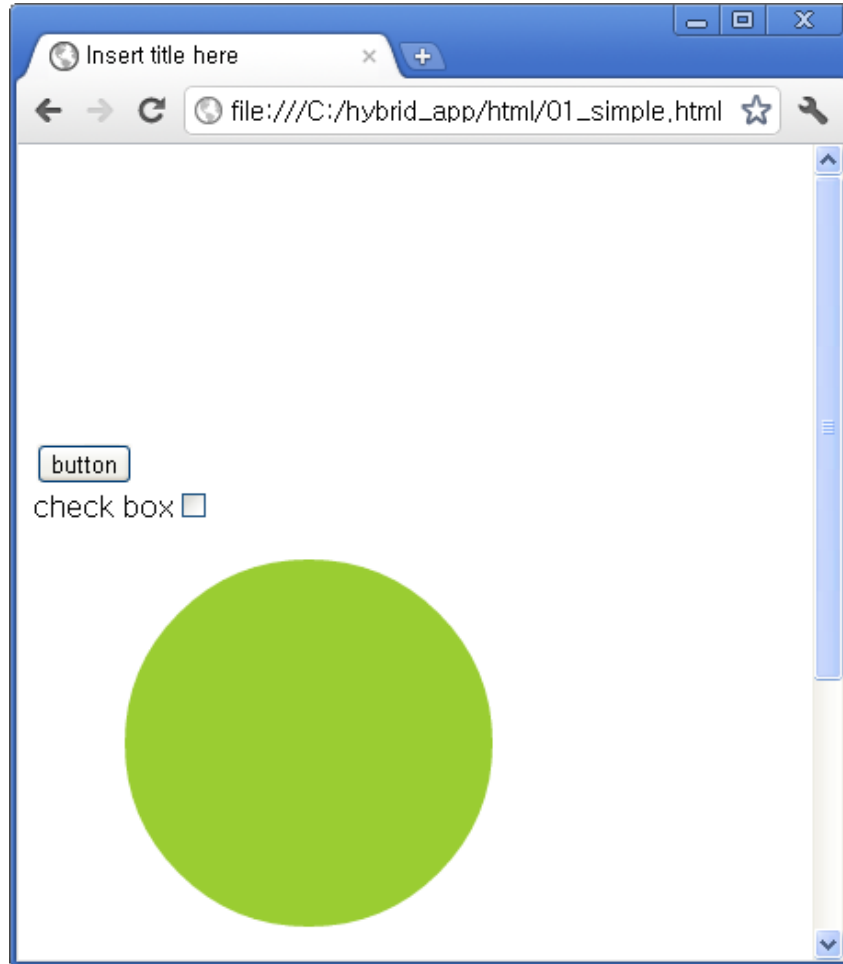
HTML5 개요

❖ HTML5 API

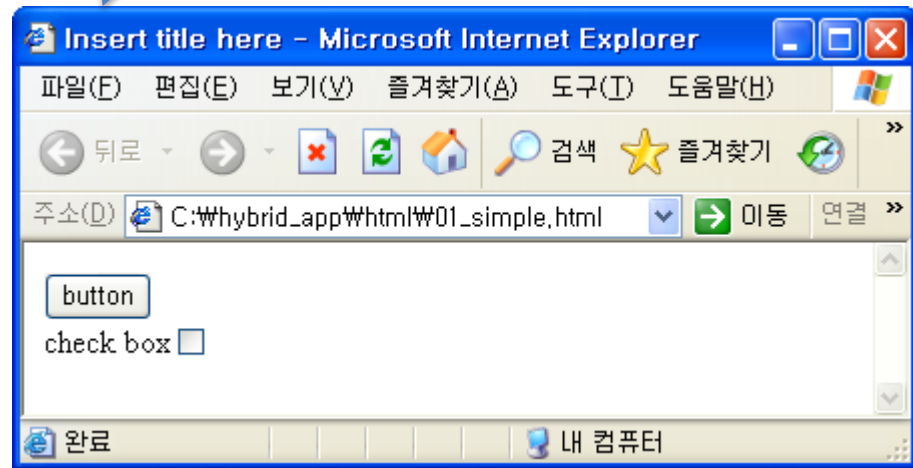
- Geolocation
- Drag/Drop
- Web Storage
- Indexed DB
- Web Workers
- App Cache
- Canvas

HTML5기반 프로그래밍

❖ HTML5 브라우저 테스트



Chrome



다음과 같이 같은 소스라 하더라도
브라우저 종류 및 버전에 따라
다를 수 있다.

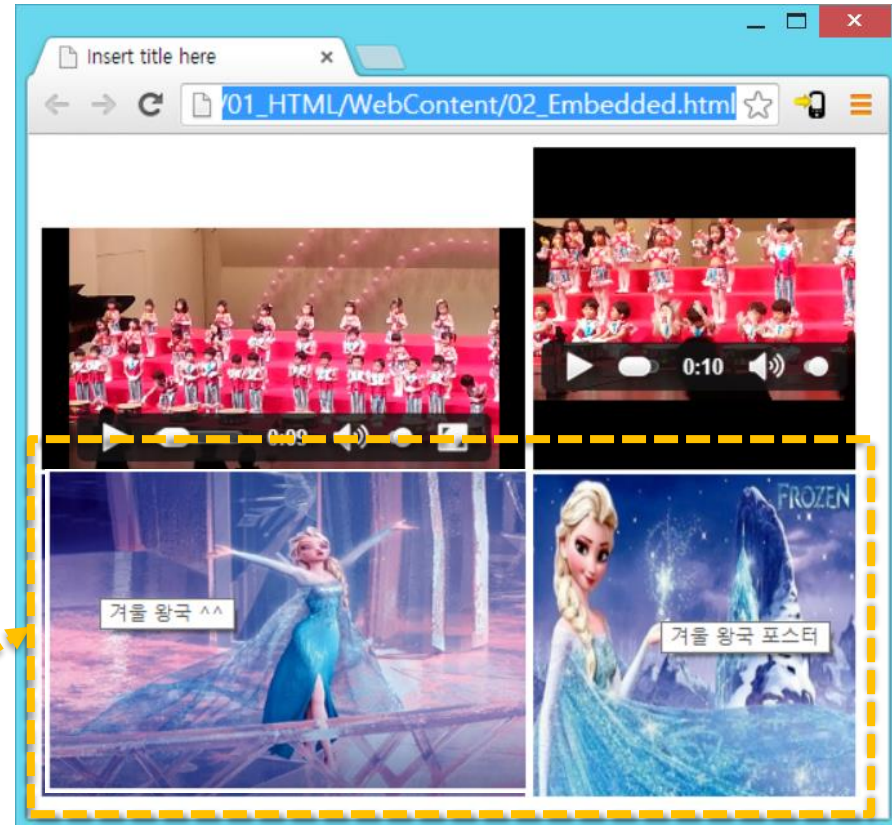
Internet Explorer 8

HTML5기반 프로그래밍

❖ embed

- embed는 mp3, mp4, png.swf 등 다양한 멀티미디어 (외부 애플리케이션)삽입
- 기본 플레이어를 트래커, 컨트롤로, 파일정보, 상황선 등 속성으로 설정 가능
 - type : MIME_Type
 - showtraker : 트래커
 - showaudiocontrols : 볼륨 컨트롤
 - showpositioncontrols : 이전/다음곡
 - showgotobar : 바로가기
 - showdisplay : 파일정보
 - showstatusbar : 상황선
 - volume(-10000~0)
 - balance(-10000 ~ 10000)

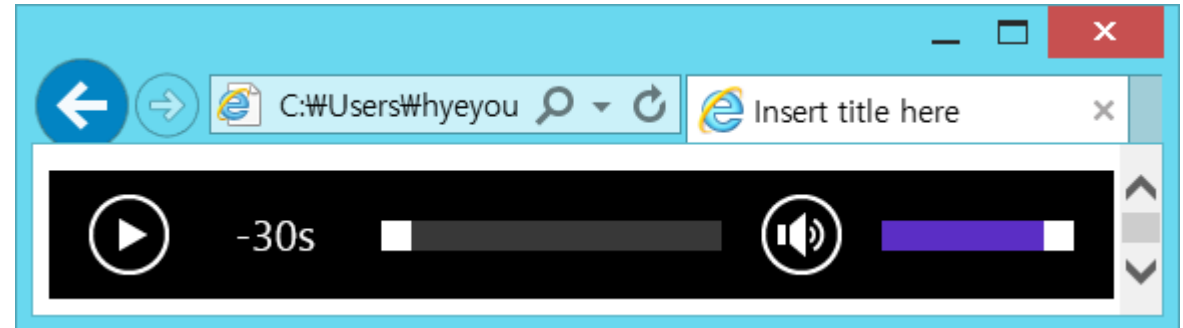
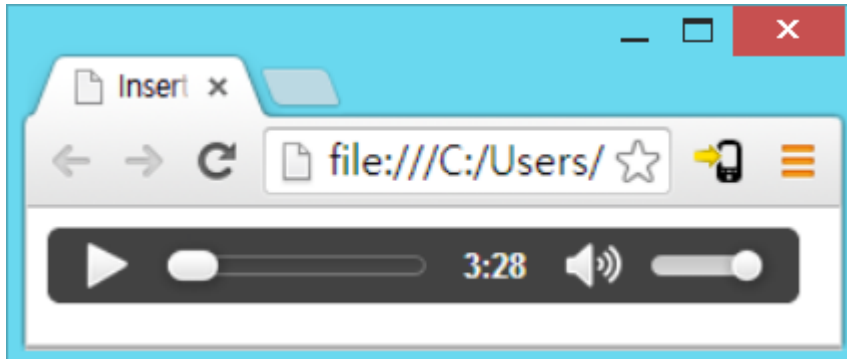
이미지 위에 마우스 선택 시
title 속성값 출력



HTML5기반 프로그래밍

❖ audio

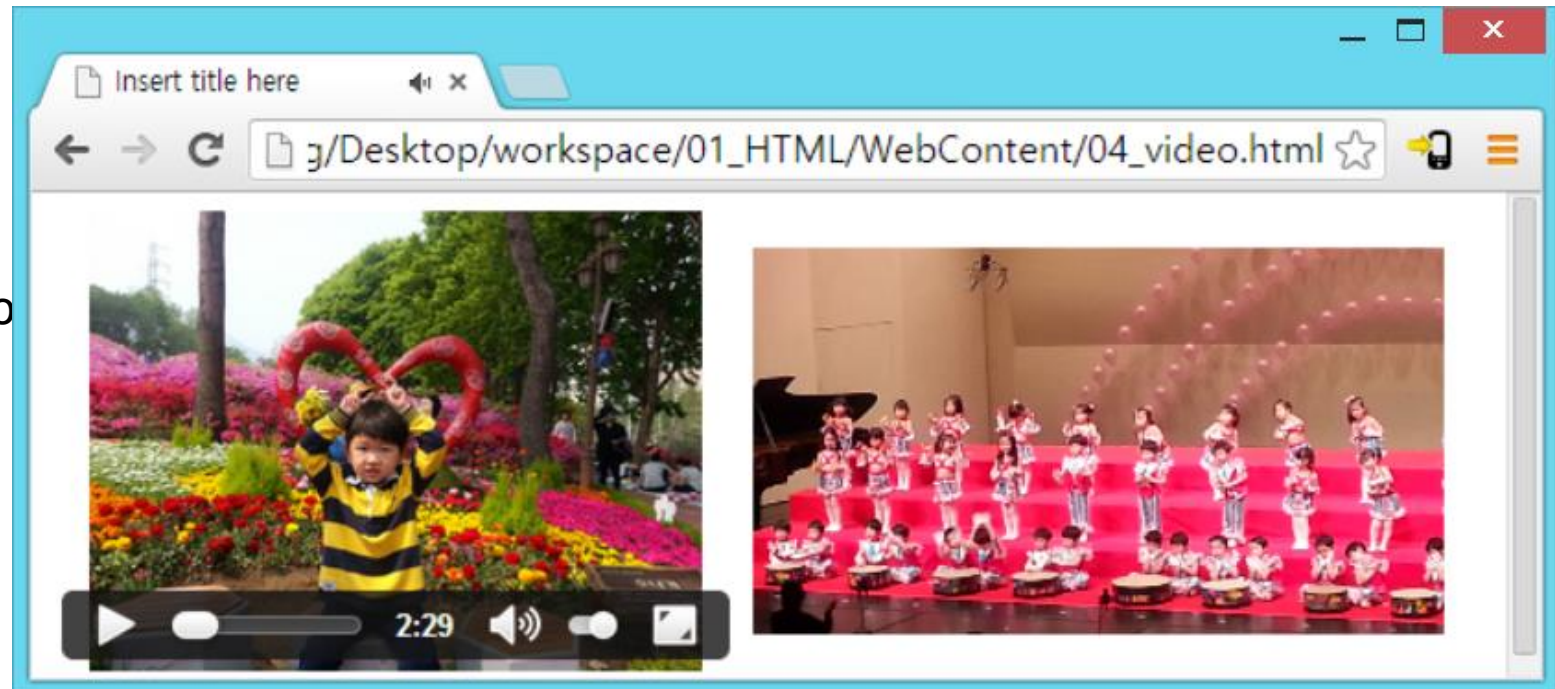
- 소리 재생
- <source>를 이용해서 1개 이상의 소리를 추가할 수 있다.
- audio 속성
 - src
 - autoplay
 - loop
 - controls
 - preload={auto | metadata | none}



HTML5기반 프로그래밍

❖ video

- 멀티미디어 재생
- <source>를 이용해서 1개 이상의 소리를 추가할 수 있다.
- audio 속성
 - src
 - autoplay
 - loop
 - controls
 - preload={auto}
 - poster
 - audio
 - width
 - height



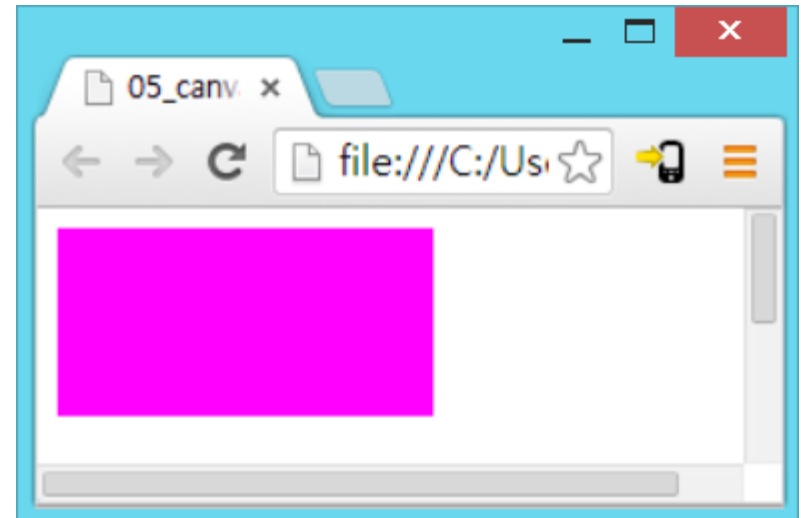
HTML5기반 프로그래밍

❖ canvas

- 동적으로 비트맵 그래픽 그리기
- CSS, Javascript 응용



CSS 응용

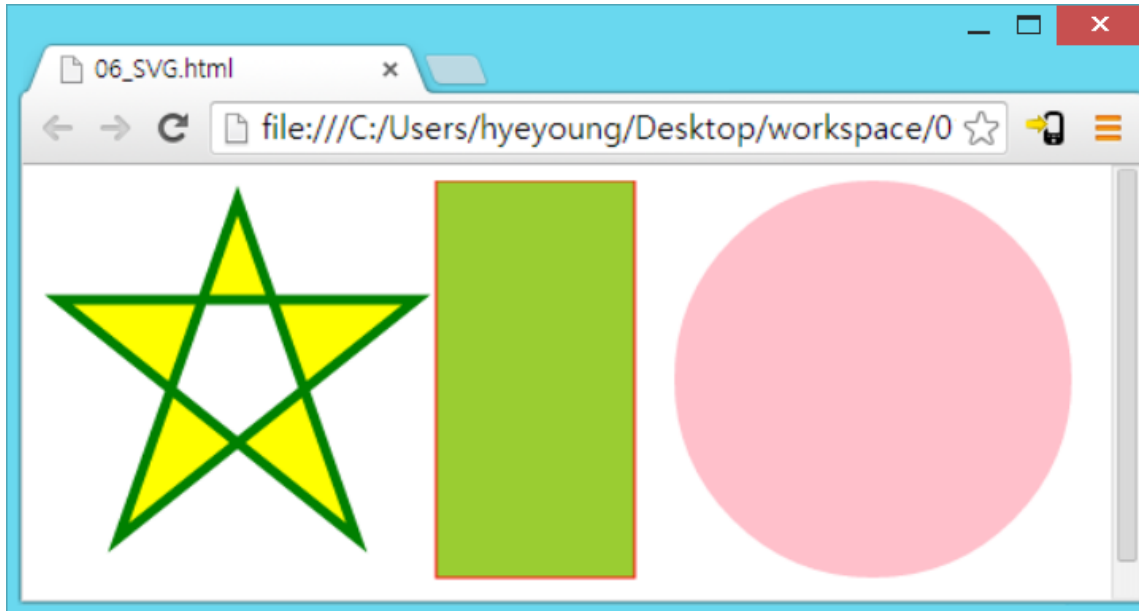


Javascript 응용

HTML5기반 프로그래밍

❖ SVG

- SVG(Scalable Vector Graphics)는 2차원 벡터 그래픽을 표현하기 위한 XML기반의 파일 형식
- SVG 규격은 월드 와이드 웹 컨소시엄 (W3C)에 의해 1999년 개발된 개방형 표준
- SVG 형식의 파일은 SVG기반의 전문 그래픽 편집 프로그램을 사용하여 편집이
- HTML5에서 SVG 엘리먼트 제공



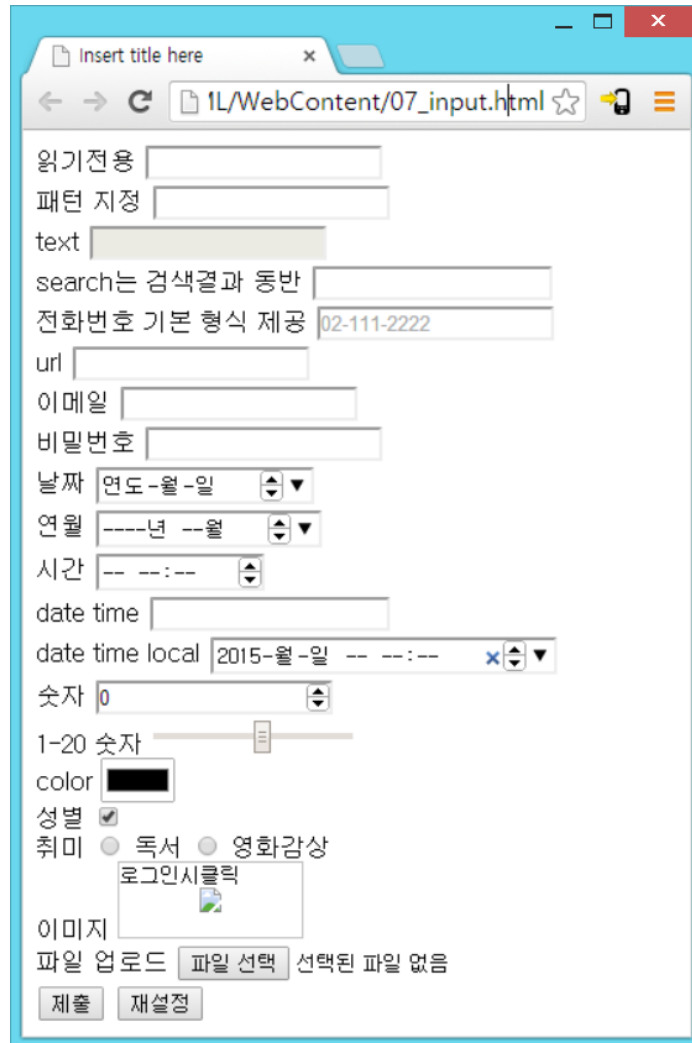
HTML5기반 프로그래밍

❖ input Element Attribute

Element	Attribute	내용
input	type	text, password, email, date, ...
	autofocus	문서가 로드되었을 때 focus를 여기로 옮긴다.
	placeholder	값이 입력되기 전에 힌트 정보로 표시 된다.
	required	필수 입력 요소를 나타낸다.
	disabled	필드셋을 비활성화
	multiple	한개 이상의 속성을 선택 가능
	min	최소값
	max	최대값
	step	증가값
	datalist	가질수 있는 값의 목록
	autocomplete	자동 완성
	pattern	필드값이 가질 수 있는 패턴 지정
	onFormInput	필드의 입력값을 검사 한다.

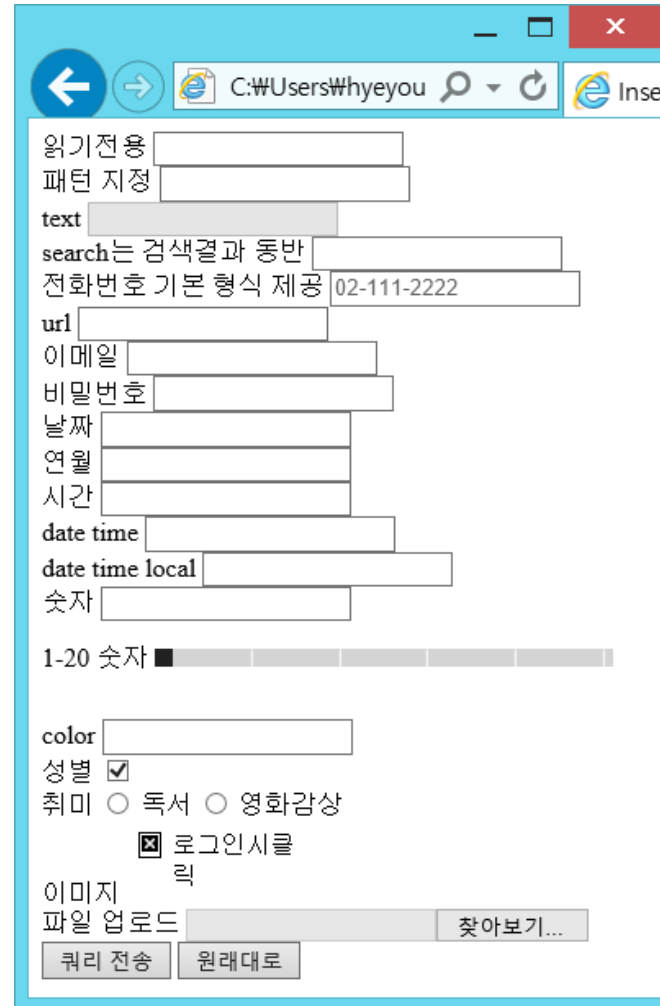
HTML5기반 프로그래밍

❖ input Element Attribute



A screenshot of a web browser window displaying a form titled "Insert title here". The form contains several input fields and controls:

- 읽기전용 (Read-only text input)
- 패턴 지정 (Pattern specification text input)
- text (Text input)
- search는 검색결과 동반 (Search input with placeholder text)
- 전화번호 기본 형식 제공 (Phone number input with value "02-111-2222")
- url (URL input)
- 이메일 (Email input)
- 비밀번호 (Password input)
- 날짜 (Date input with "연도-월-일" format)
- 연월 (Month and year input with "----년 --월" format)
- 시간 (Time input with "-- :--" format)
- date time (Date and time input)
- date time local (Date and time local input with value "2015-월-일 -- :--")
- 숫자 (Number input with value "0")
- 1-20 숫자 (Range input with value "0")
- color (Color input with black swatch)
- 성별 (Gender input with checked checkbox)
- 취미 (Hobby input with radio buttons for "독서" and "영화감상")
- 로그인시클릭 (Login button)
- 이미지 (Image input)
- 파일 업로드 (File upload input with "파일 선택" button and "선택된 파일 없음" text)
- 제출 (Submit button)
- 재설정 (Reset button)



A screenshot of a web browser window displaying a form titled "Insert title here". The form contains several input fields and controls:

- 읽기전용 (Read-only text input)
- 패턴 지정 (Pattern specification text input)
- text (Text input)
- search는 검색결과 동반 (Search input with placeholder text)
- 전화번호 기본 형식 제공 (Phone number input with value "02-111-2222")
- url (URL input)
- 이메일 (Email input)
- 비밀번호 (Password input)
- 날짜 (Date input with "연도-월-일" format)
- 연월 (Month and year input with "----년 --월" format)
- 시간 (Time input with "-- :--" format)
- date time (Date and time input)
- date time local (Date and time local input with value "2015-월-일 -- :--")
- 숫자 (Number input with value "0")
- 1-20 숫자 (Range input with value "0")
- color (Color input with black swatch)
- 성별 (Gender input with checked checkbox)
- 취미 (Hobby input with radio buttons for "독서" and "영화감상")
- 로그인시클릭 (Login button)
- 이미지 (Image input)
- 파일 업로드 (File upload input with "파일 선택" button and "선택된 파일 없음" text)
- 제출 (Submit button)
- 재설정 (Reset button)

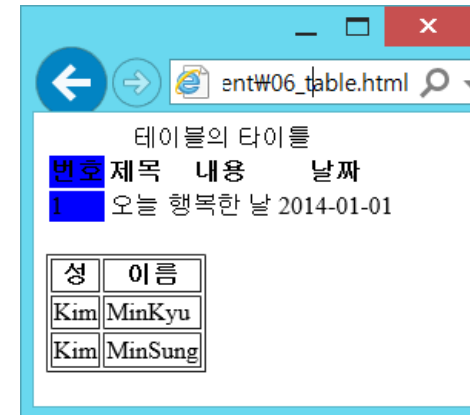
HTML5기반 프로그래밍

❖ table

- thead, tfoot, tbody 엘리먼트의 순서가 바뀌더라도 출력순서는 head, body, foot를 유지
- table 등과 같이 뷰 속성을 포함하는 엘리먼트는 다음과 같이 브라우저에 따라 다르게 디스플레이 되기 때문에 선호하지 않는 것이 바람직하다.
- 대신 <div>와 CSS를 응용하여 사용된다.



CSS 응용



Javascript 응용

2. HTML5

2.1 History

2.2 HTML5 개요

2.3 HTML5기반 프로그래밍

2.4 UI 디자인 사례

2.5 프로젝트마크

❖ html table을 이용한 UI Layout

-
- The screenshot shows a web browser window with the address bar displaying `https://www.gabia.com`. The page header includes the GABIA logo and navigation links: `도메인`, `호스팅`, `IDC`, `기업솔루션`, `웹사이트`, `쇼핑몰`, `서비스전체보기 +`, `My가비아`, and `고객센터`. The browser's developer tools are open, showing the `Elements` panel with the following HTML structure:
- ```

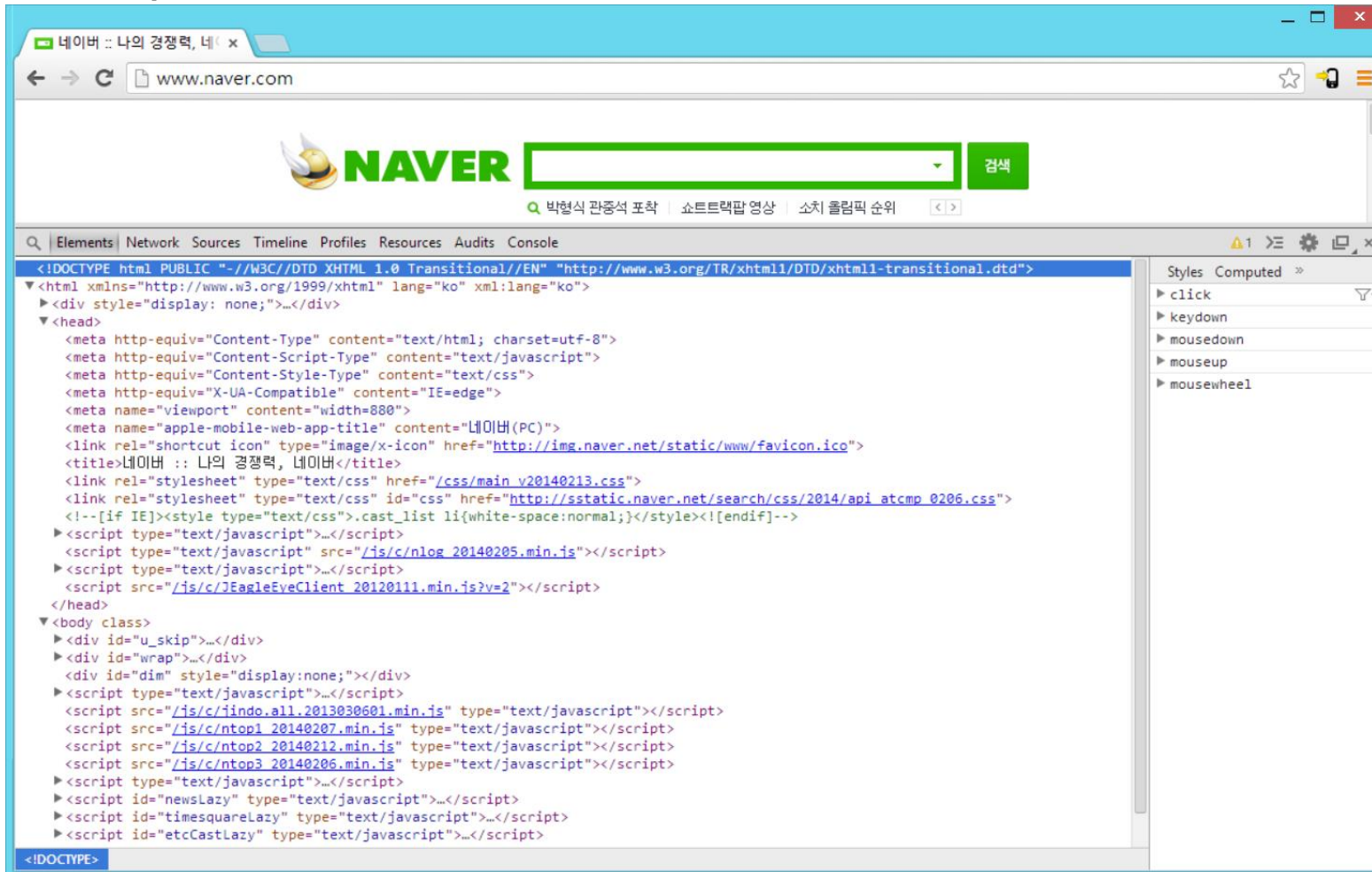
<script type="text/javascript">...</script>
<!--div id="DivPop" style="display:none; position:absolute; z-index:20; margin:0 auto;">
 <table width="870" border="0" cellspacing="0" cellpadding="0">
 <tr>
 <td></td>
 </tr>
 <tr>
 <td align="center" background="/assets/images/inc/2011/cs_img02.png" class="png24">
 <table width="812" border="0" cellspacing="0" cellpadding="0">
 <tr>
 <td></td>
 <td width="22"> </td>
 <td></td>
 </tr>
 </table>
 </td>
 </tr>
 <tr>
 <td></td>
 </tr>
 </table>
<!-- 스크롤메뉴
</div-->
<div id="DivPop" class="csLayer" style="display: none; position: absolute; z-index: 20; margin: 0px auto; top: 80px; left:
0px;">...</div>

```
- The `Styles` panel on the right shows the computed styles for the selected element, including `body`, `HTMLBodyElement`, `HTMLElement`, `Element`, `Node`, `EventTarget`, and `Object`.

# UI 디자인 사례

## ❖ XHTML 기반

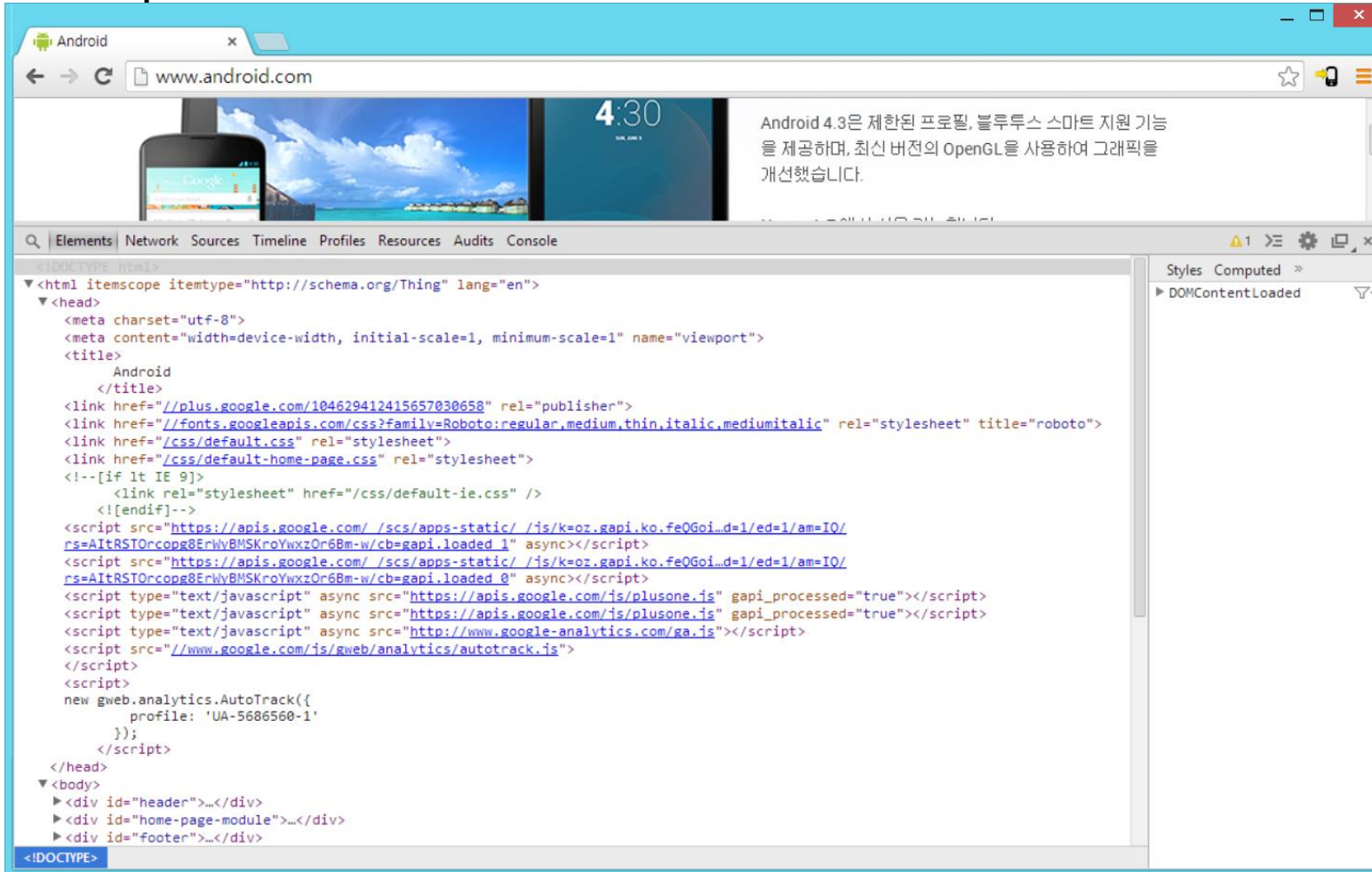
- <http://www.naver.com/>



# UI 디자인 사례

## ❖ HTML5 기반

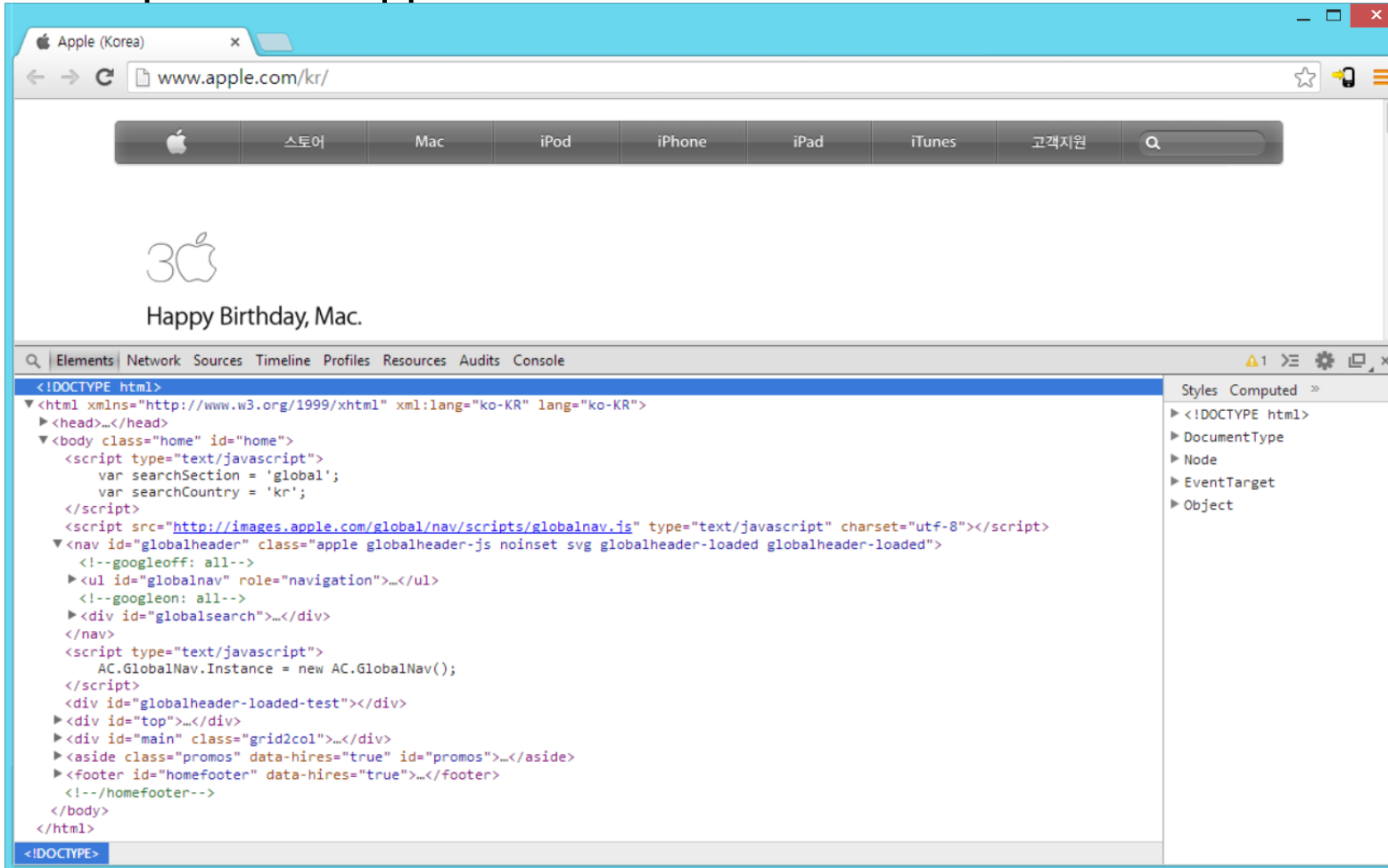
- <http://www.android.com/>



# UI 디자인 사례

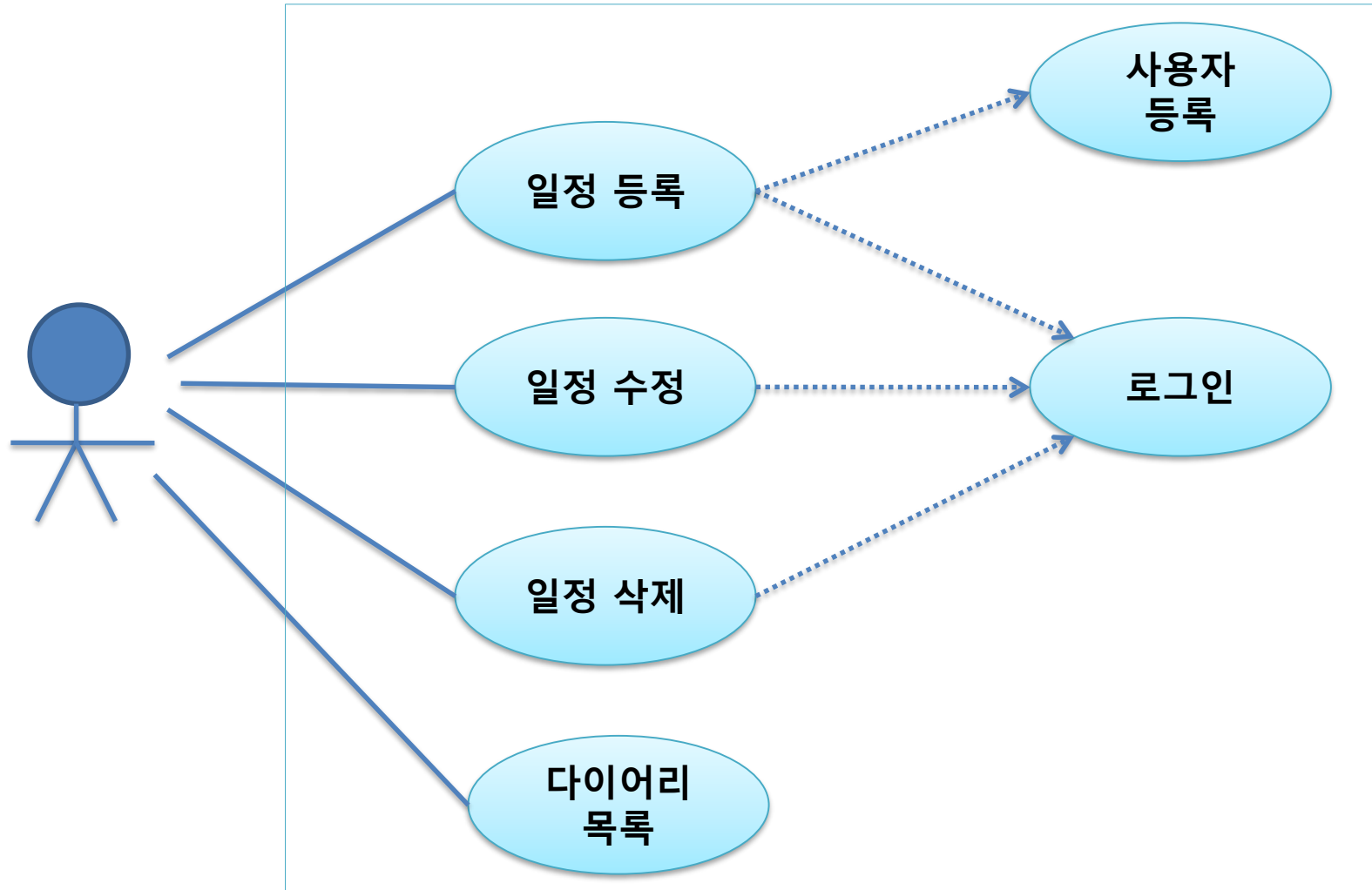
## ❖ XHTML + HTML5 기반

- <http://www.apple.com/kr/>



# 프로젝트

## ❖ 다이어리 프로젝트 요구사항



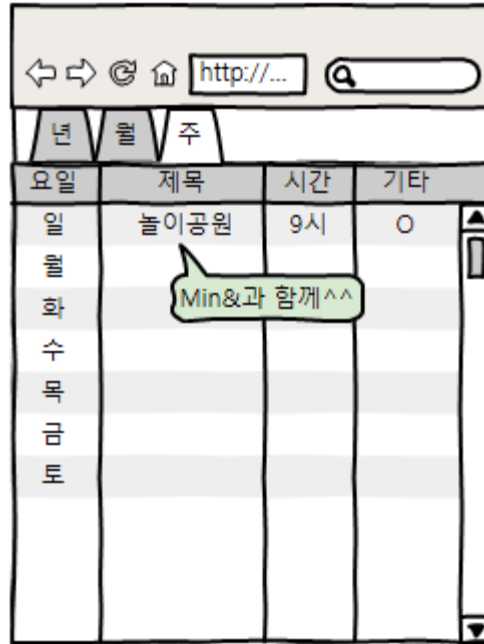
# 프로젝트

## ❖ UI 명세서



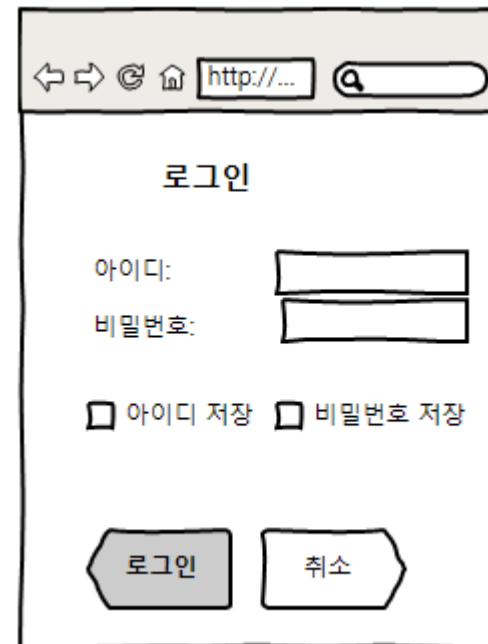
A browser window mockup for a user registration page. The title bar shows navigation icons and a URL field with 'http://...'. The page title is '사용자 등록' (User Registration). The form includes fields for '아이디:' (ID), '비밀번호:' (Password), '비밀번호 확인:' (Confirm Password), '이름:' (Name), and '전화번호:' (Phone Number). The '아이디:' field has a yellow highlight and a tooltip that says '중복체크' (Check for duplicates). The '전화번호:' field has a pre-filled value '010-0000-1111'. At the bottom are two buttons: '사용자등록' (User Registration) and '취소' (Cancel).

1) 사용자 등록



A browser window mockup for a calendar application. The title bar shows navigation icons and a URL field with 'http://...'. The page title is '일정목록' (Calendar). The interface includes a header with '년' (Year), '월' (Month), and '주' (Week) tabs. Below is a table with columns: '요일' (Day of the Week), '제목' (Subject), '시간' (Time), and '기타' (Other). The table has rows for '일' (Sun), '월' (Mon), '화' (Tue), '수' (Wed), '목' (Thu), '금' (Fri), and '토' (Sat). The '일' row has '놀이공원' (Amusement Park) in the '제목' column, '9시' (9 AM) in the '시간' column, and 'O' in the '기타' column. A tooltip with the text 'Min&과 함께^^' (With Min& ^^) is shown over the '일' row. A vertical scrollbar is on the right side of the table.

2) 일정목록



A browser window mockup for a login page. The title bar shows navigation icons and a URL field with 'http://...'. The page title is '로그인' (Login). The form includes fields for '아이디:' (ID) and '비밀번호:' (Password). Below these fields are two checkboxes: '아이디 저장' (Save ID) and '비밀번호 저장' (Save Password). At the bottom are two buttons: '로그인' (Login) and '취소' (Cancel).

3) 로그인



## 2. CSS3

# CSS 개요

## ❖ CSS(Cascading Style Sheets)

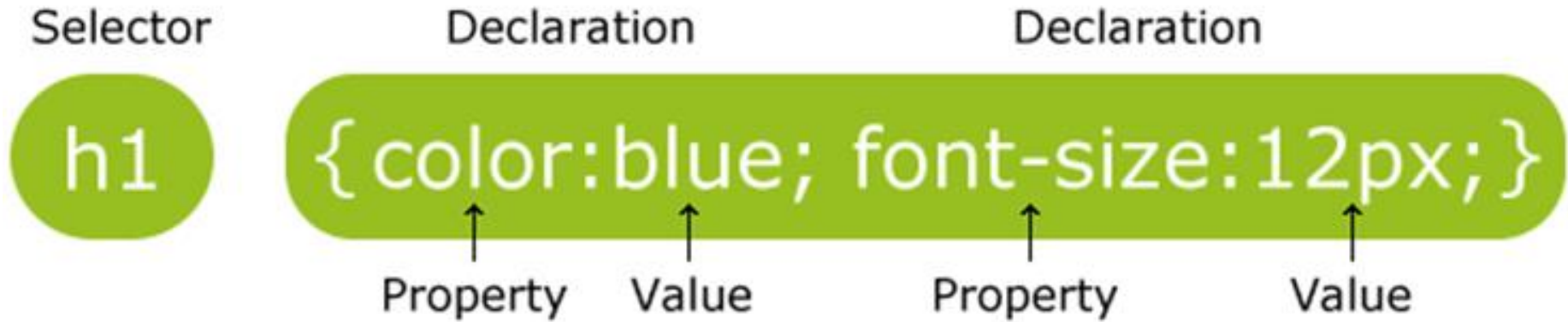
- CSS는 콘텐츠를 정의하는 ML의 디자인 속성을 제공한다.

## ❖ CSS History

- CSS1 1996년 발표
- CSS2 1998년 발표
- 현재는 CSS3 버전으로 다양한 속성을 제공

# CSS 개요

## ❖ CSS 구조



# CSS 개요

## ❖ CSS 사용법

- 엘리먼트 style 속성

```
<p style="color : yellow">
```

- <head> 내장 CSS

```
<head>
 <style type="text/css">p{ color:yellow;}</style>
</head>
```

- 외부 CSS 링크

```
<head>
 <link rel="stylesheet" href="pStyle.css" type="text/css" />
</head>
```

# CSS 개요

## ❖ id 속성 사용법

- #으로 스타일 속성 정의
  - `H1#test {letter-spacing: 5em }`
- id로 스타일 속성 적용
  - `<H1 id="test">Wide text</H1>`

# CSS 개요

## ❖ class 속성 사용법

- .으로 스타일 속성 정의
  - . test {color : blue}
  - H1.red {color : red }
- class로 스타일 속성 적용
  - <STRONG class = test>파란색</STRONG>
  - <EM class = pastoral>마찬가지</EM>
  - <H1 class = red>빨강색</H1>
  - <H1>검은색</H1>

# CSS 개요

## ❖ Contextual selectors

- H1 엘리먼트의 EM 엘리먼트에만 속성 정의
  - H1 EM {color : blue }
- 속성 적용
  - <H1>검은색</H1>
  - <EM>검은색</EM>
  - <H1>검은색 <EM>푸른색</EM> 다시 검은색</H1>

# CSS 개요

## ❖ 단위

Unit	Description
%	percentage
in	inch
cm	centimeter
mm	millimeter
em	1 font size
ex	1/2 font size
pt	point, 1/72 inch
pc	pica, 12pt
px	pixels



# 자주 사용되는 CSS 속성CS

## ❖ text

- color
- direction
- letter-spacing
- line-height
- text-align
- text-decoration
- text-indent
- text-shadow
- text-transform
- unicode-bidi
- vertical-align
- white-space
- word-spacing

# 자주 사용되는 CSS 속성CS

## ❖ font

- font
- font-family
- font-size
- font-style xt
- font-variant
- font-weight



Sans-serif



Serif



Serif  
(red serifs)

# 자주 사용되는 CSS 속성CS

## ❖ link

- a:link
- a:visited
- a:hover
- a:active

# 자주 사용되는 CSS 속성CS

## ❖ list

- list-style
- list-style-image
- list-style-position
- list-style-type

# 자주 사용되는 CSS 속성CS

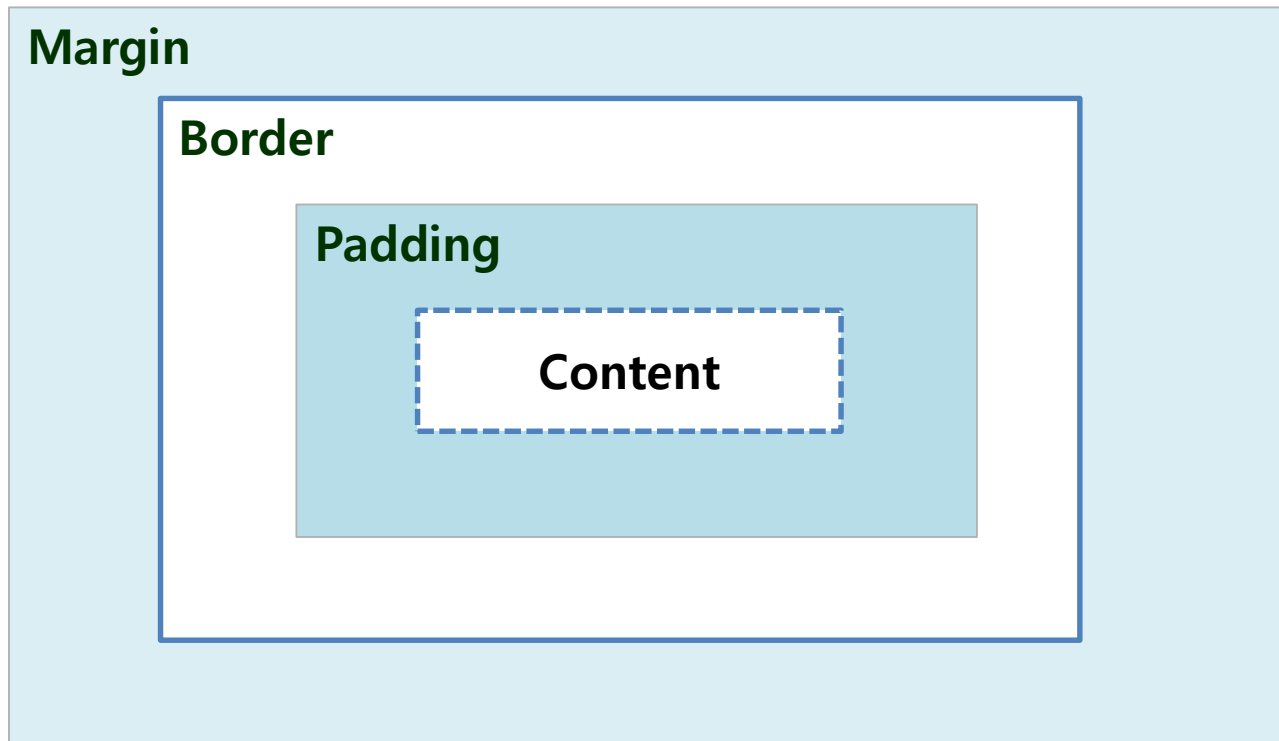
## ❖ background

- background-color
  - `h1 {background-color:#6495ed;}`
  - `p {background-color:#e0ffff;}`
  - `div {background-color:#b0c4de;}`
- background-image
  - `body {background-image:url('paper.gif');}`
- background-repeat
  - background-attachment
- background-position

# 자주 사용되는 CSS 속성CS

## ❖ BOX

- margin
- border
- padding



# 자주 사용되는 CSS 속성

## ❖ border

### ▪ border-style

dotted: Defines a dotted border

dashed: Defines a dashed border

solid: Defines a solid border

double: Defines two borders. The width of the two borders are the same as the border-width value

groove: Defines a 3D grooved border. The effect depends on the border-color value

ridge: Defines a 3D ridged border. The effect depends on the border-color value

inset: Defines a 3D inset border. The effect depends on the border-color value

outset: Defines a 3D outset border. The effect depends on the border-color value

# 자주 사용되는 CSS 속성

## ❖ padding

- padding
- padding-top
- padding-bottom
- padding-right
- padding-left



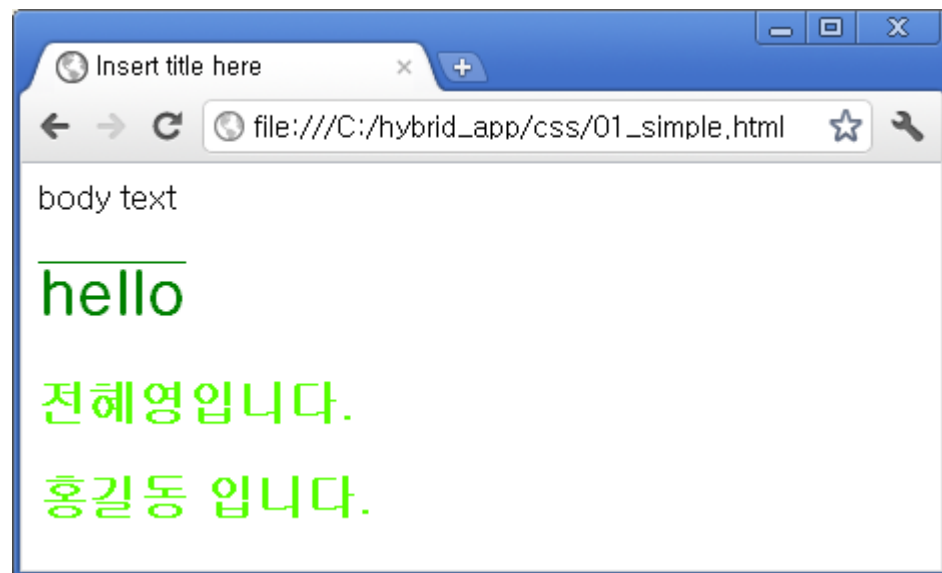
# 자주 사용되는 CSS 속성

## ❖ margin

- margin
- margin-top
- margin-bottom
- margin-right
- margin-left

# CSS 응용 프로그래밍

## ❖ 엘리먼트에 CSS 속성 적용



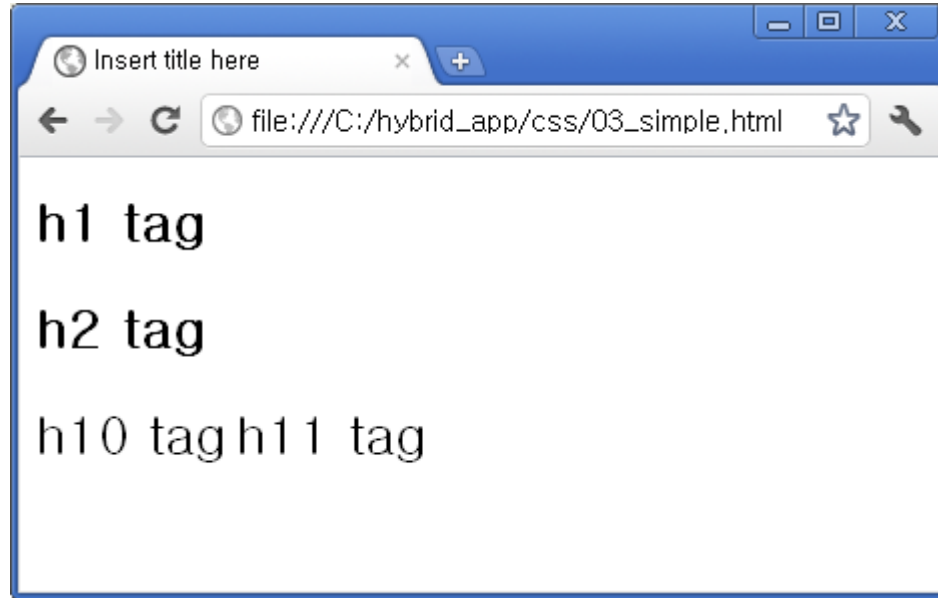
# CSS 응용 프로그래밍

❖ 특정 엘리먼트에 ., #으로 CSS 속성 적용



# CSS 응용 프로그래밍

❖ 여러 엘리먼트 한꺼번에 CSS 속성 적용



# CSS 응용 프로그래밍

❖ 다른 엘리먼트에 같은 CSS 속성 적용



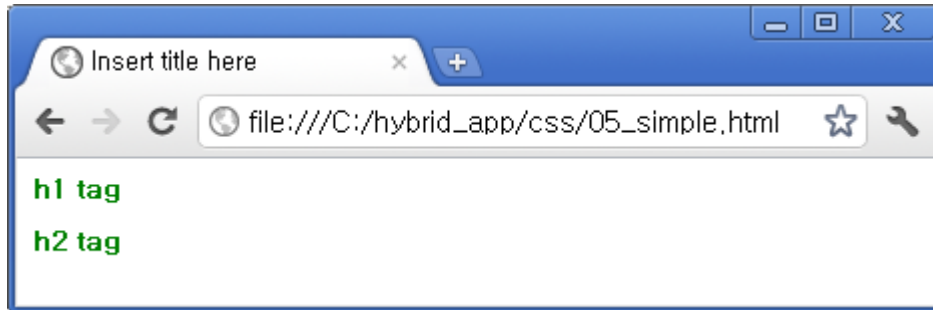
# CSS 응용 프로그래밍

## ❖ 외부 CSS 속성 파일을 가져와 적용

- `<link rel="stylesheet" href="05_simple.css" type="text/css">`

Or

- `<style type="text/css">`  
    `@import "05_simple.css";`  
    `</style>`



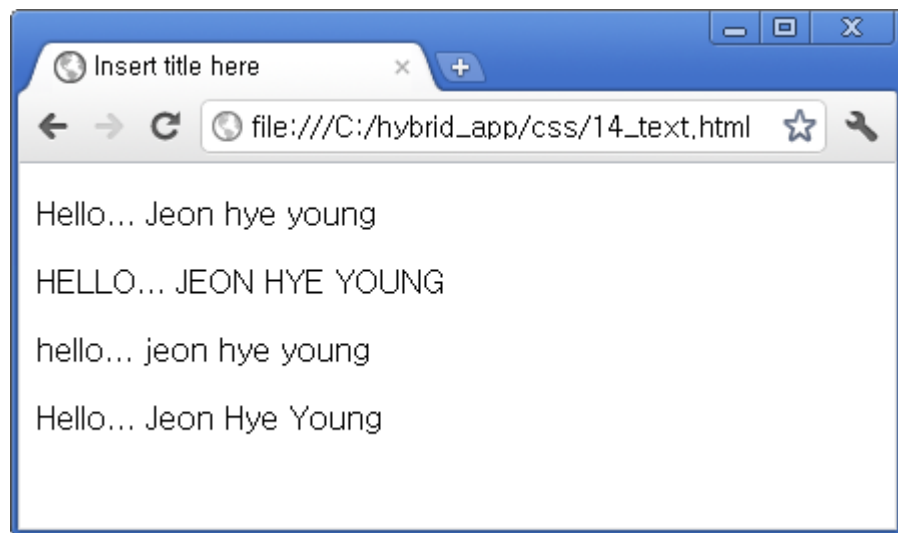
# CSS 응용 프로그래밍

## ❖ 리스트와 링크에 CSS 속성 적용



# CSS 응용 프로그래밍

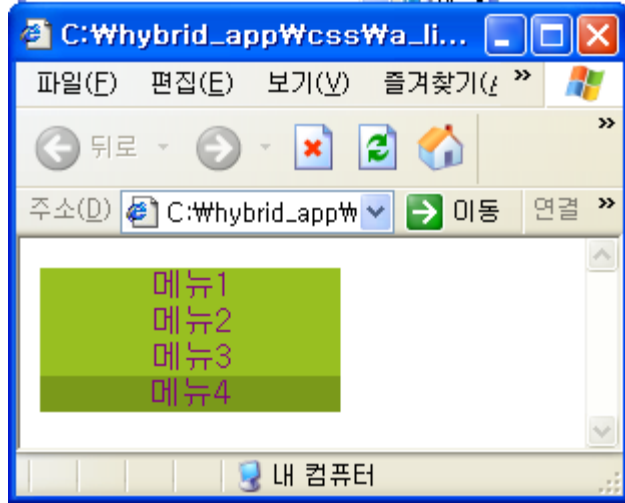
## ❖ 문자열 변환 CSS 속성 적용





# CSS 응용 프로그래밍

## ❖ 메뉴 및 링크 CSS 속성 적용



# CSS3 응용 프로그래밍

## ❖ CSS3 Modules

- Selectors
- Box Model
- Backgrounds and Borders
- Text Effects
- 2D/3D Transformations
- Animations
- Multiple Column Layout
- User Interface

# CSS3 응용 프로그래밍

## ❖ Selectors

Selectors		Description
el1 ~ el2	div ~ ul	<div>뒤에 나오는 모든 <ul> 엘리먼트
[attr ^=value]	a[src ^= 'http']	src 속성값이 http로 시작 모든<a> 엘리먼트
[attr \$=value]	a[src ^= '.ppt']	src 속성값이 ppt로 끝나는 모든<a> 엘리먼트
[attr *=value]	a[src ^= 'sam']	src 속성값이 sam을 포함한 모든 <a> 엘리먼트
:first-of-type	div:first-of-type	<div>로 시작되는 첫번째 엘리먼트
:last-of-type	div:last-of-type	<div>로 시작되는 마지막 엘리먼트
:only-of-type	div:only-of-type	자식이 <div>로만 구성된 <div> 엘리먼트
:nth-child(n)	div:nth-child(2)	부모에서 2번째 <div> 자식 엘리먼트
:nth-last-child(n)	div:nth-last-child(3)	부모에서 뒤에서 3번째 <div> 자식 엘리먼트
:nth-of-type(n)	div:nth-of-type(3)	부모로 부터 2번째 <div> 엘리먼트

# CSS3 응용 프로그래밍

## ❖ Selectors

Selectors		Description
:nth-of-last-type(n)	div:nth-of-last-type(2)	부모로부터 뒤에서 2번째 <div> 엘리먼트
:last-child	div:last-child	부모로부터 마지막 <div> 엘리먼트
:root	:root	document root 엘리먼트
:empty	div:empty	자식 요소를 갖지 않는 <div> 엘리먼트
:target	#news:target	#news 엘리먼트의 url 대상
:enabled	input:enabled	Enabled된 상태의 모든 <input> 엘리먼트
:disabled	input:disabled	disabled된 상태의 모든 <input> 엘리먼트
:checked	input:checked	checked된 상태의 모든 <input> 엘리먼트
:not	:not(div)	<div>가 아닌 모든 엘리먼트
::selection	::selection	사용자가 선택한 영역의 있는 모든 요소

# CSS3 응용 프로그래밍

## ❖ User Interface

- Resize
  - 사용자 크기 조절 핸들 제공
  - `resize : {none : both : horizontal : vertical}`
- Box Sizing
  - Box Model Sizing 방법 변경
  - `box-sizing : {content-box | border-box | inherit}`
- Out Line
  - 외곽선
  - `outline : width style color`
  - `outline-offset : { length : inherit}`

# CSS3 응용 프로그래밍

## ❖ Media Query

- CSS2.1 media type 개선

- <link>

```
<link rel="stylesheet" media="screen and (max-width:400px)" ref="example.css">
```

```
<style type="text/css">
```

```
@import url(example.css) screen and (max-width:400px);
```

```
</style>
```

- <style>

```
<style type="text/css" media="screen and (max-width:400px)" >
```

```
div { back-ground:green}
```

```
</style>
```

- @media style 속성

```
<style type="text/css" >
```

```
@media all and (max-width:400px) { div { back-ground:green}
```

```
</style>
```

# CSS3 응용 프로그래밍

## ❖ Media Query

```
media_query : [ONLY | NOT]? S* media_type S* [AND S* expression]*
| expression [AND S* expression]* ;
```

- media\_type
  - all - 모든 미디어 타입
  - aural - 음성 합성 장치
  - braille - 점자표시장치
  - handheld - 휴대용 장치
  - print - 인쇄장치
  - projection - 프로젝터
  - screen - 컴퓨터 모니터
  - tty - 디스플레이 능력이 한정된 장치
  - tv - TV와 같이 audio/vidio 동시 출력 장치
  - embossed - 페이지에 인쇄된 점자 표시 장치

# 프로젝트

## ❖ CSS 응용 프로그램 디자인 완성



The image displays three wireframe designs for a web application, each within a browser window frame.

**Left Wireframe: 사용자 등록 (User Registration)**

- 아이디: (아이디) 중복체크
- 비밀번호: (비밀번호)
- 비밀번호 확인: (비밀번호 확인)
- 이름: (이름)
- 전화번호: (전화번호) 010-0000-1111
- Buttons: 사용자등록, 취소

**Middle Wireframe: Calendar**

년	월	주	
요일	제목	시간	기타
일	놀이공원	9시	0
월	Min&과 함께^^		
화			
수			
목			
금			
토			

**Right Wireframe: 로그인 (Login)**

- 아이디: (아이디)
- 비밀번호: (비밀번호)
- Checkboxes: ☐ 아이디 저장 ☐ 비밀번호 저장
- Buttons: 로그인, 취소



# Reference

## ❖ CSS Properties Reference

- Animation
- Background
- Border and outline
- Box
- Color
- Content Paged Media
- Dimension
- Flexible Box
- Font
- Generated content
- Grid
- Hyperlink
- Linebox
- List
- Margin
- Marquee
- Multi-column
- Padding
- Paged Media
- Positioning
- Print
- Ruby
- Speech
- Table
- Text
- 2D/3D Transform
- Transition
- User-interface

# Reference

## ❖ Font Properties

Property	Description	CSS
<a href="#"><u>font</u></a>	Sets all the font properties in one declaration	1
<a href="#"><u>font-family</u></a>	Specifies the font family for text	1
<a href="#"><u>font-size</u></a>	Specifies the font size of text	1
<a href="#"><u>font-style</u></a>	Specifies the font style for text	1
<a href="#"><u>font-variant</u></a>	Specifies whether or not a text should be displayed in a small-caps font	1
<a href="#"><u>font-weight</u></a>	Specifies the weight of a font	1
<a href="#"><u>@font-face</u></a>	A rule that allows websites to download and use fonts other than the "web-safe" fonts	3
<a href="#"><u>font-size-adjust</u></a>	Preserves the readability of text when font fallback occurs	3
<a href="#"><u>font-stretch</u></a>	Selects a normal, condensed, or expanded face from a font family	3

# Reference

## ❖ Animation Properties

Property	Description	CSS
<a href="#"><u>@keyframes</u></a>	Specifies the animation	3
<a href="#"><u>animation</u></a>	A shorthand property for all the animation properties below, except the animation-play-state property	3
<a href="#"><u>animation-name</u></a>	Specifies a name for the @keyframes animation	3
<a href="#"><u>animation-duration</u></a>	Specifies how many seconds or milliseconds an animation takes to complete one cycle	3
<a href="#"><u>animation-timing-function</u></a>	Specifies the speed curve of the animation	3
<a href="#"><u>animation-delay</u></a>	Specifies when the animation will start	3
<a href="#"><u>animation-iteration-count</u></a>	Specifies the number of times an animation should be played	3
<a href="#"><u>animation-direction</u></a>	Specifies whether or not the animation should play in reverse on alternate cycles	3
<a href="#"><u>animation-play-state</u></a>	Specifies whether the animation is running or paused	3

# Reference

## ❖ Background Properties

Property	Description	CSS
<a href="#"><u>background</u></a>	Sets all the background properties in one declaration	1
<a href="#"><u>background-attachment</u></a>	Sets whether a background image is fixed or scrolls with the rest of the page	1
<a href="#"><u>background-color</u></a>	Sets the background color of an element	1
<a href="#"><u>background-image</u></a>	Sets the background image for an element	1
<a href="#"><u>background-position</u></a>	Sets the starting position of a background image	1
<a href="#"><u>background-repeat</u></a>	Sets how a background image will be repeated	1
<a href="#"><u>background-clip</u></a>	Specifies the painting area of the background	3
<a href="#"><u>background-origin</u></a>	Specifies the positioning area of the background images	3
<a href="#"><u>background-size</u></a>	Specifies the size of the background images	3

# Reference

## ❖ Border and Outline Properties

Property	Description	CSS
<a href="#"><u>border-bottom-left-radius</u></a>	Defines the shape of the border of the bottom-left corner	3
<a href="#"><u>border-bottom-right-radius</u></a>	Defines the shape of the border of the bottom-right corner	3
<a href="#"><u>border-image</u></a>	A shorthand property for setting all the border-image-* properties	3
<a href="#"><u>border-image-outset</u></a>	Specifies the amount by which the border image area extends beyond the border box	3
<a href="#"><u>border-image-repeat</u></a>	Specifies whether the image-border should be repeated, rounded or stretched	3
<a href="#"><u>border-image-slice</u></a>	Specifies the inward offsets of the image-border	3
<a href="#"><u>border-image-source</u></a>	Specifies an image to be used as a border	3
<a href="#"><u>border-image-width</u></a>	Specifies the widths of the image-border	3
<a href="#"><u>border-radius</u></a>	A shorthand property for setting all the four border-*-radius properties	3
<a href="#"><u>border-top-left-radius</u></a>	Defines the shape of the border of the top-left corner	3
<a href="#"><u>border-top-right-radius</u></a>	Defines the shape of the border of the top-right corner	3
<a href="#"><u>box-decoration-break</u></a>		3
<a href="#"><u>box-shadow</u></a>	Attaches one or more drop-shadows to the box	3

# Reference

## ❖ Box Properties

Property	Description	CSS
<a href="#"><u>overflow-x</u></a>	Specifies whether or not to clip the left/right edges of the content, if it overflows the element's content area	3
<a href="#"><u>overflow-y</u></a>	Specifies whether or not to clip the top/bottom edges of the content, if it overflows the element's content area	3
<a href="#"><u>overflow-style</u></a>	Specifies the preferred scrolling method for elements that overflow	3
<a href="#"><u>rotation</u></a>	Rotates an element around a given point defined by the rotation-point property	3
<a href="#"><u>rotation-point</u></a>	Defines a point as an offset from the top left border edge	3

# Reference

## ❖ Content for Paged Media Properties

Property	Description	CSS
bookmark-label	Specifies the label of the bookmark	3
bookmark-level	Specifies the level of the bookmark	3
bookmark-target	Specifies the target of the bookmark link	3
float-offset	Pushes floated elements in the opposite direction of the where they have been floated with float	3
hyphenate-after	Specifies the minimum number of characters in a hyphenated word after the hyphenation character	3
hyphenate-before	Specifies the minimum number of characters in a hyphenated word before the hyphenation character	3
hyphenate-character	Specifies a string that is shown when a hyphenate-break occurs	3
hyphenate-lines	Indicates the maximum number of successive hyphenated lines in an element	3
hyphenate-resource	Specifies a comma-separated list of external resources that can help the browser determine hyphenation points	3
hyphens	Sets how to split words to improve the layout of paragraphs	3
image-resolution	Specifies the correct resolution of images	3
marks	Adds crop and/or cross marks to the document	3
string-set		3

# Reference

## ❖ Dimension Properties

Property	Description	CSS
<u><a href="#">height</a></u>	Sets the height of an element	1
<u><a href="#">max-height</a></u>	Sets the maximum height of an element	2
<u><a href="#">max-width</a></u>	Sets the maximum width of an element	2
<u><a href="#">min-height</a></u>	Sets the minimum height of an element	2
<u><a href="#">min-width</a></u>	Sets the minimum width of an element	2
<u><a href="#">width</a></u>	Sets the width of an element	1



# Reference

## ❖ Flexible Box Properties

Property	Description	CSS
<a href="#"><u>box-align</u></a>	Specifies how to align the child elements of a box	3
<a href="#"><u>box-direction</u></a>	Specifies in which direction the children of a box are displayed	3
<a href="#"><u>box-flex</u></a>	Specifies whether the children of a box is flexible or inflexible in size	3
<a href="#"><u>box-flex-group</u></a>	Assigns flexible elements to flex groups	3
<a href="#"><u>box-lines</u></a>	Specifies whether columns will go onto a new line whenever it runs out of space in the parent box	3
<a href="#"><u>box-ordinal-group</u></a>	Specifies the display order of the child elements of a box	3
<a href="#"><u>box-orient</u></a>	Specifies whether the children of a box should be laid out horizontally or vertically	3
<a href="#"><u>box-pack</u></a>	Specifies the horizontal position in horizontal boxes and the vertical position in vertical boxes	3

# Reference

## ❖ Generated Content Properties

Property	Description	CSS
<u><a href="#">content</a></u>	Used with the :before and :after pseudo-elements, to insert generated content	2
<u><a href="#">counter-increment</a></u>	Increments one or more counters	2
<u><a href="#">counter-reset</a></u>	Creates or resets one or more counters	2
<u><a href="#">quotes</a></u>	Sets the type of quotation marks for embedded quotations	2
crop	Allows a replaced element to be just a rectangular area of an object, instead of the whole object	3
move-to	Causes an element to be removed from the flow and reinserted at a later point in the document	3
page-policy	Determines which page-based occurrence of a given element is applied to a counter or string value	3

# Reference

## ❖ Grid Properties

Property	Description	CSS
<a href="#"><u>grid-columns</u></a>	Specifies the width of each column in a grid	3
<a href="#"><u>grid-rows</u></a>	Specifies the height of each column in a grid	3

## ❖ Hyperlink Properties

Property	Description	CSS
<a href="#"><u>target</u></a>	A shorthand property for setting the target-name, target-new, and target-position properties	3
<a href="#"><u>target-name</u></a>	Specifies where to open links (target destination)	3
<a href="#"><u>target-new</u></a>	Specifies whether new destination links should open in a new window or in a new tab of an existing window	3
<a href="#"><u>target-position</u></a>	Specifies where new destination links should be placed	3

# Reference

## ❖ Linebox Properties

Property	Description	CSS
alignment-adjust	Allows more precise alignment of elements	3
alignment-baseline	Specifies how an inline-level element is aligned with respect to its parent	3
baseline-shift	Allows repositioning of the dominant-baseline relative to the dominant-baseline	3
dominant-baseline	Specifies a scaled-baseline-table	3
drop-initial-after-adjust	Sets the alignment point of the drop initial for the primary connection point	3
drop-initial-after-align	Sets which alignment line within the initial line box is used at the primary connection point with the initial letter box	3
drop-initial-before-adjust	Sets the alignment point of the drop initial for the secondary connection point	3
drop-initial-before-align	Sets which alignment line within the initial line box is used at the secondary connection point with the initial letter box	3
drop-initial-size	Controls the partial sinking of the initial letter	3
drop-initial-value	Activates a drop-initial effect	3
inline-box-align	Sets which line of a multi-line inline block align with the previous and next inline elements within a line	3
line-stacking	A shorthand property for setting the line-stacking-strategy, line-stacking-ruby, and line-stacking-shift properties	3
line-stacking-ruby	Sets the line stacking method for block elements containing ruby annotation elements	3
line-stacking-shift	Sets the line stacking method for block elements containing elements with base-shift	3
line-stacking-strategy	Sets the line stacking strategy for stacked line boxes within a containing block element	3
text-height	Sets the block-progression dimension of the text content area of an inline box	3

# Reference

## ❖ List Properties

Property	Description	CSS
<a href="#"><u>list-style</u></a>	Sets all the properties for a list in one declaration	1
<a href="#"><u>list-style-image</u></a>	Specifies an image as the list-item marker	1
<a href="#"><u>list-style-position</u></a>	Specifies if the list-item markers should appear inside or outside the content flow	1
<a href="#"><u>list-style-type</u></a>	Specifies the type of list-item marker	1

## ❖ Margin Properties

Property	Description	CSS
<a href="#"><u>margin</u></a>	Sets all the margin properties in one declaration	1
<a href="#"><u>margin-bottom</u></a>	Sets the bottom margin of an element	1
<a href="#"><u>margin-left</u></a>	Sets the left margin of an element	1
<a href="#"><u>margin-right</u></a>	Sets the right margin of an element	1
<a href="#"><u>margin-top</u></a>	Sets the top margin of an element	1

# Reference

## ❖ Marquee Properties

Property	Description	CSS
marquee-direction	Sets the direction of the moving content	3
marquee-play-count	Sets how many times the content move	3
marquee-speed	Sets how fast the content scrolls	3
marquee-style	Sets the style of the moving content	3

# Reference

## ❖ Multi-column Properties

Property	Description	CSS
<a href="#"><u>column-count</u></a>	Specifies the number of columns an element should be divided into	3
<a href="#"><u>column-fill</u></a>	Specifies how to fill columns	3
<a href="#"><u>column-gap</u></a>	Specifies the gap between the columns	3
<a href="#"><u>column-rule</u></a>	A shorthand property for setting all the column-rule-* properties	3
<a href="#"><u>column-rule-color</u></a>	Specifies the color of the rule between columns	3
<a href="#"><u>column-rule-style</u></a>	Specifies the style of the rule between columns	3
<a href="#"><u>column-rule-width</u></a>	Specifies the width of the rule between columns	3
<a href="#"><u>column-span</u></a>	Specifies how many columns an element should span across	3
<a href="#"><u>column-width</u></a>	Specifies the width of the columns	3
<a href="#"><u>columns</u></a>	A shorthand property for setting column-width and column-count	3

# Reference

## ❖ Padding Properties

Property	Description	CSS
<a href="#"><u>padding</u></a>	Sets all the padding properties in one declaration	1
<a href="#"><u>padding-bottom</u></a>	Sets the bottom padding of an element	1
<a href="#"><u>padding-left</u></a>	Sets the left padding of an element	1
<a href="#"><u>padding-right</u></a>	Sets the right padding of an element	1
<a href="#"><u>padding-top</u></a>	Sets the top padding of an element	1

## ❖ Paged Media Properties

Property	Description	CSS
fit	Gives a hint for how to scale a replaced element if neither its width nor its height property is auto	3
fit-position	Determines the alignment of the object inside the box	3
image-orientation	Specifies a rotation in the right or clockwise direction that a user agent applies to an image	3
page	Specifies a particular type of page where an element SHOULD be displayed	3
size	Specifies the size and orientation of the containing box for page content	3



# Reference

## ❖ Positioning Properties

Property	Description	CSS
<u><a href="#">bottom</a></u>	Specifies the bottom position of a positioned element	2
<u><a href="#">clear</a></u>	Specifies which sides of an element where other floating elements are not allowed	1
<u><a href="#">clip</a></u>	Clips an absolutely positioned element	2
<u><a href="#">cursor</a></u>	Specifies the type of cursor to be displayed	2
<u><a href="#">display</a></u>	Specifies the type of box an element should generate	1
<u><a href="#">float</a></u>	Specifies whether or not a box should float	1
<u><a href="#">left</a></u>	Specifies the left position of a positioned element	2
<u><a href="#">overflow</a></u>	Specifies what happens if content overflows an element's box	2
<u><a href="#">position</a></u>	Specifies the type of positioning method used for an element (static, relative, absolute or fixed)	2
<u><a href="#">right</a></u>	Specifies the right position of a positioned element	2
<u><a href="#">top</a></u>	Specifies the top position of a positioned element	2
<u><a href="#">visibility</a></u>	Specifies whether or not an element is visible	2
<u><a href="#">z-index</a></u>	Sets the stack order of a positioned element	2

# Reference

## ❖ Print Properties

Property	Description	CSS
orphans	Sets the minimum number of lines that must be left at the bottom of a page when a page break occurs inside an element	2
<a href="#">page-break-after</a>	Sets the page-breaking behavior after an element	2
<a href="#">page-break-before</a>	Sets the page-breaking behavior before an element	2
<a href="#">page-break-inside</a>	Sets the page-breaking behavior inside an element	2
widows	Sets the minimum number of lines that must be left at the top of a page when a page break occurs inside an element	2

## ❖ Ruby Properties

Property	Description	CSS
ruby-align	Controls the text alignment of the ruby text and ruby base contents relative to each other	3
ruby-overhang	Determines whether, and on which side, ruby text is allowed to partially overhang any adjacent text in addition to its own base, when the ruby text is wider than the ruby base	3
ruby-position	Controls the position of the ruby text with respect to its base	3
ruby-span	Controls the spanning behavior of annotation elements	3

# Reference

## ❖ Speech Properties

Property	Description	CSS
mark	A shorthand property for setting the mark-before and mark-after properties	3
mark-after	Allows named markers to be attached to the audio stream	3
mark-before	Allows named markers to be attached to the audio stream	3
phonemes	Specifies a phonetic pronunciation for the text contained by the corresponding element	3
rest	A shorthand property for setting the rest-before and rest-after properties	3
rest-after	Specifies a rest or prosodic boundary to be observed after speaking an element's content	3
rest-before	Specifies a rest or prosodic boundary to be observed before speaking an element's content	3
voice-balance	Specifies the balance between left and right channels	3
voice-duration	Specifies how long it should take to render the selected element's content	3
voice-pitch	Specifies the average pitch (a frequency) of the speaking voice	3
voice-pitch-range	Specifies variation in average pitch	3
voice-rate	Controls the speaking rate	3
voice-stress	Indicates the strength of emphasis to be applied	3
voice-volume	Refers to the amplitude of the waveform output by the speech synthesises	3

# Reference

## ❖ 2D/3D Transform Properties

Property	Description	CSS
<a href="#"><u>transform</u></a>	Applies a 2D or 3D transformation to an element	3
<a href="#"><u>transform-origin</u></a>	Allows you to change the position on transformed elements	3
<a href="#"><u>transform-style</u></a>	Specifies how nested elements are rendered in 3D space	3
<a href="#"><u>perspective</u></a>	Specifies the perspective on how 3D elements are viewed	3
<a href="#"><u>perspective-origin</u></a>	Specifies the bottom position of 3D elements	3
<a href="#"><u>backface-visibility</u></a>	Defines whether or not an element should be visible when not facing the screen	3

# Reference

## ❖ Transition Properties

Property	Description	CSS
<a href="#"><u>transition</u></a>	A shorthand property for setting the four transition properties	3
<a href="#"><u>transition-property</u></a>	Specifies the name of the CSS property the transition effect is for	3
<a href="#"><u>transition-duration</u></a>	Specifies how many seconds or milliseconds a transition effect takes to complete	3
<a href="#"><u>transition-timing-function</u></a>	Specifies the speed curve of the transition effect	3
<a href="#"><u>transition-delay</u></a>	Specifies when the transition effect will start	3

# Reference

## ❖ User-interface Properties

Property	Description	CSS
<a href="#"><u>appearance</u></a>	Allows you to make an element look like a standard user interface element	3
<a href="#"><u>box-sizing</u></a>	Allows you to define certain elements to fit an area in a certain way	3
<a href="#"><u>icon</u></a>	Provides the author the ability to style an element with an iconic equivalent	3
<a href="#"><u>nav-down</u></a>	Specifies where to navigate when using the arrow-down navigation key	3
<a href="#"><u>nav-index</u></a>	Specifies the tabbing order for an element	3
<a href="#"><u>nav-left</u></a>	Specifies where to navigate when using the arrow-left navigation key	3
<a href="#"><u>nav-right</u></a>	Specifies where to navigate when using the arrow-right navigation key	3
<a href="#"><u>nav-up</u></a>	Specifies where to navigate when using the arrow-up navigation key	3
<a href="#"><u>outline-offset</u></a>	Offsets an outline, and draws it beyond the border edge	3
<a href="#"><u>resize</u></a>	Specifies whether or not an element is resizable by the user	3

# 3. JavaScript

# JavaScript 개요

## ❖ History

- 1995년 넷스케이프 커뮤니케이션즈 코퍼레이션의 브렌던 아이크(Brendan Eich) 개발
- 이름 변화
  - 모카(Mocha) --> 라이브스크립트(LiveScript) --> 자바스크립트
- Java와 C의 구분을 바탕으로 하나 독립적인 언어
- ECMA-262 3판에 대응하는 자바스크립트 버전은 1.5
  - **ECMAS크립트**는 자바스크립트의 표준화된 버전
- 현재 2013년 1월 자바스크립트 1.8.5 발표
- DOM은 사실 ECMA스크립트 표준의 일부 아닌 XML 표준

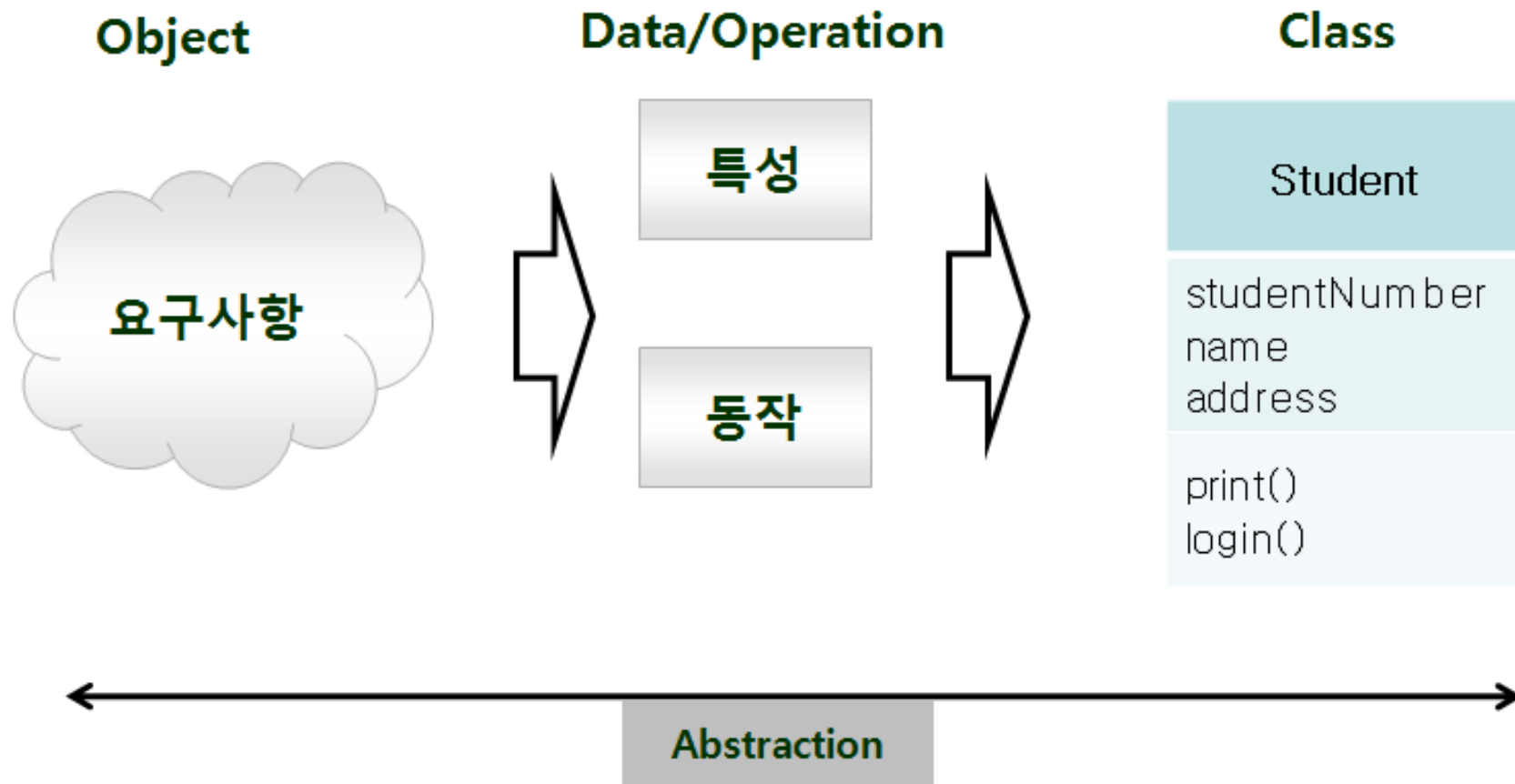


# Object

## ❖ Object

- 실세계에 존재하는 모든 것
- 프로그램 대상
- Data와 Operation으로 구분

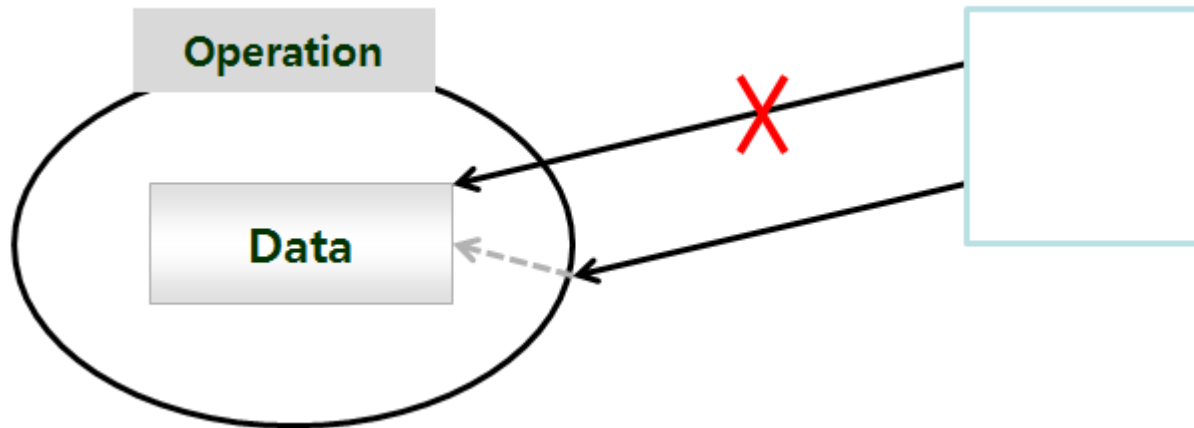
# Object



# Object

## ❖ Encapsulation

- 외부 객체가 내부의 데이터를 직접 접근 못하고 Operation을 통해서만 간접 접근이 가능
- 보안
- Operation의 재사용



# 4. JavaScript

4.1 Javascript 개요

4.2 Object

4.3 Javascript 객체

4.4 Javascript 기본 문법

4.5 사용자 정의 객체

4.6 DOM

4.7 JSON

4.8 AJAX

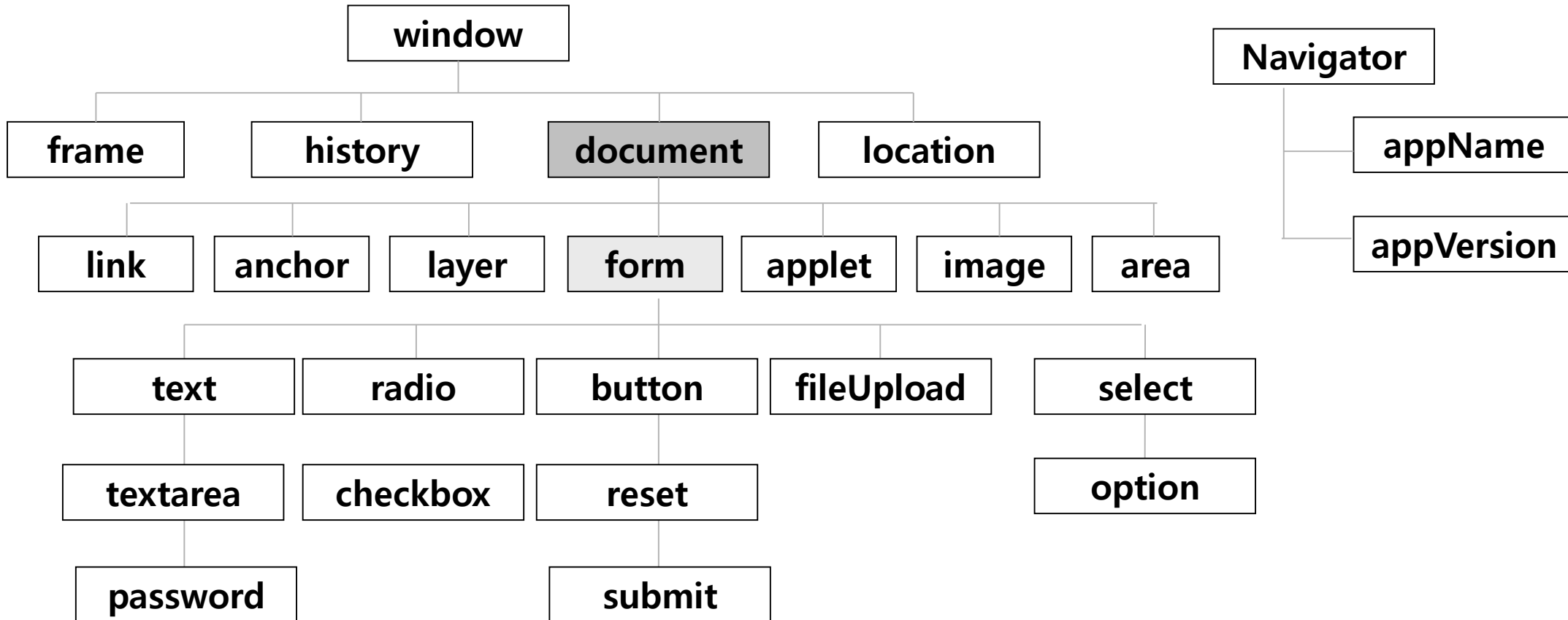
# JavaScript 객체

## ❖ 객체

- 자바 스크립트는 객체 기반 언어
- 객체에 대한 다양한 기능 제공
  - function
  - Overloading / Overriding
  - API
  - prototype
- Javascript 객체 종류
  - 브라우저 내장객체
  - 자바스크립트 내장객체
  - 사용자 정의 객체

# JavaScript 개요

## ❖ 브라우저 내장객체 계층도



# JavaScript 개요

## ❖ 기본 구문

- <head> 엘리먼트 내 선언적으로 자바스크립트 정의 및 호출
- 외부파일로 정의된 자바스크립트(.js) 불러오기
- <body>내 자바스크립트 함수 호출 및 정의

# JavaScript 기본 문법

## ❖ 자바스크립트 문법

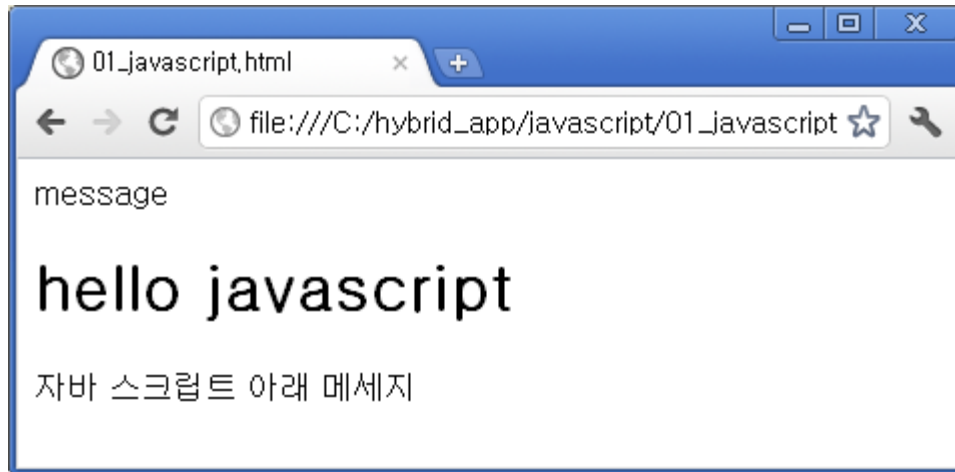
- 자바스크립트에서는 키워드를 제공한다.
- 식별자 정의
  - 키워드 불가
  - 숫자로 시작하거나 공백을 포함할 수 없다.
  - 특수문자는 \_, \$만 허용
  - 유니코드 형식(한글 등이 허용하나 일반적으로 알파벳을 사용)
  - 식별자에 ()가 있으면 함수나 생성자, 없으면 변수로 처리된다.
- 주석
  - //단일 라인
  - /\* 다중라인\*/
- Data type
  - number, string, boolean, Object(null, Object, Array), function, undefined



# JavaScript 기본 문법

## ❖ 함수 호출

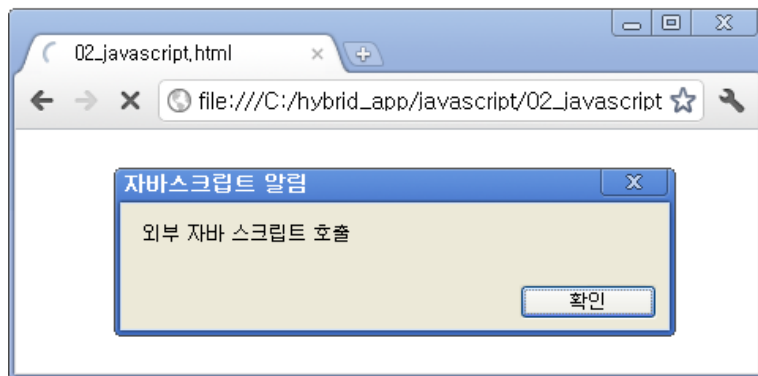
- 자바스크립트는 head, body 등 html내의 어디에서든 정의할 수 있다.



# JavaScript 기본 문법

## ❖ 함수 호출

- 외부 파일로 자바스크립트 함수 호출 구문 선언



# JavaScript 기본 문법

## ❖ 식별자 정의

- 변수 선언

`var num; //var 생략 가능`

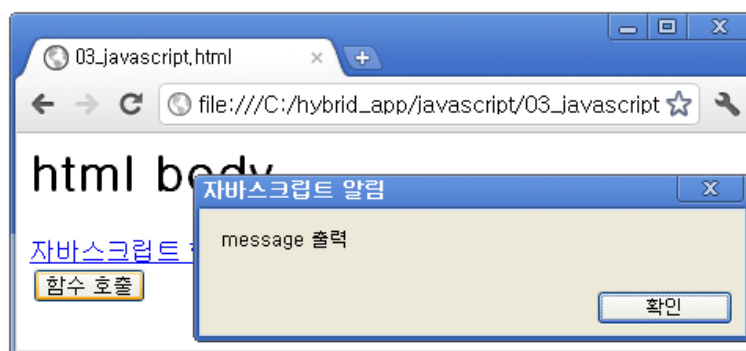
- 사용자 정의 함수

`function 이름([매개인자,...]){`

`...`

`[return 값;]`

`}`



# JavaScript 기본 문법

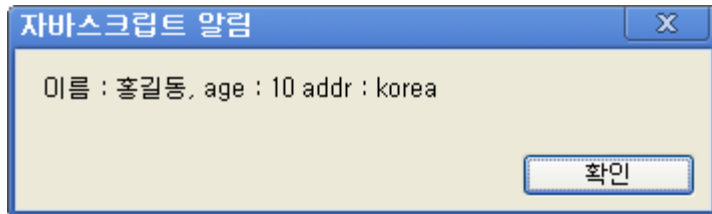
## ❖ 매개인자 있는 함수 정의

```
function message(name, age, addr){}
```

or

```
function message(var name, var age, var addr){}
```

## ❖ 자바스크립트는 함수를 호출할 때 입력된 값이 정의된 매개인자 수보다 적을 경우 앞에서 부터 순서대로 값을 매핑한다.



# JavaScript 기본 문법

## ❖ 조건문

- if/else

```
if(조건문){
 }else{
 }
```

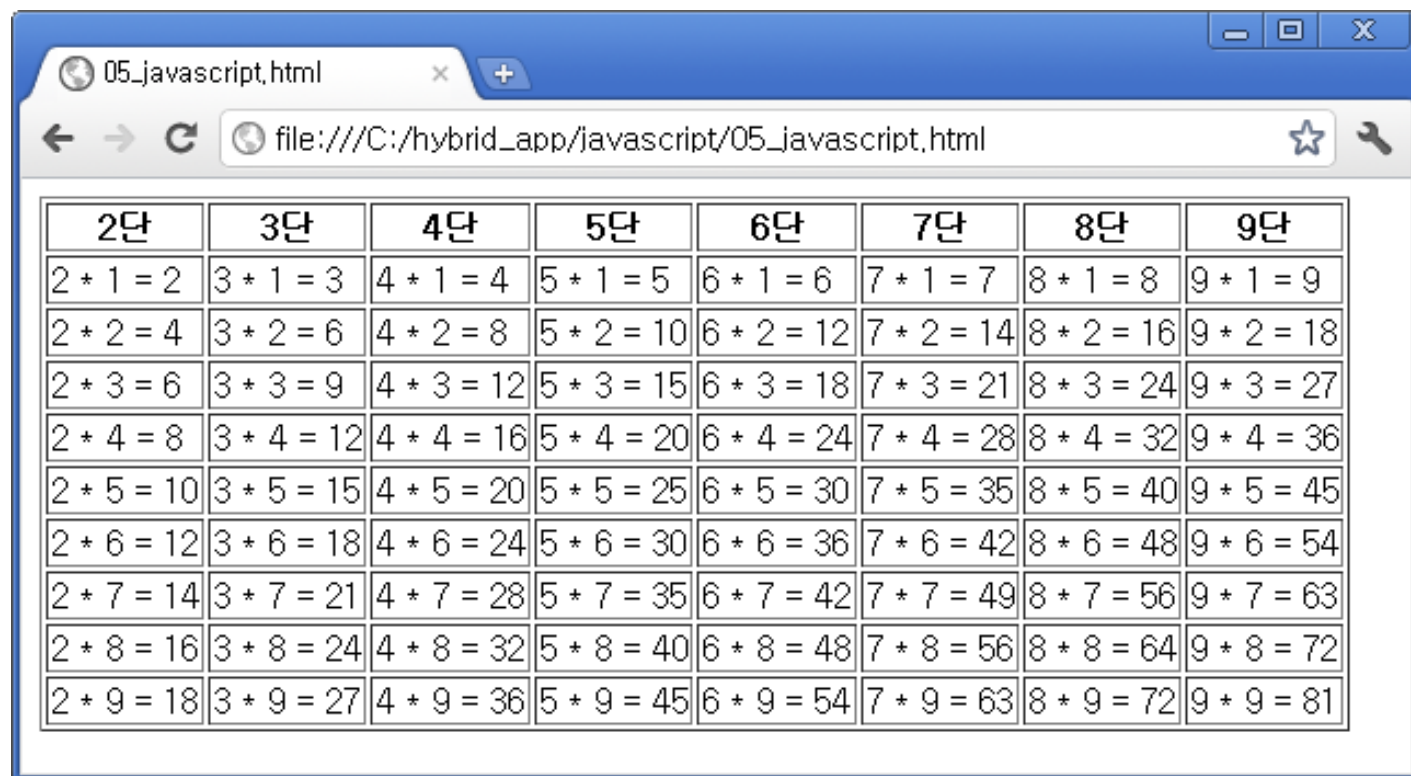
- switch/case

```
switch(변수명){
 case 값1: [break;]
 case 값2: [break;]
 case 값3: [break;]
 default :
}
```

# JavaScript 기본 문법

## ❖ 반복문

- `for( i=0 ; i<10 ; i++ ){ i ; }`
- `for( i in names ){ names[i] ; }`
- `while(i<10){ }`



A screenshot of a web browser window showing a file named '05\_javascript.html'. The address bar indicates the file path is 'file:///C:/hybrid\_app/javascript/05\_javascript.html'. The main content area displays a multiplication table with 8 columns and 9 rows. The columns are labeled '2단' through '9단' (2 times table to 9 times table). Each cell contains a multiplication problem and its result, such as '2 \* 1 = 2' in the first row, first column, and '9 \* 9 = 81' in the last row, last column.

2단	3단	4단	5단	6단	7단	8단	9단
2 * 1 = 2	3 * 1 = 3	4 * 1 = 4	5 * 1 = 5	6 * 1 = 6	7 * 1 = 7	8 * 1 = 8	9 * 1 = 9
2 * 2 = 4	3 * 2 = 6	4 * 2 = 8	5 * 2 = 10	6 * 2 = 12	7 * 2 = 14	8 * 2 = 16	9 * 2 = 18
2 * 3 = 6	3 * 3 = 9	4 * 3 = 12	5 * 3 = 15	6 * 3 = 18	7 * 3 = 21	8 * 3 = 24	9 * 3 = 27
2 * 4 = 8	3 * 4 = 12	4 * 4 = 16	5 * 4 = 20	6 * 4 = 24	7 * 4 = 28	8 * 4 = 32	9 * 4 = 36
2 * 5 = 10	3 * 5 = 15	4 * 5 = 20	5 * 5 = 25	6 * 5 = 30	7 * 5 = 35	8 * 5 = 40	9 * 5 = 45
2 * 6 = 12	3 * 6 = 18	4 * 6 = 24	5 * 6 = 30	6 * 6 = 36	7 * 6 = 42	8 * 6 = 48	9 * 6 = 54
2 * 7 = 14	3 * 7 = 21	4 * 7 = 28	5 * 7 = 35	6 * 7 = 42	7 * 7 = 49	8 * 7 = 56	9 * 7 = 63
2 * 8 = 16	3 * 8 = 24	4 * 8 = 32	5 * 8 = 40	6 * 8 = 48	7 * 8 = 56	8 * 8 = 64	9 * 8 = 72
2 * 9 = 18	3 * 9 = 27	4 * 9 = 36	5 * 9 = 45	6 * 9 = 54	7 * 9 = 63	8 * 9 = 72	9 * 9 = 81

# JavaScript 기본 문법

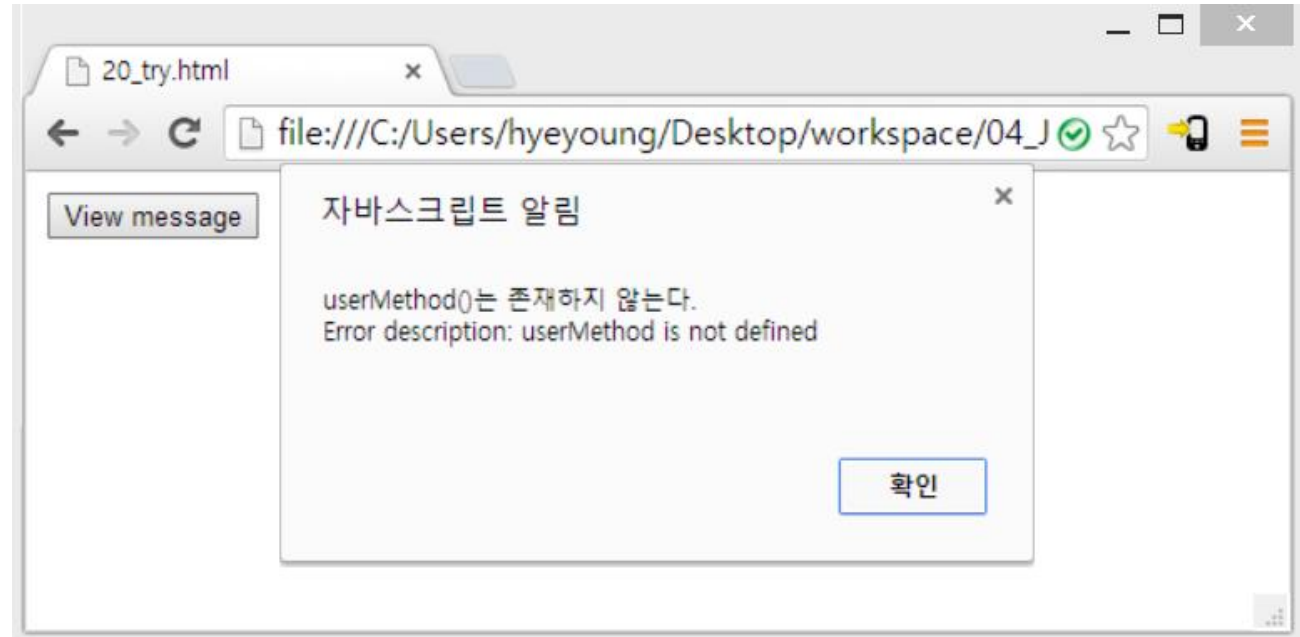
## ❖ 예외 처리

- 예외 처리

```
try{
 실행 코드
}catch(err){
 //예외 발생 처리
}finally{
 //항상 실행
}
```

- 예외 발생

```
if(names) throw new Error("names가 존재하지 않습니다");
```

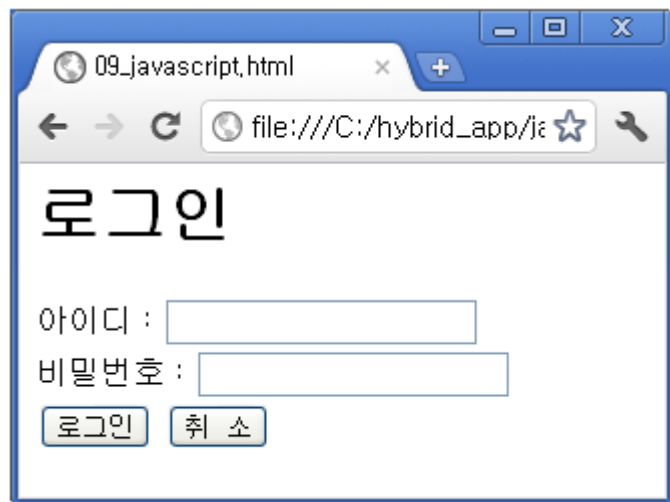


# JavaScript 기본 문법

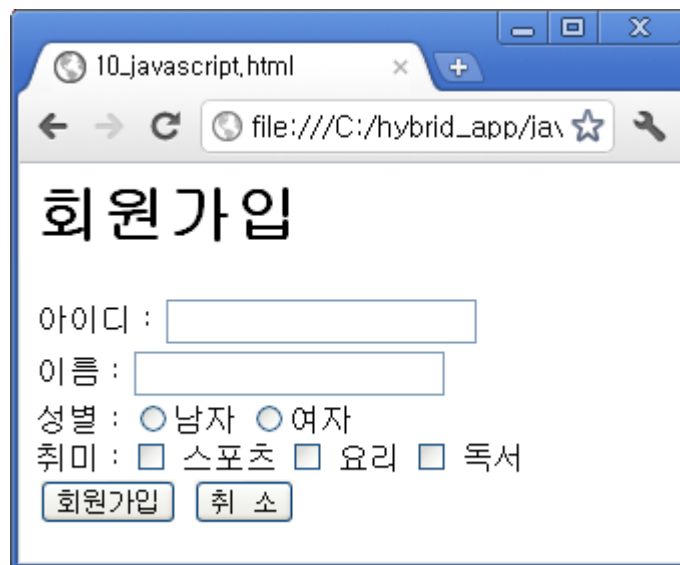
## ❖ 입력 데이터 체크

### ▪ 데이터 접근

- `window.document.forms[0].<input>name_속성값.value`
- `document.getElementById(“ ”) : Element`
- `document.getElementsByName(“ ”) : NodeList`
- `document.querySelector( tagName | .Name | #Name ) //html5`
- `document.querySelectorAll( tagName | .Name | #Name ) //html5`



A screenshot of a web browser window displaying a login form. The browser's address bar shows the file path `file:///C:/hybrid_app/jav`. The form has a title "로그인" (Login). It contains two input fields: "아이디 :" (ID) and "비밀번호 :" (Password). Below the fields are two buttons: "로그인" (Login) and "취 소" (Cancel).



A screenshot of a web browser window displaying a registration form titled "회원가입" (Sign Up). The browser's address bar shows the file path `file:///C:/hybrid_app/jav`. The form includes input fields for "아이디 :" (ID) and "이름 :" (Name). It also features radio buttons for "성별 :" (Gender) with options "남자" (Male) and "여자" (Female). Below these are checkboxes for "취미 :" (Hobby) with options "스포츠" (Sports), "요리" (Cooking), and "독서" (Reading). At the bottom are two buttons: "회원가입" (Sign Up) and "취 소" (Cancel).



# JavaScript 기본 문법

## ❖ 입력 데이터 체크

# JavaScript 기본 문법

## ❖ 입력 데이터 체크

- 체크박스 객체
- form, 체크박스, 라디오 버튼은 여러 값이 전송될 수 있어서 자동으로 배열 객체가 생성되어 처리된다.

`document.forms[0].c1[0].value`



# JavaScript 기본 문법

## ❖ Array 객체

```
var s1, s2, s3;
```

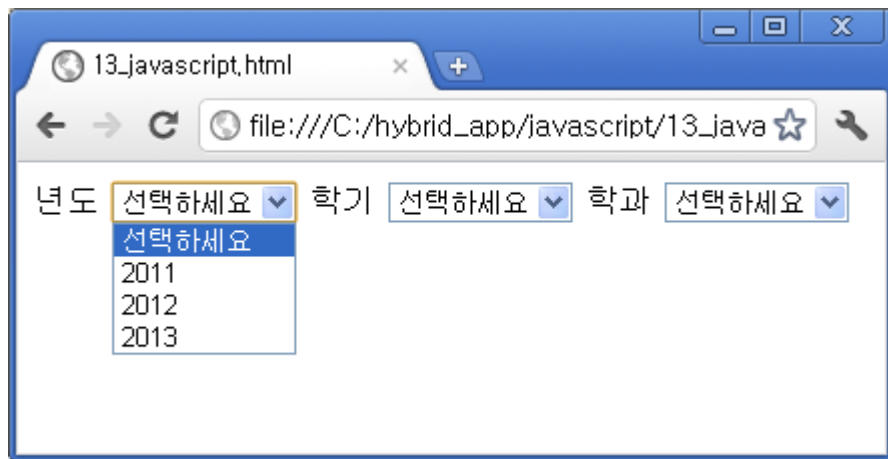
```
var s1Option =new Array("2011", "2012", "2013");
```

```
var s21Option =new Array("1학기", "2학기"); //2011
```

```
var s22Option =new Array("여름학기", "1학기", "2학기"); //2012
```

```
var s23Option =new Array("1학기", "2학기", "여름학기", "겨울학기"); //2013
```

```
var s3Option=new Array("컴퓨터학과", "미디어", "수학", "경영");
```



# JavaScript 기본 문법

## ❖ Array 객체

```
var s1, s2, s3;
```

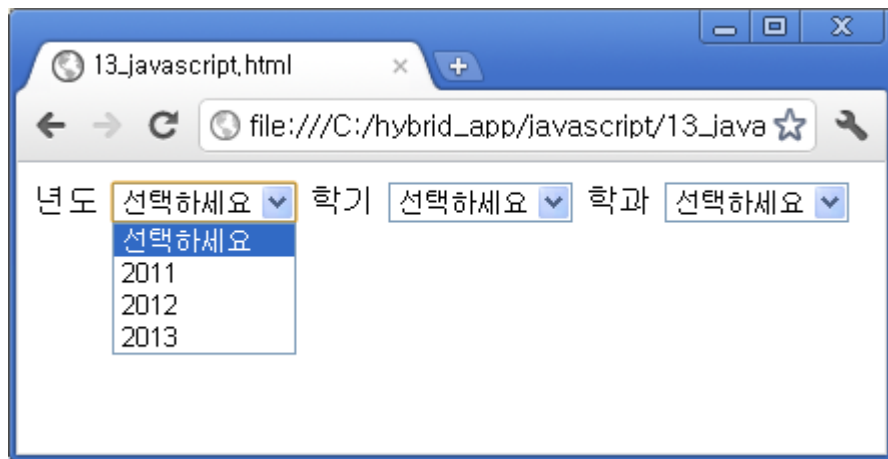
```
var s1Option =new Array("2011", "2012", "2013");
```

```
var s21Option =new Array("1학기", "2학기"); //2011
```

```
var s22Option =new Array("여름학기", "1학기", "2학기"); //2012
```

```
var s23Option =new Array("1학기", "2학기", "여름학기", "겨울학기"); //2013
```

```
var s3Option=new Array("컴퓨터학과", "미디어", "수학", "경영");
```



# 사용자 정의 객체

## ❖ function을 이용한 객체 정의

- 사용자 정의 객체는 함수의 형식을 그대로 사용
- 단 일반적으로 함수는 소문자, 객체를 표현하는 생성자는 대문자로 정의한다.
- 초기화는 구현부에 this.변수명을 정의한다.

```
function Student(name, age){
 this.name=name;
 this.age=age;
 this.method=message;
}
```

- 생성자는 new를 이용하여 사용한다.  
 stu=new Student("홍길동", 20);

# 사용자 정의 객체

## ❖ 이벤트 메서드 재정의

- 자바스크립트 내장객체의 메서드의 이름을 같게 하고 메서드의 구현부를 정의하면 사용자가 정의한 메서드가 우선된다.

```
function onload(){
```

```
 ...
```

```
}
```

```
or
```

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

```
....
```

```
}
```

# 사용자 정의 객체

## ❖ prototype을 이용한 객체 정의

- 객체에 관련된 메서드를 별도로 등록할 필요가 없다.

```
Person.prototype.getName = function(){
 return this.name;
}
```

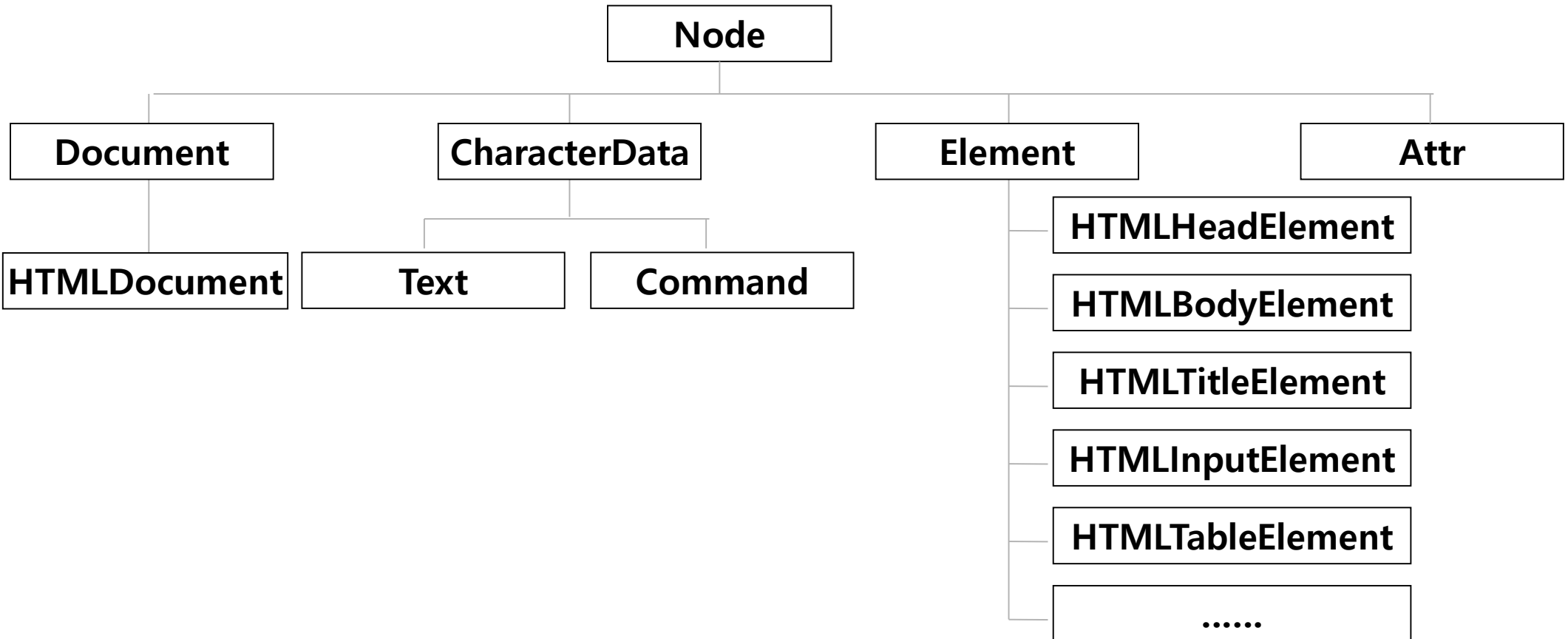
- 객체 상속에서도 사용이 가능

```
Student.prototype=new Person;
```

# DOM

## ❖ DOM 개요

- XML 표준 (<http://www.w3.org/TR/2000/PR-DOM-Level-2-Core-20000927/>)
- Document Object Model

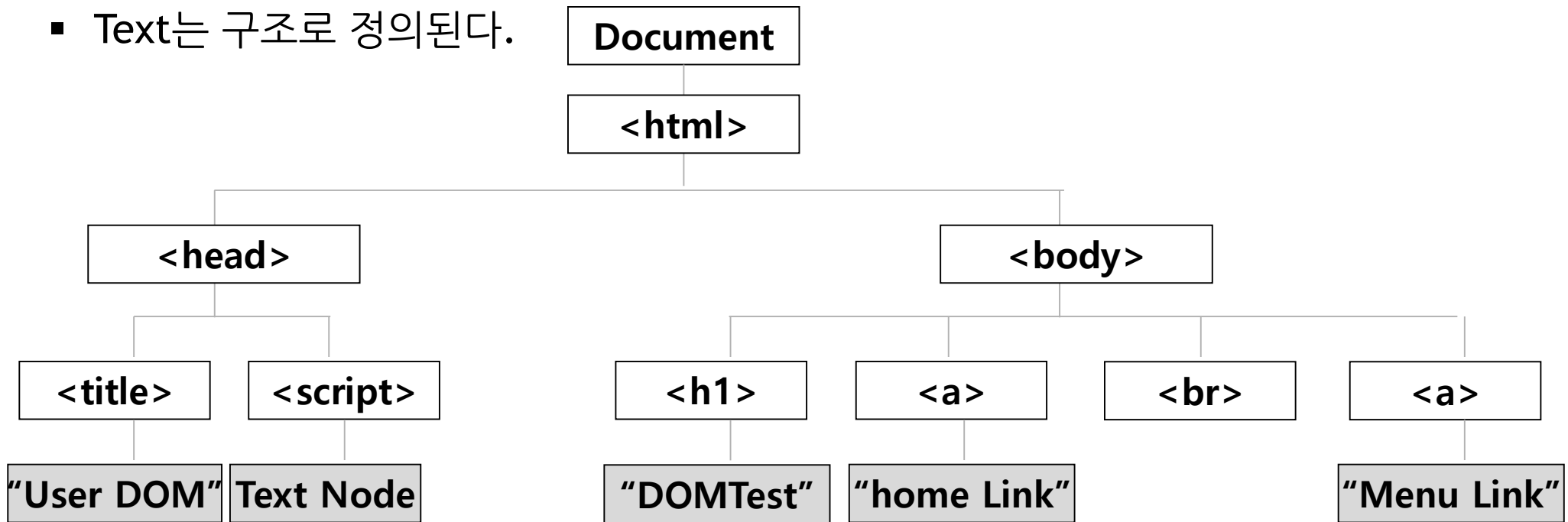




# DOM

## ❖ HTML DOM Tree

- Document는 1개의 RootElement를 가진다.
- Element는 Attr과 Text, Element를 가질 수 있다.
- Attr는 Element에 종속되고 키 = “값”으로 정의되어야 한다.
- Text는 구조로 정의된다.



# DOM

## ❖ Method

- Element 접근
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.forms[]
- Element 변경 / 추가 / 삭제
  - document.write(text)
  - document.getElementById(id).innerHTML=
  - document.getElementById(id).attribute=
  - document.getElementById(id).style.attribute=
  - document.createElement()
  - document.removeChild()
  - document.appendChild()
  - document.replaceChild()

# DOM

## ❖ Method

- Element 접근
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.forms[]
- Element 변경 / 추가 / 삭제
  - document.write(text)
  - document.getElementById(id).innerHTML=
  - document.getElementById(id).attribute=
  - document.getElementById(id).style.attribute=
  - document.createElement()
  - document.removeChild()
  - document.appendChild()
  - document.replaceChild()

# DOM

## ❖ Method

- Element 접근
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.forms[]
- Element 변경 / 추가 / 삭제
  - document.write(text)
  - document.getElementById(id).innerHTML=
  - document.getElementById(id).attribute=
  - document.getElementById(id).style.attribute=
  - document.createElement()
  - document.removeChild()
  - document.appendChild()
  - document.replaceChild()

# DOM

## ❖ Method

- Element 접근
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.forms[]
- Element 변경 / 추가 / 삭제
  - document.write(text)
  - document.getElementById(id).innerHTML=
  - document.getElementById(id).attribute=
  - document.getElementById(id).style.attribute=
  - document.createElement()
  - document.removeChild()
  - document.appendChild()
  - document.replaceChild()

# DOM

## ❖ Method

- Element 접근
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.forms[]
- Element 변경 / 추가 / 삭제
  - document.write(text)
  - document.getElementById(id).innerHTML=
  - document.getElementById(id).attribute=
  - document.getElementById(id).style.attribute=
  - document.createElement()
  - document.removeChild()
  - document.appendChild()
  - document.replaceChild()

# DOM

## ❖ Method

- Element 접근
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.forms[]
- Element 변경 / 추가 / 삭제
  - document.write(text)
  - document.getElementById(id).innerHTML=
  - document.getElementById(id).attribute=
  - document.getElementById(id).style.attribute=
  - document.createElement()
  - document.removeChild()
  - document.appendChild()
  - document.replaceChild()

# DOM

## ❖ Method

- Element 접근
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.forms[]
- Element 변경 / 추가 / 삭제
  - document.write(text)
  - document.getElementById(id).innerHTML=
  - document.getElementById(id).attribute=
  - document.getElementById(id).style.attribute=
  - document.createElement()
  - document.removeChild()
  - document.appendChild()
  - document.replaceChild()



# DOM

## ❖ Method

- Element 접근
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.forms[]
- Element 변경 / 추가 / 삭제
  - document.write(text)
  - document.getElementById(id).innerHTML=
  - document.getElementById(id).attribute=
  - document.getElementById(id).style.attribute=
  - document.createElement()
  - document.removeChild()
  - document.appendChild()
  - document.replaceChild()

# JSON

## ❖ JSON개요

### ▪ Javascript Object Notation

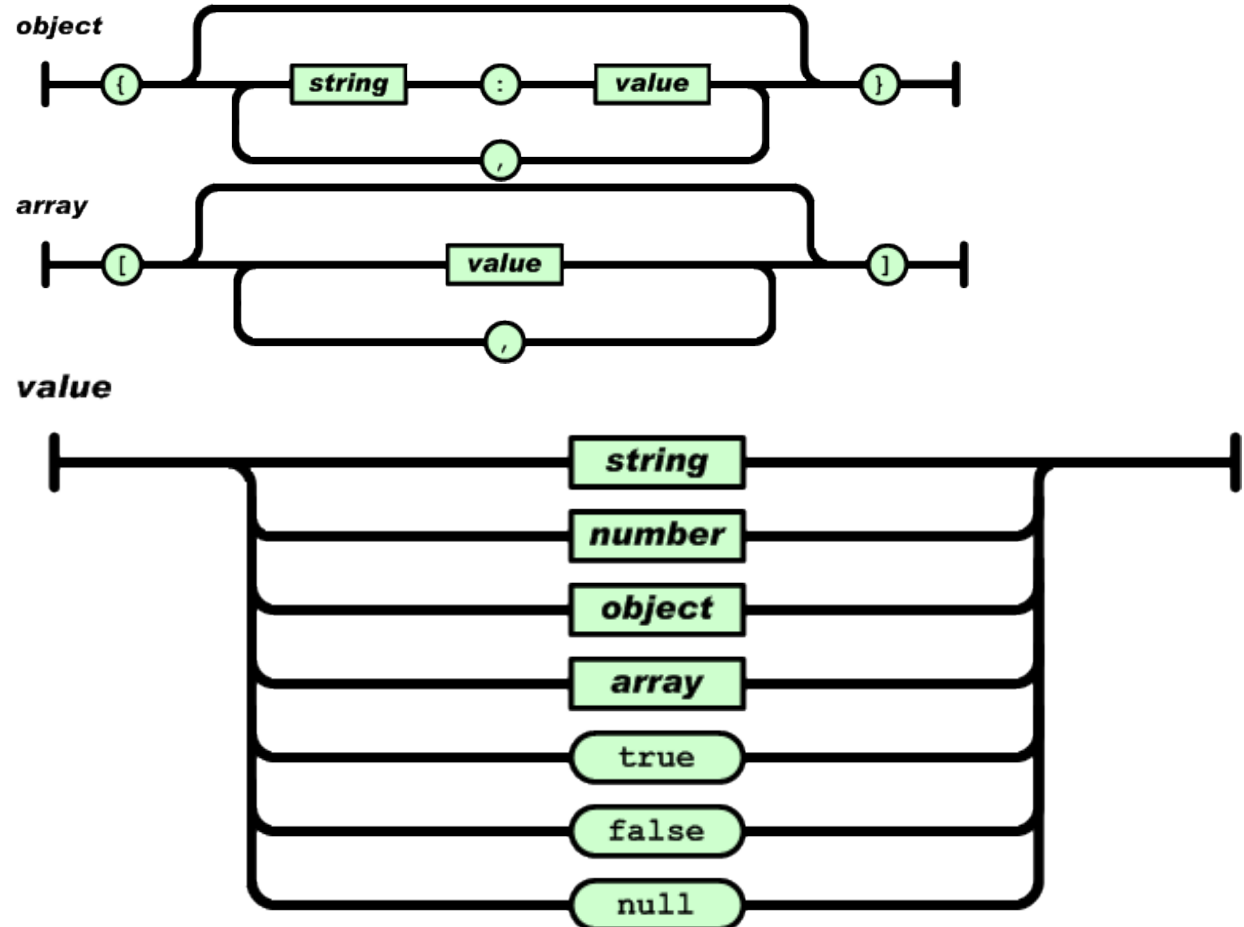
- Object

{ "key" : value }

- Array

datas [ value1, value2 ... ]

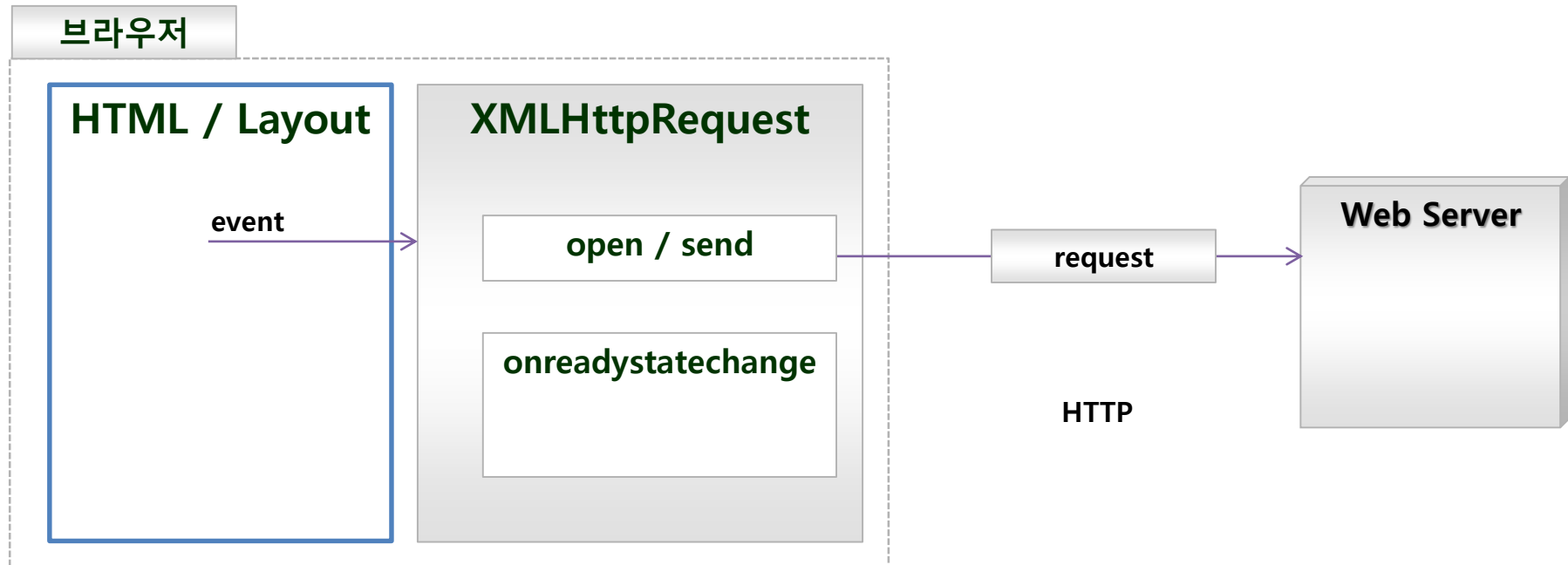
- value



# AJAX

## ❖ AJAX개요

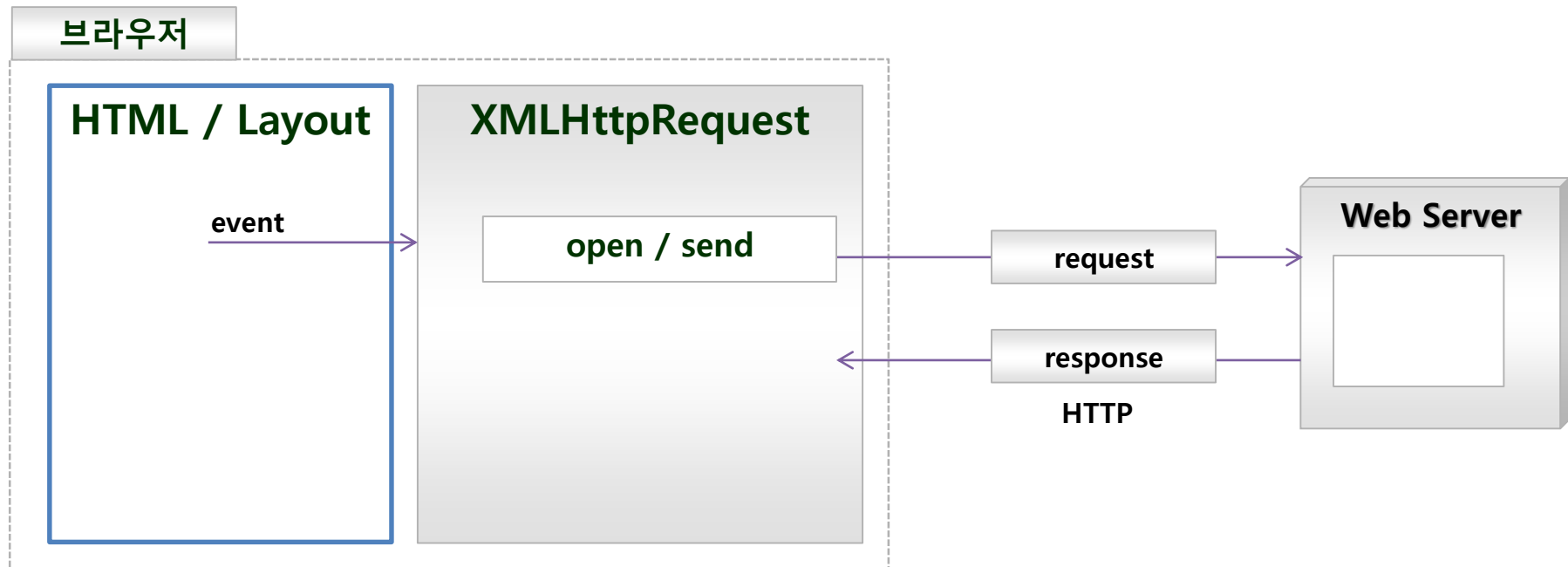
- AJAX (Asynchronous JavaScript And XML)
  - 비동기 처리를 위해 XMLHttpRequest 객체를 생성
  - 생성된 객체에서 onreadystatechange를 정의
  - 사용자가 입력받은 데이터를 구분해서 open()에 정보를 담고 send()로 요청



# AJAX

## ❖ AJAX개요

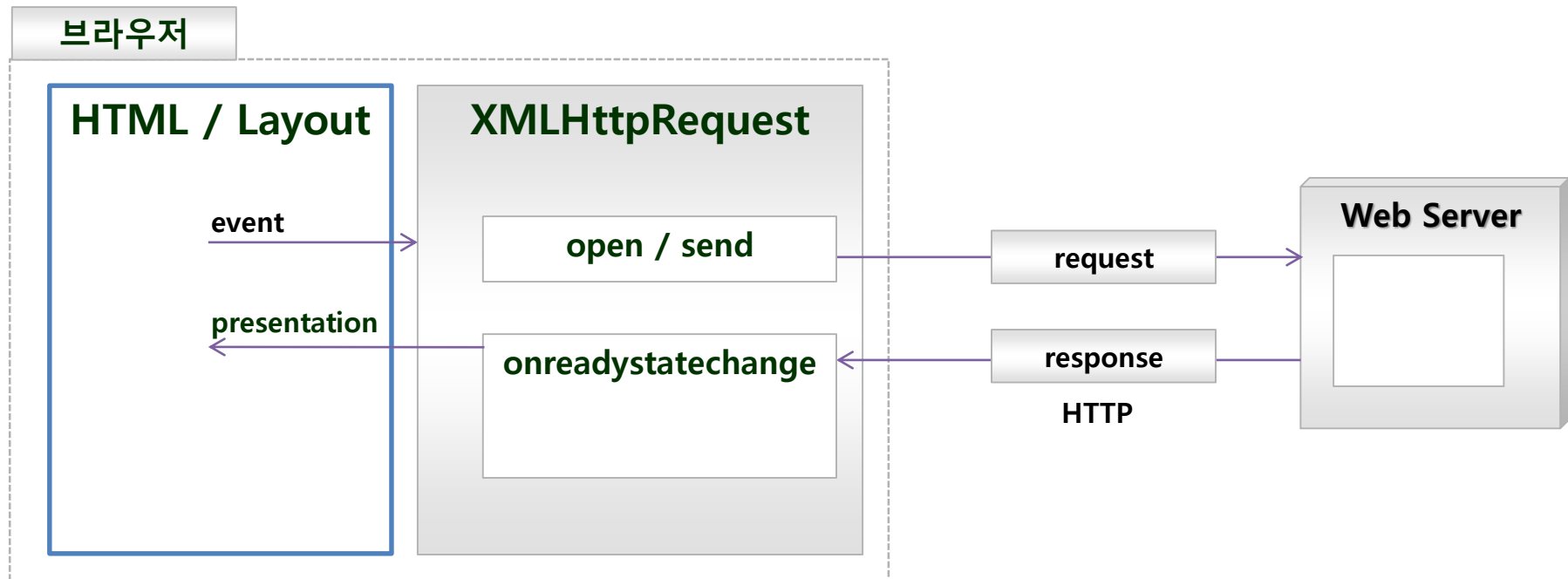
- AJAX (Asynchronous JavaScript And XML)
  - 서버는 요청받은 데이터를 처리하고 결과를 전송한다.
  - 전송 데이터는 일반 Text, JSON, XML 등을 구분할 수 있다.



# AJAX

## ❖ AJAX개요

- AJAX (Asynchronous JavaScript And XML)
  - 서버에서 전송받은 데이터는 대기중인 onreadystatechange 영역에서 처리된다.
  - status / readyState 에 따라 처리 방식을 달리 해야 한다.



## 4. HTML5 API

# WebStorage

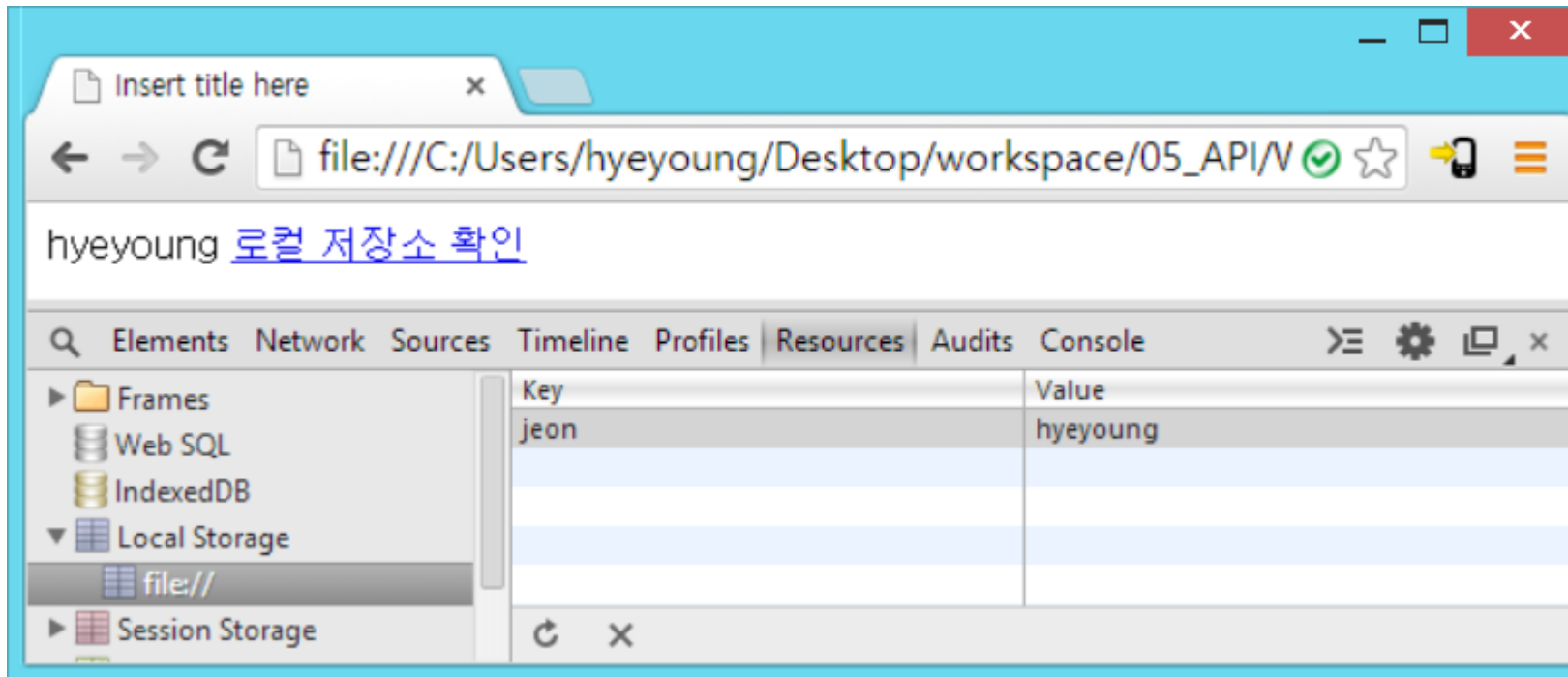
## ❖ WebStorage 개요

- 웹 서비스시 사용자의 정보를 관리 Map(Key : Value) 형태로 저장
- 쿠키의 단점(4kb, 서버에서 확인, 자동삭제)을 보완
- 도메인 별 저장소 생성
- 종류
  - localStorage : 저장기간 제한없음
  - sessionStorage : 브라우저가 닫히면 소멸
- Event
  - key : 변경된 키, 없으면 null 반환
  - oldValue : 키 이전 값, 없으면 null
  - newValue : 키 새로운 값, 없으면 null
  - url : Event가 발생한 URL
  - storageArea : 현재 저장소가 세션인지 로컬인지 확인

# WebStorage

## ❖ localStorage

- 저장소 저장 / 가져오기

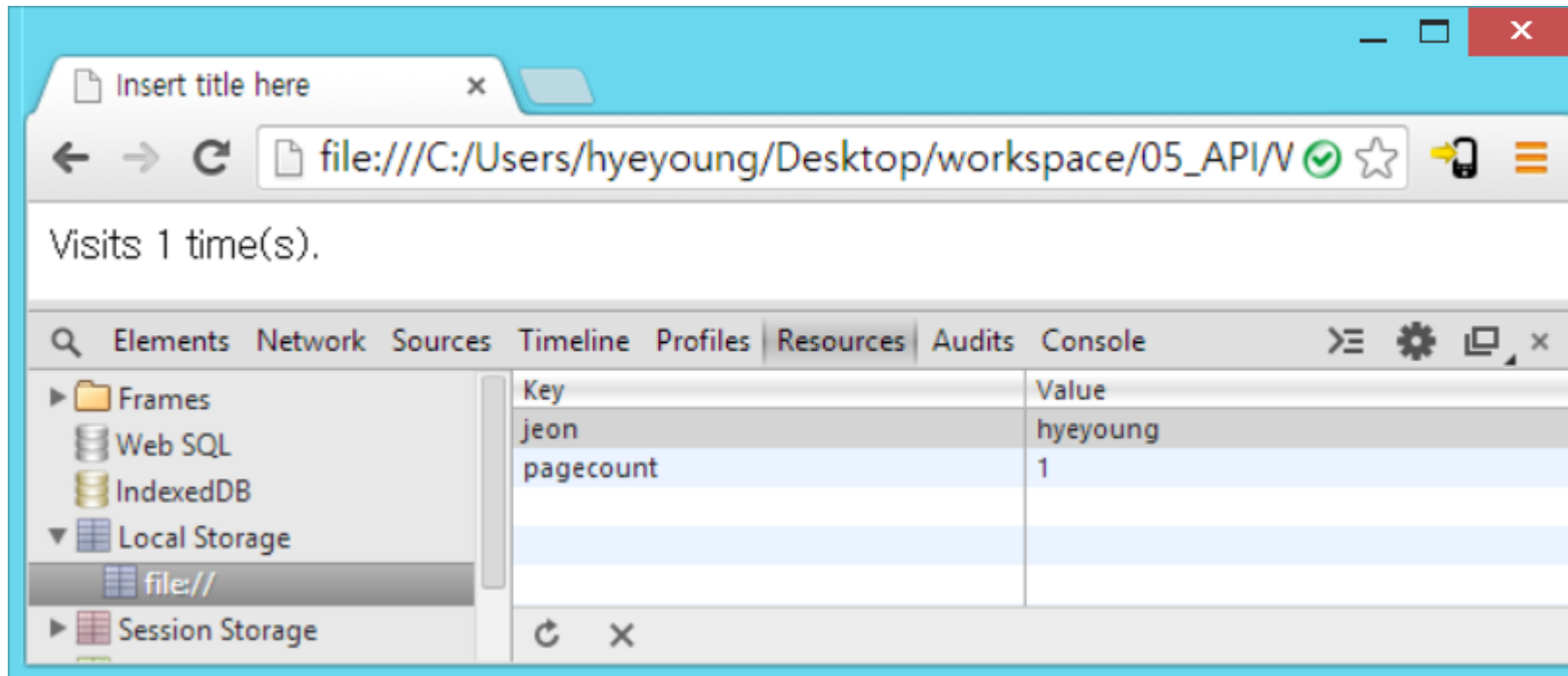




# WebStorage

## ❖ localStorage

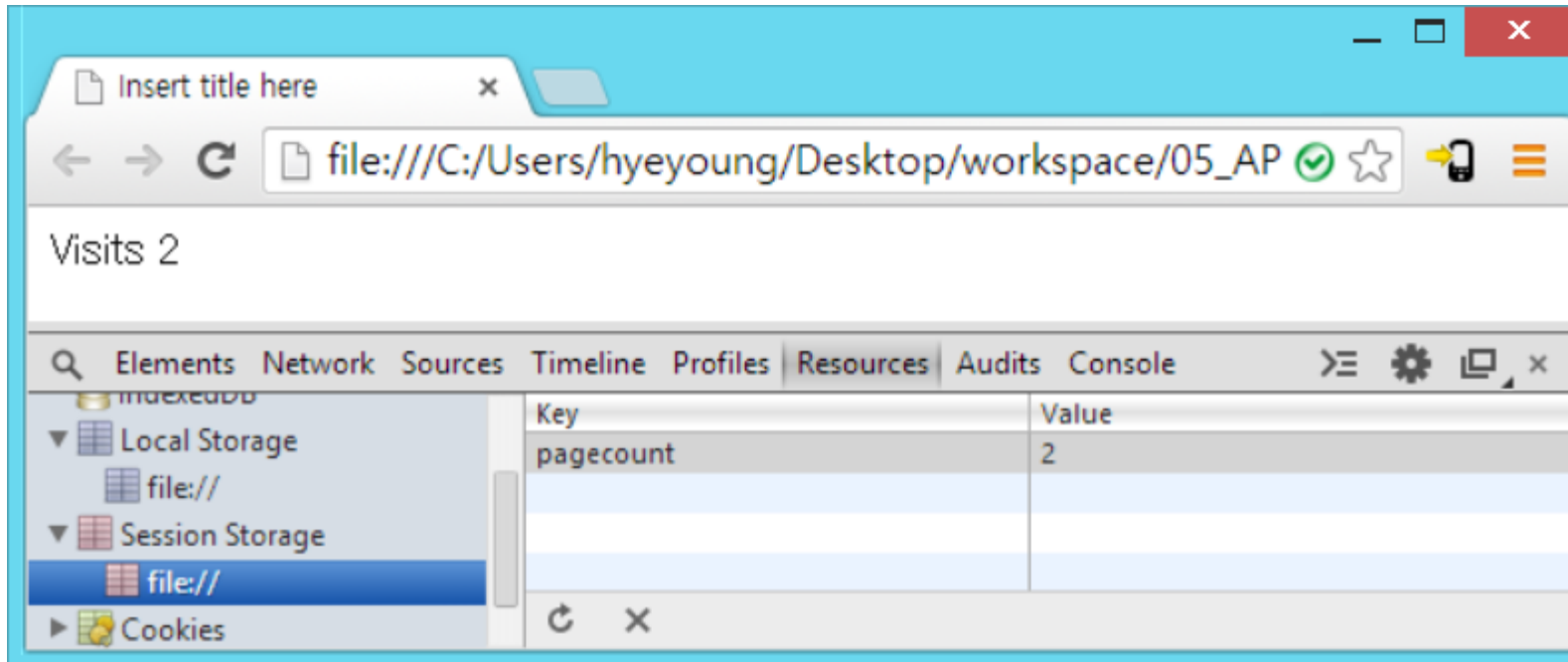
- 저장소 정보 확인



# WebStorage

## ❖ sessionStorage

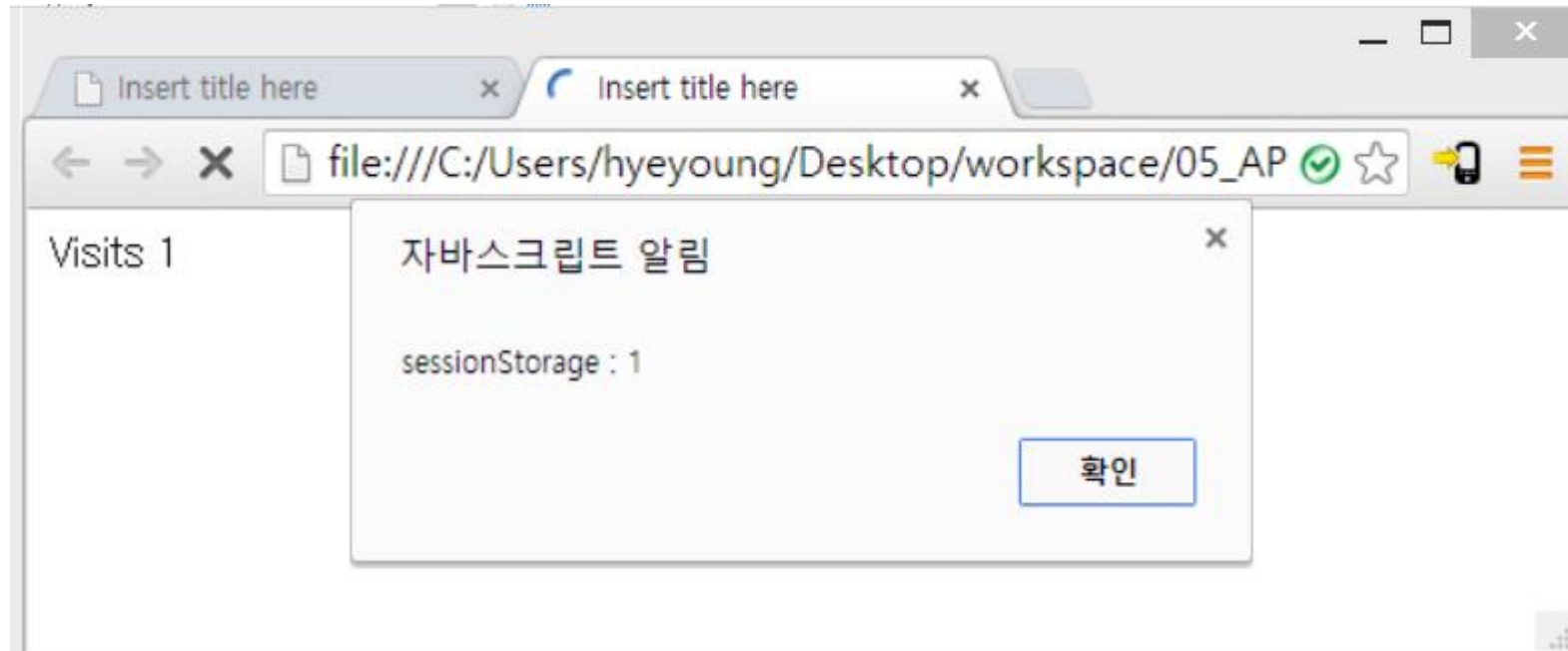
- 세션정보 등록
- 브라우저가 닫히면 값은 사라진다.
- Number() : 저장된 값을 정수로 처리할때 사용



# WebStorage

## ❖ sessionStorage

- 저장소 정보 확인



# GeoLocation

## ❖ GeoLocation 개요

- 현재 위치 정보를 파악

`getCurrentPosition()`

- 현재 위치 계속 확인

`watchPosition()`

- 중지

`clearWatch()`

- 비동기적 콜백함 수 호출

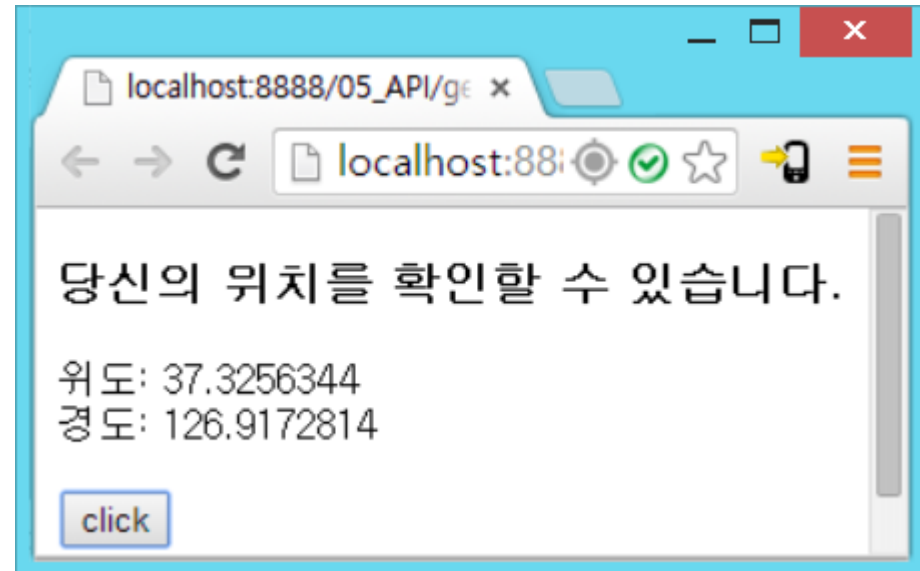
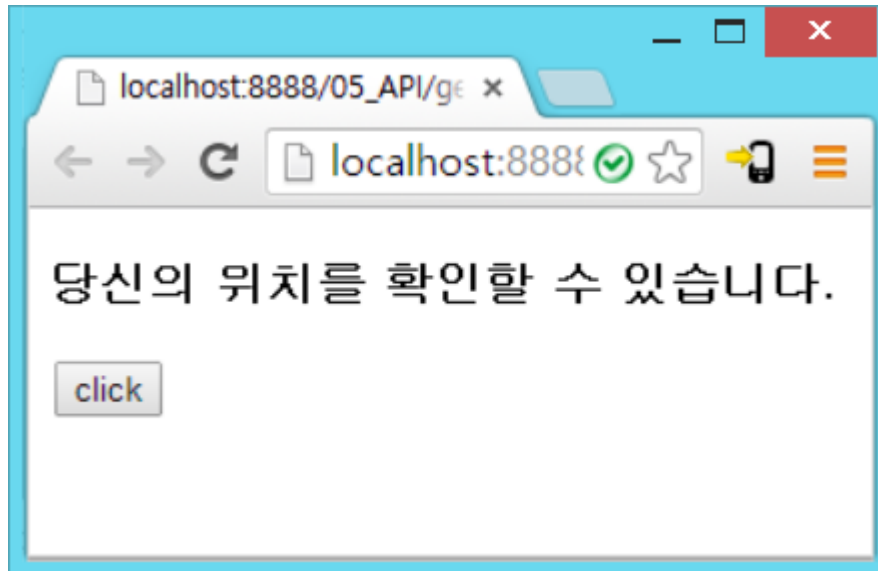
`showLocation(position)`

`errorHandler(error)`

# GeoLocation

## ❖ 현재 위치값 확인하기

- 단 테스트는 웹 서버로 확인해야 한다.



# GeoLocation

## ❖ 현재 위치 정보 이미지로 확인하기

