



클라우드 서버 교육

WEB CLIENT

문성훈

강사소개



- 문성훈
- (주) ATGLab. 개발 / 컨설팅 / IT직무교육 전문기업
- Ph.D. in Computer Science
- 대구 가톨릭대학교 IT 공학부 산학교수(현)
과학기술정보통신부 – 국가초고성능 컴퓨팅위원회 실
무위원회 민간위원(현)
- 삼성전자 / LGCNS / KT 직무교육
한국IBM / SK C&C 직무교육
LGCNS 신입사원교육 및 과정설계 / 직무평가 컨설팅
CJON 직무교육
한화시스템/ICT 신입사원교육
- Email : moon9342@gmail.com

과정목적

- HTML5 & JavaScript

- CSS Styling

- jQuery
 - jQuery Selector
 - jQuery Method
 - jQuery AJAX

개발환경

- Download

- WebStorm ↗ <https://www.jetbrains.com/webstorm/>

- Google Chrome

- <http://www.google.co.kr/chrome/browser/desktop/>

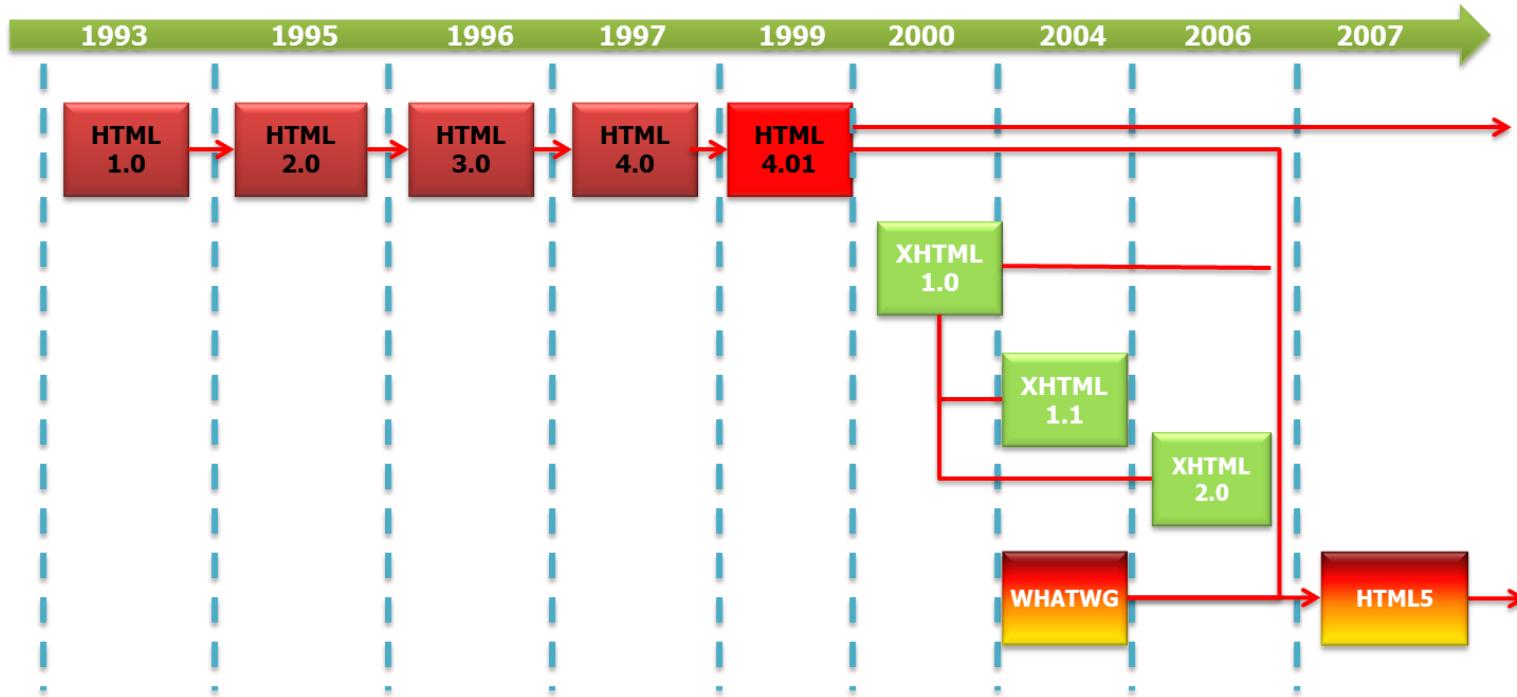
목차

- 
1. HTML5 & JavaScript
 2. Cascading Style Sheet
 3. jQuery & AJAX
 4. Mini Project

1. HTML5 & JAVASCRIPT

❖ HTML5 탄생

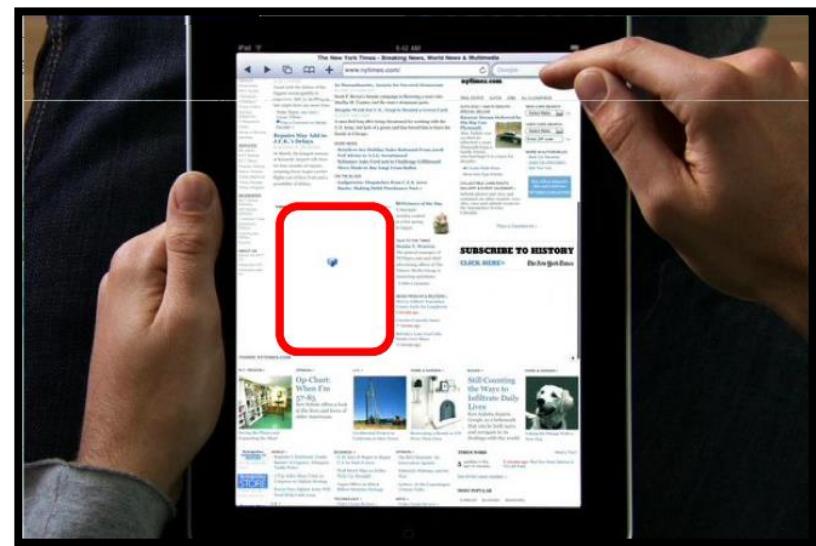
- WHATWG : Web Hypertext Application Technology Working Group



1. HTML5 & JAVASCRIPT

❖ Apple의 HTML5의 지원

- 2010년 1월 플래시를 배제하고 HTML5를 채택



1. HTML5 & JAVASCRIPT

❖ HTML5 표준화

- HTML5 표준화 (2014.10)
- Web Application을 만들기 위한 도구

1. HTML5 & JAVASCRIPT

❖ HTML5의 구성



1. HTML5 & JAVASCIPT

❖ Google의 HTML5의 지원

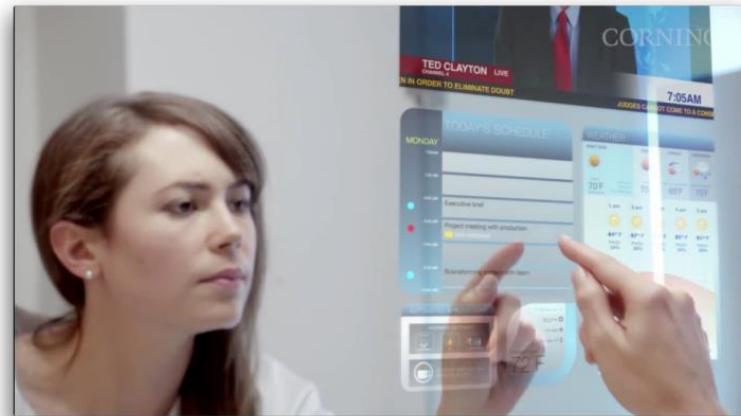
- Google Gears 개발 중단, HTML5로 방향 전환
- “Google은 HTML5에 사활을 걸었다.” (2009 Google I/O Conference)



1. HTML5 & JAVASCRIPT

❖ 왜 HTML5를 지향하는가?

- IT goes to CE market
(Consumer Electronics)
- Apple과 Google이 노리는 시장으로 소비자 가전 시장이 있다.
- Smart TV , Tablet, Set-top box,
Glass, Camera, Watch 등 그 영역을
넓히고 있다.
- 하지만 가전 시장은 제조사가 매우 다양하다는 문제



1. HTML5 & JAVASCRIPT

❖ HTML5에서 달라진 점 (Element, Attribute 측면)

- DOCTYPE의 변화

- HTML 4.01 의 DOCTYPE

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

- HTML5 DOCTYPE

```
1 | <!DOCTYPE html>
```

1. HTML5 & JAVASCRIPT

❖ HTML5에서 달라진 점 (Element, Attribute 측면)

- 불필요한 선언 제거

- HTML 4.01

```
1 | <meta http-equiv="content-type"
2 |   content="text/html; charset=UTF-8">
```

- HTML5

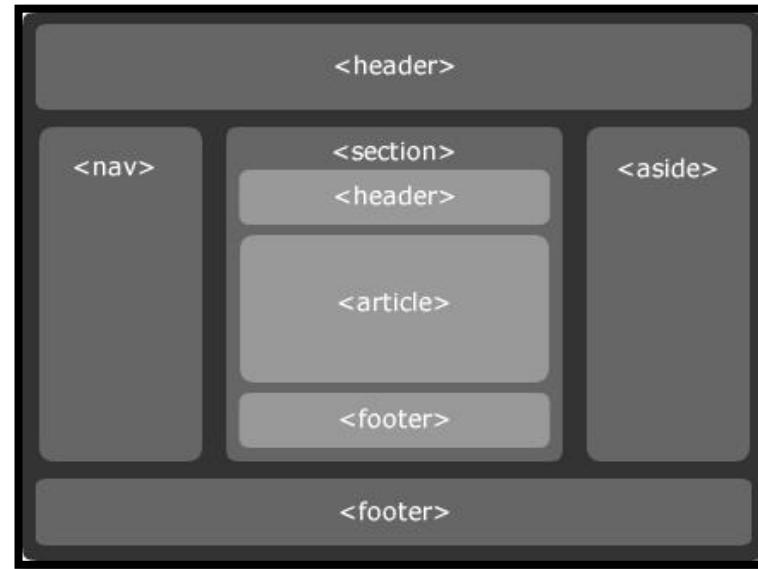
```
1 | <meta charset="UTF8">
```

1. HTML5 & JAVASCRIPT

❖ HTML5에서 달라진 점 (Element, Attribute 측면)

- 영역을 구분하는 새로운 Element
(more semantic and better structure)

```
<header>
<footer>
<nav>
<section>
<article>
<aside>
```



1. HTML5 & JAVASCRIPT

❖ HTML5에서 달라진 점 (Element, Attribute 측면)

- data-* attribute

- data-* attribute는 HTML 페이지에 임의의 데이터를 추가하는 방법.
- 유효하지 않은 HTML

```
1 | 
```

- 유효한 HTML

```
1 | 
```

1. HTML5 & JAVASCRIPT

❖ HTML5에서 달라진 점 (Element, Attribute 측면)

- 추가적인 속성

```
<input type="date">  
  
<input type="number">  
  
<input type="range">  
  
<progress value="30" max="100" ></progress>  
  
<input type="color">
```

1. HTML5 & JAVASCRIPT

❖ HTML5에서 달라진 점 (JavaScript 측면)

- <canvas> drawing API
- <video>, <audio> control API
- Web Storage API
- Indexed DB API
- Geolocation API
- Offline web apps API
- Web Socket
- Web Worker
- Drag and Drop API
- File API
- Server-Sent Event, Web Notification

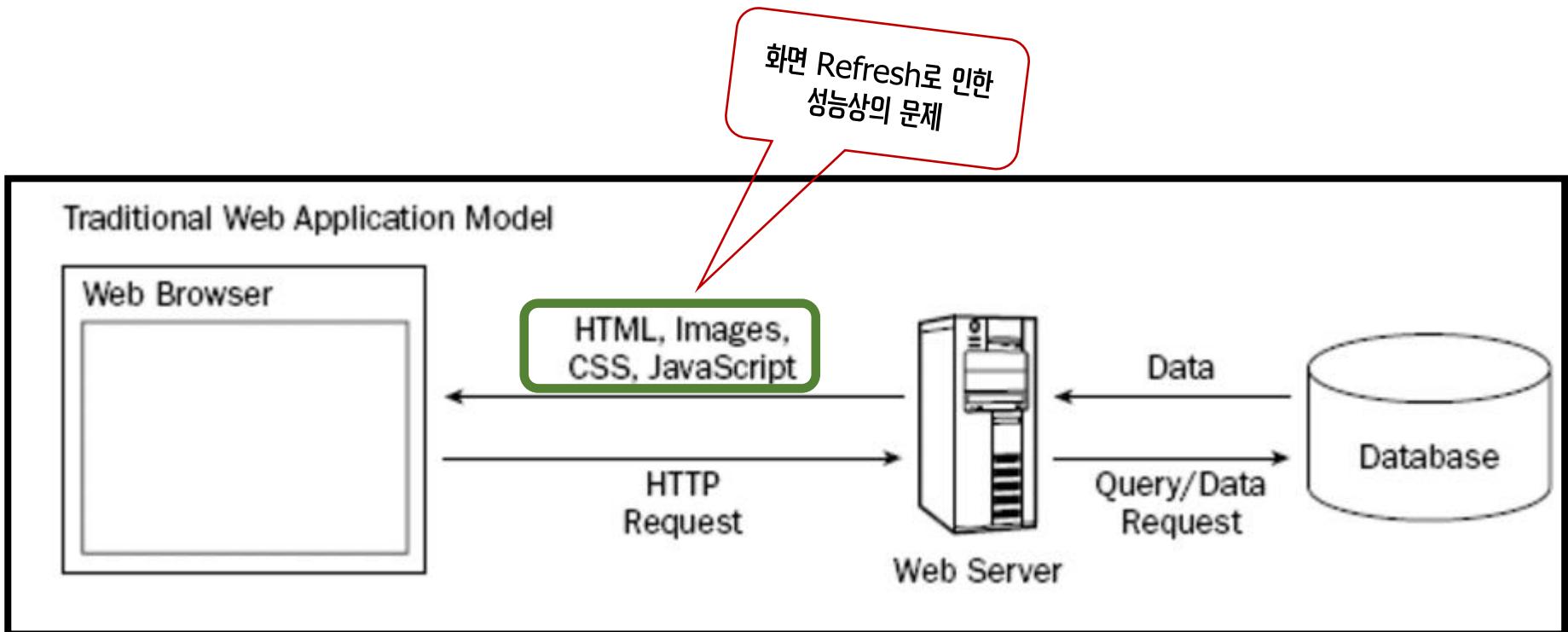
1. HTML5 & JAVASCRIPT

❖ JavaScript

- 웹 브라우저에서 사용하는 프로그래밍 언어
- 유연하지만 상당히 loose한 언어
- The world's most misunderstood programming language
- JavaScript = ECMAScript + BOM + DOM (1개의 core와 2개의 model로 구성)

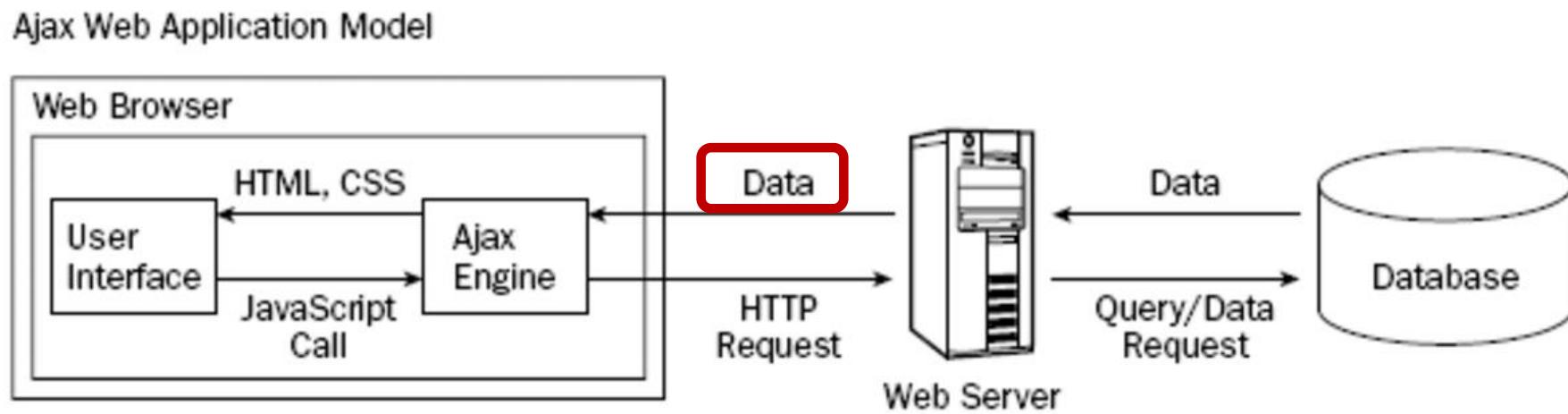
1. HTML5 & JAVASCRIPT

❖ 전통적인 Web Application Model (Round Trip 방식)



1. HTML5 & JAVASCRIPT

❖ AJAX Web Application Model (Single Page Application 방식)



1. HTML5 & JAVASCRIPT

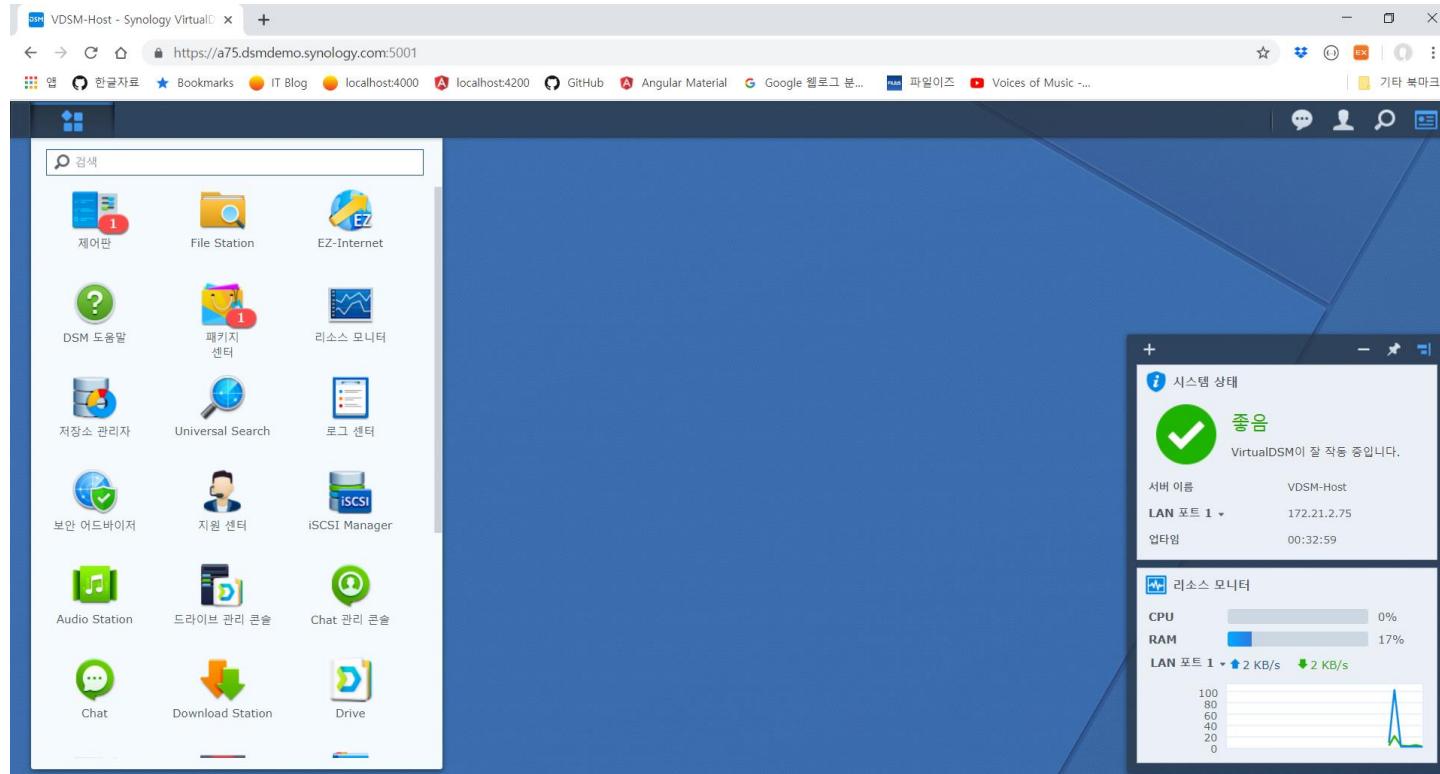
❖ Web개발 환경의 변화

- Front End Web Application 등장
- AJAX (서버와의 통신을 위한)
- Design Pattern (application의 유지보수와 재활용을 위한)
- 다양한 Framework, Library의 활용
 - Angular, React.js, View.js, Backbone, Ext JS, jQuery, jQuery Mobile, etc.

1. HTML5 & JAVASCRIPT

❖ Ext JS를 이용한 Web Application (Synology NAS DSM)

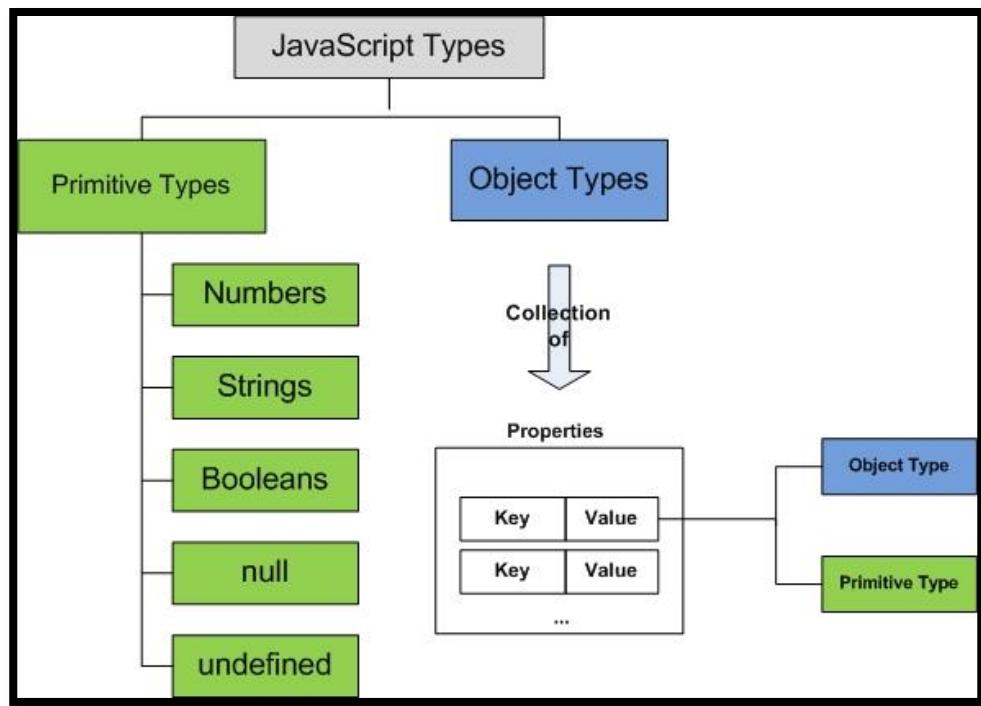
● Synology NAS DSM live demo



1. HTML5 & JAVASCRIPT

❖ JavaScript Data Type

- Primitive Type



- Reference Type (Object Type)

1. HTML5 & JAVASCRIPT

❖ JavaScript Variable

- var keyword
- var 변수에 어떤 type의 값이라도 할당 가능(weakly typed language)
- Literal
 - 각 type의 literal을 code로 확인

```
// 각 type에 대한 literal 표현
var tmp1 = "Sample";           // string literal
var tmp2 = 10;                 // number literal
var tmp3 = true;               // boolean literal
var tmp4 = [ 1, 2, 3, 4 ];     // array literal

// object literal
var tmp5 = { name : "홍길동", age : 25 };

// function literal
var tmp6 = function(x,y) {
    return x + y;
};
```

1. HTML5 & JAVASCRIPT

❖ JavaScript 함수

- 선언적 함수

```
function add(x,y) {  
    return x+y;  
}
```

- 람다 함수 (익명 함수)

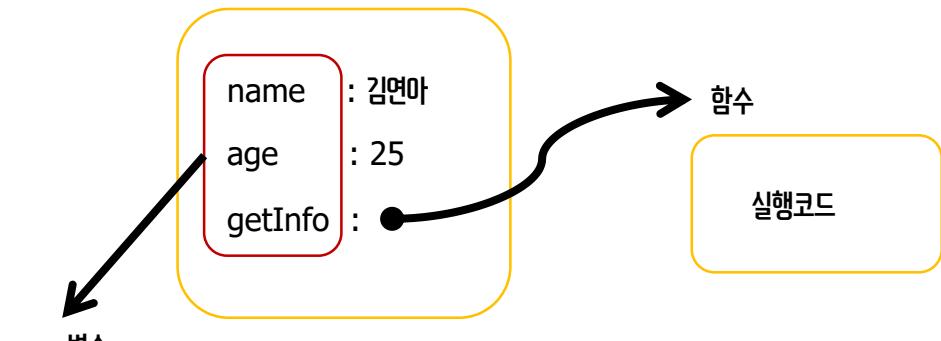
```
var add = function(x,y) { return x+y; };
```

1. HTML5 & JAVASCRIPT

❖ JavaScript 객체

- JavaScript 객체의 member는 모두 var 변수.
- Property와 Method라는 용어를 사용하지만 실제 JavaScript engine은 Property, Method, 변수를 구분하지 않는다. (모두 var 변수)

```
var obj = {  
    name : "김연아",  
    age : 25,  
    getInfo : function() {  
        return this.name;  
    }  
};
```



1. HTML5 & JAVASCRIPT

❖ JavaScript Event 처리

- Web page에 전자식 시계 출력 예제 작성

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Drag and Drop API

- 기존에도 Drag and Drop 기술이 존재.
JavaScript 의 각종 Event를 이용해 비슷하게 구현
하지만, 전용 API가 아니기에 제약사항도 많고 구현도 쉽지 않음.
- Drag and Drop 전용 API로 아주 쉽게 구현이 가능.

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Drag and Drop API

- Drag 대상(Source)

- HTML element에 draggable 속성 값을 true로 설정.

- Drop 대상(Target)

- drop 대상은 말 그대로 끌어다 놓을 곳에 해당.
 - HTML Element들의 기본 값은 drop을 받아 들이지 않게 설정.
 - drop이 가능하도록 설정하려면 event의 preventDefault() 사용.

- Drag 데이터(Data)

- Drag 데이터는 이동할 대상 컨텐츠.
 - 이동할 데이터는 dataTransfer객체를 통해 사용

1. HTML5 & JAVASCIPT

❖ HTML5 JavaScript API – Drag and Drop API

- Drag and Drop Event (1)

- dragstart

- drag 대상에서 처리해야 하는 event.
- drag가 시작 될 때 발생하는 event이며 이동할 데이터를 dataTransfer 객체에 저장

- dragenter

- drop 대상에서 처리해야 하는 event.
- 이동할 데이터를 마우스로 끌어서 drop 대상에 들어 오는 순간 발생

1. HTML5 & JAVASCIPT

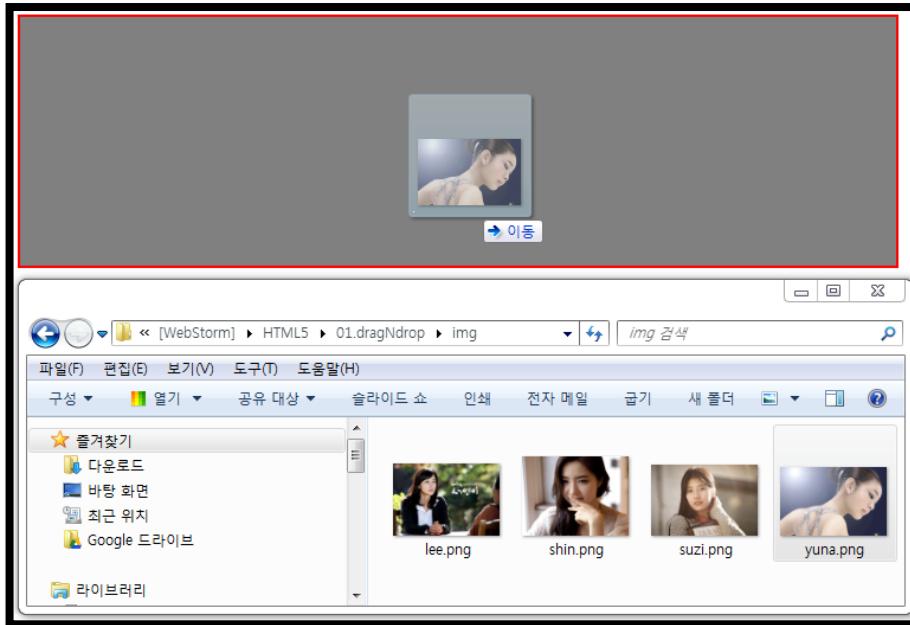
❖ HTML5 JavaScript API – Drag and Drop API

- Drag and Drop Event (2)
- dragover
 - drop 대상에서 처리해야 하는 event.
 - drop 공간을 요소가 drag한 상태로 지나갈 때 발생하는 event
- drop
 - drop 대상에서 처리해야 하는 event.
 - drop이 될 때 발생하는 event이며 dataTransfer 객체로부터 데이터를 꺼내 오고 drop 대상에 끼워 넣는 작업.

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Drag and Drop API

● 예제 작성 (File API 포함)



1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Video API

- HTML5가 나오기 전까지 Web은 Video를 지원하지 않았다.
- 단지 Plug-in을 통해서 Video 서비스가 가능 했으나, 모든 browser가 Plug-in을 사용할 수 있는 것은 아니다. (특히 Mobile 환경)
- Web browser에 소프트웨어 Codec을 탑재.
- Codec이란?
비디오 압축 및 재생 기술. 상당수 관련 기술에 특허가 걸려있어 유료 Codec이 많다.

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Video API

- <video> API
 - <source> element를 이용하여 Codec문제 해결.
 - 다양한 property를 이용하여 제어
 - JavaScript API를 이용하여 video 제어

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Canvas API

- <canvas> API
- <canvas> 태그 내에 id 지정
- 공간의 크기는 width 와 height 값으로 지정
- 그림을 그리는 부분은 canvas JavaScript API 가 담당
- <canvas> 태그에 정의된 아이디를 이용해 문서객체를 생성한 후 getContext("2d") method를 통해 context 객체를 생성.
- Context 객체를 통해 drawing API 를 이용하여 drawing.

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Canvas API

● <canvas> API – Line

- beginPath() : 선 그리기 시작 method
- moveTo(x,y) : 선 그리기 시작점 이동
- lineTo(x,y) : 선 그리기
- closePath() : 끝점과 시작점 연결 후 path 종료
- stroke() : 윤곽선 그리기
- strokeStyle : 윤곽선 색상 지정
- lineWidth : 윤곽선 두께
- lineCap : 윤곽선 모양
- fill() : 색 채우기 (default : black)
- fillStyle : fill() method의 색상 지정

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Canvas API

- <canvas> API
- Rectangle
 - fillRect(x,y,width,height) : 속이 채워져 있는 직사각형
 - strokeRect(x,y,width,height) : 선만 있는 직사각형
 - clearRect(x,y,width,height) : 직사각형 지우기
- Arc
 - arc(중심점 x, 중심점 y, 반지름, 시작 angle, 끝 angle, 시계방향여부)
- Curve
 - quadraticCurveTo(control x, control y, 끝 x, 끝 y)

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Canvas API

- <canvas> API
- Image drawing
 - drawImage(ImageObj, x, y) : 이미지 그리기
 - drawImage(ImageObj, x, y, width, height) : 이미지 그리기 (크기조절)
- Pixel handling method
 - getImageData(x, y, width, height)
캔버스로부터 이미지 정보를 구해 data값에 배열로 정보를 담음 Red/Green/Blue/Alpha 값들이 순차적으로 저장
 - putImageData(x, y)
getImageData를 이용해 구한 정보를 수정하여 반영할 때 사용
 - RGB값 조절을 이용한 필터효과 (이미지 밝기 처리 및 흑백처리)
 - HTML5 video capture 장면을 canvas에 drawing

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Canvas API



1. HTML5 & JAVASCIPT

❖ HTML5 JavaScript API – Web Storage API

- 웹 페이지의 정보는 페이지가 닫히면 모두 사라짐.
 웹 페이지의 정보를 임시적으로 저장하기 위해서 Web Storage 기능을 사용.
- Web Storage 에 저장된 데이터는 Offline 상태에서도 정상 작동. Online/Offline 동시에 지원하는 어플리케이션 개발에 유용.
- 유사 기술로 Cookie 가 있으나 유용성에서 월등히 앞선다.

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Web Storage API

- 클라이언트(브라우저)에 소량의 데이터를 저장하는 API
(이점은 기존의 Cookie 기술과 유사)
- 비교적 큰 용량을 제공. (5~10MByte)
- Cookie의 크기는 4K
- JavaScript 객체를 저장 할 수 있다. (JSON 형태)
- 서버로의 자동 전송이 없어 불필요한 traffic을 유발하지 않음.

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Web Storage API

● 데이터 저장

- `localStorage.myName = "Kim";`
- `localStorage["myName"] = "Kim";`
- `localStorage.setItem("myName", "Kim");`

● 데이터 삭제

- `delete localStorage.myName;`
- `delete localStorage["myName"];`
- `localStorage.removeItem("myName");`

● 데이터 모두 삭제

- `localStorage.clear();`

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Web Storage API

● JSON(JavaScript Object Notation)

- 인터넷에서 자료를 주고받을 때 그 자료를 표현하는 방법
- 자바스크립트 형식을 따르지만, 프로그래밍 언어나 플랫폼에 독립적이다
- JSON 문법은 자바스크립트 표준인 ECMA-262 3판의 객체 문법
- Unicode 인코딩
- 표현할 수 있는 자료형 : 숫자, 문자열, true/false, 배열, 객체
- Key : Value 구조이며 중괄호로 감싼다.
- Key 값에 따옴표를 쓸 수 있다. (일반적인 사용법)
- { “name” : “Kim” , “age” : 27 , “married” : true }

1. HTML5 & JAVASCRIPT

❖ HTML5 JavaScript API – Web Storage API

- 게시판 자동저장 구현



2. CASCADING STYLE SHEET

❖ CSS

- HTML은 문서의 내용을 담당하고, CSS는 문서의 모양을 담당.

구매주문 이력 조회

구매주문번호 | 주문일자 | 배송예정일자 |
주문자 | 공급자 | 주문상태 | 모든항목 조회 |

검색 | 초기화 |

연번	구매주문번호	설명	공급자	주문자	주문일자	배송예정일자	주문상태
1	P0-01-0001	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-09-16	2006-09-19	처리전
2	P0-01-0002	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-03-13	2006-03-19	처리전
3	P0-01-0003	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-05-12	2006-05-17	처리전
4	P0-01-0004	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-08-03	2006-08-06	처리전
5	P0-01-0005	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-03-04	2006-03-06	처리전

페이지 이동 | 1 / 5 | 1 | 2 | 3 | 4 | 5 | END | Total 21

구매주문 이력 조회

구매주문번호 | 주문일자 | 배송예정일자 |
주문자 | 공급자 | 주문상태 | 모든항목 조회 |

검색 | 초기화 |

연번	구매주문번호	설명	공급자	주문자	주문작성일	배송예정일	주문상태
1	P0-01-0001	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-09-16	2006-09-19	처리전
2	P0-01-0002	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-03-13	2006-03-19	처리전
3	P0-01-0003	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-05-12	2006-05-17	처리전
4	P0-01-0004	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-08-03	2006-08-06	처리전
5	P0-01-0005	생산1팀 부품 주문요청 처리	(주)행운전자	장원석	2006-03-04	2006-03-06	처리전

페이지 이동 | 1 / 5 | 1 | 2 | 3 | 4 | 5 | END | Total 21

2. CASCADING STYLE SHEET

❖ CSS

- CSS는 웹 페이지의 "시각적 표현"을 제어하기 위한 언어.
- 시각적 표현이란 <h1> 요소의 글꼴 크기를 지정하거나, <p>요소의 배경색을 지정하는 것을 의미.
- 원래 HTML은 문서의 "구조"를 지정하는 언어이고, 웹 페이지는 브라우저에 따라 다르게 표현하는데, 이 때 CSS를 이용하여 웹 페이지의 시각적 표현을 제어.
- 웹 페이지의 가치는 콘텐츠에 있지만 이것 만으로는 많은 사용자들을 만족시킬 수 없음.
CSS를 사용하여 디자인이라는 시각적 가치를 부여함으로써 웹 페이지 전체의 가치를 높일 수 있다.

2. CASCADING STYLE SHEET

❖ 구조와 표현의 분리

- HTML은 내용을 전달할 목적으로 만들어 짐.
초기에는 기초적인 모양과 구조만을 사용한 연구용 문서로 사용되었기에 큰 문제는 되지 않았다.
- 하지만 웹이 대중화되고 웹 페이지의 디자인이 점차 중시되면서 HTML을 "모양내기"용 언어로서 사용하기 시작. 예를 들면, 페이지 제목을 나타내는데 `<h1>`과 같은 제목 표시 element를 사용하는 것이 아니라 ``를 이용해서 원하는 글꼴 및 모양으로 바꾸는 표현하는 일이 빈번해짐.
- HTML이 이렇게 모양내기를 위해 활용되면서 불필요하고 의미 없는 태그와 속성이 많아지고, 웹 페이지가 매우 복잡해지면서, 코드의 수정이 매우 어려운 상황에 직면하게 되었다.
- 이러한 문제의 해결을 위해 CSS가 등장.

2. CASCADING STYLE SHEET

❖ 좋지 않은 예

```
1 <table width="100%" border="0" cellpadding="0" cellspacing="1px">
2   <tbody>
3     <tr height="30" style="background-color:#FFFFFF">
4       <td width="650" style="background-color:#EEFFFF"></td>
5       <td width="70" align="center">
6         <input type="submit" style="font-weight:bold" value="검색">
7       </td>
8       <td width="10"></td>
9       <td width="70" align="center">
10        <input type="button" onclick="clearForm();" style="width:200px" value="지우기">
11      </td>
12    </tr>
13  </tbody>
14 </table>
```

2. CASCADING STYLE SHEET

❖ CSS 기본 문법

Style을 적용할 대상

Style을 적용할 대상으로
<h1>을 지정

```
1 h1 {  
2   font-family: 콜림체;  
3   font-size: 20px;  
4   color: steelblue;  
5 }
```

적용될 스타일 내용

<h1> 태그에 적용될 글꼴의
모양, 크기, 색깔 지정

2. CASCADING STYLE SHEET

❖ HTML 사용

```
1 <html>
2   <head>
3     <title> CSS </title>
4   </head>
5   <body>
6     <h1>Hello World</h1>
7     This is sample message for you.
8   </body>
9 </html>
```

스타일 시트를 정의한 후 브라우저로 확인하면,
<h1> 태그 부분은 20px 크기의 파란 글씨로
표현된다

2. CASCADING STYLE SHEET

❖ CSS 기본 문법

- 스타일을 적용하는 대상
- html tag명, id명, class명 등이 해당됨
- 반드시 알파벳으로 시작해야 함
- 대소문자의 구분이 있음

선택자 { 속성 : 속성값; }

• CSS 속성명

• 선택자에 어떤 스타일을
부여할지 지정

```
1 | p { font-size: 1em; }  
2 | .odd { background-color:#edf5ff; }  
3 | #mInfo { width:400px; }
```

2. CASCADING STYLE SHEET

❖ CSS 적용 방법

- <head> 범위 안에 link를 사용하여 외부 스타일시트를 참조. (권장)

```
1 <head>
2   <link rel="stylesheet" href="css/default.css" />
3 </head>
4 <body>
5   <p>
6     웹 표준은 월드 와이드 웹의 측면을 서술하고 정의하는 공식 표준이나
7     다른 기술 규격을 가리키는 일반적인 용어이다.
8   </p>
9 </body>
```

default.css

```
1 | p { background-color: lime; }
```

2. CASCADING STYLE SHEET

❖ CSS 선택자 (Selector)

- 전체 선택자 (Universal Selector)

- 가장 기본적인 선택자
- 하위에 있는 모든 element를 선택하는 선택자
- 예제

2. CASCADING STYLE SHEET

❖ CSS 선택자 (Selector)

- 태그 선택자 (Tag Selector)

- 태그 선택자는 특정한 태그만 선택하는 선택자
- 하나 이상의 태그 선택자를 동시에 사용하고 싶을 때 콤마(,)로 선택자 구분
- 예제

2. CASCADING STYLE SHEET

❖ CSS 선택자 (Selector)

- 아이디 선택자 (ID Selector)

- HTML 페이지 내에 존재하는 id속성의 값은 각각 다르다. (unique)

- id 속성의 값을 이용하여 element를 선택하는 선택자

- 유일한 element를 선택하게 되며 속도 면에서 가장 빠르고 편하다.

- 예제

2. CASCADING STYLE SHEET

❖ CSS 선택자 (Selector)

- 클래스 선택자 (Class Selector)

- class 속성의 값을 이용하여 element를 선택하는 선택자

- 예제

2. CASCADING STYLE SHEET

❖ Block level element vs. Inline level element

- HTML element는 한 줄을 전부 차지하는 block 요소와 한 줄의 일부로 들어갈 수 있는 inline 요소로 구분

- 대표적인 block 요소

```
<h1>~<h6> , <br> , <p> , <ul> , <ol>, <div>
```

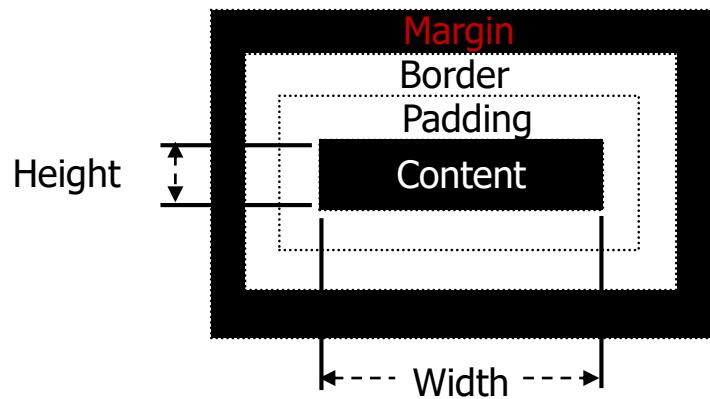
- 대표적인 inline 요소

```
<img> , <a> , <span>
```

2. CASCADING STYLE SHEET

❖ Box Model

- Box Model은 element들이 어떻게 화면상에서 보여지고 상호작용하는지에 대한 규정으로 html의 모든 element는 사각형의 박스로 다루어 진다.



- content : 요소에 포함된 내용
- margin : content의 border와 다른 요소 사이의 여백
- border : content의 테두리
- padding : content와 border 사이의 여백

- 요소에 배경을 지정하는 경우 배경스타일은 content와 padding까지 적용됨
- 마진 영역의 배경은 투명하고 보이지 않으므로, 주로 element들 간의 간격을 지정하는데 사용함

3. JQUERY & AJAX

❖ jQuery 개요

- 모든 브라우저에서 동작하는 클라이언트 자바스크립트 라이브러리
- 무료로 사용 가능한 오픈 소스 라이브러리
- jQuery 사용 목적
 - DOM과 관련된 처리 쉽게 구현
 - 일관된 이벤트 연결 쉽게 구현
 - 시각적 효과 쉽게 구현
 - Ajax 애플리케이션 쉽게 개발



3. JQUERY & AJAX

❖ jQuery 다운로드

- <http://www.jquery.com>
- CDN 방식 (Content Delivery Network)
 - 사용자에게 간편하게 Contents를 제공하는 방식
 - Google, MS, jQuery측에서 jQuery를 사용하기 편하게 URL 제공
- 일반적으로, 개발 시점에는 download 방식
서비스 시점에는 CDN방식을 이용.

3. JQUERY & AJAX

❖ jQuery Selector

- 전체 선택자 (Universal Selector)

- 가장 기본적인 선택자
- 하위에 있는 모든 문서객체를 선택하는 선택자
- 예제 (모든 문서객체의 color 스타일 속성을 “red” 로 변경)

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

● 태그 선택자 (Type Selector)

- 태그 선택자는 특정한 태그만 선택하는 선택자
- 하나 이상의 태그 선택자를 동시에 사용하고 싶을 때
콤마(,)로 선택자 구분
- 예제

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

● 아이디 선택자 (ID Selector)

- HTML 페이지 내에 존재하는 id속성의 값은 각각 다르다. (unique)
- id 속성의 값을 이용하여 문서 객체를 선택하는 선택자
- 유일한 문서객체를 선택하게 되며 속도 면에서 가장 빠르고 편하다.
- 예제

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

● 클래스 선택자 (Class Selector)

- class 속성의 값을 이용하여 문서 객체를 선택하는 선택자
- 예제

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

- 자식 선택자와 후손 선택자
(Child combinator & Descendant combinator)

- 기본 선택자의 뒤에 붙여 사용하며 기본 선택자의 범위 제한
- 자식 선택자는 “>” 기호로 표시
- 후손 선택자는 “ ” 공백으로 표시
- 예제

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

● 동위 선택자

(Adjacent sibling combinator & General sibling combinator)

- element A + element B : element A 바로 다음에 위치하는 element B를 선택
- element A ~ element B : element A 다음에 위치하는 모든 element B를 선택
- 예제

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

● 속성 선택자 (Attribute selectors)

- element[attribute] : 특정 속성을 가지고 있는 문서객체 선택
- element[attribute=value] : 속성값이 value값과 같은 문서객체 선택
- element[attribute~=value] : 속성값이 value값을 단어로서 포함하는 문서객체 선택
- element[attribute^=value] : 속성값이 value값으로 시작하는 문서객체 선택
- element[attribute\$=value] : 속성값이 value값으로 끝나는 문서객체 선택
- element[attribute*=value] : 속성값이 value값을 포함하는 문서객체 선택
- 예제

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

- 연습문제

```
<div>
  <ul>
    <li id="apple">사과</li>
    <li id="pineapple">파인애플</li>
    <li class="myList">참외</li>
  </ul>
  <form action="#" method="post">
    <input type="text" id="uId" size="20">
  </form>
  <ol>
    <li class="myList">고양이</li>
    <li class="myList">호랑이</li>
    <li class="myList">강아지</li>
  </ol>
</div>
```

- console 출력결과

사과

파인애플

참외

uld

고양이

호랑이

강아지

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

● 상태 선택자 (The UI element states pseudo-classes)

- :checked : 체크된 입력양식을 선택
- :disabled : 비활성화된 입력양식을 선택
- :enabled : 활성화된 입력양식을 선택
- :focus : 현재 포커스가 잡혀있는 입력 양식 선택
- :selected : option객체 중 선택된 객체를 선택
- 예제

3. JQUERY & AJAX

❖ jQuery Selector (cont.)

● 함수 필터 선택자

- :contains(string) : 특정 문자열을 포함하는 문서객체 선택
- :eq(n) : n번째 위치하는 문서객체 선택
- :gt(n) : n번째를 초과하는 위치에 있는 문서객체 선택
- :lt(n) : n번째 미만의 위치에 있는 문서객체 선택
- :not(선택자) : 선택자와 일치하는 않는 문서객체 선택
- :nth-child(2n+1) : 2n+1번째에 위치하는 문서객체 선택
- 예제

3. JQUERY & AJAX

❖ jQuery Method

● jQuery의 배열처리

- each() method 사용
- `$.each(object, function(index,item) { })`
- `$(selector).each(function(index,item) { })`
- 예제

3. JQUERY & AJAX

❖ jQuery Method (cont.)

- css() method

- addClass() method

- attr() method

- removeAttr() method

- html() method

- text() method

- 예제

3. JQUERY & AJAX

❖ jQuery Method (cont.)

- remove(), empty() method
- \$()
- \$(A).append(B) , \$(A).prepend(B)
- \$(A).after(B) , \$(A).before(B)
- clone() method
- 예제

3. JQUERY & AJAX

❖ jQuery Event

- jQuery 1.70이후부터는 on() method 사용
- `$(selector).on(event, function(event) { … })`
- Event handler안에서의 this는 event source 문서객체를 지칭
- 예제

3. JQUERY & AJAX

❖ jQuery Event (cont.)

- jQuery Event 제거
- off() : 모든 이벤트 핸들러를 제거
- off(“eventName”) : 특정 이벤트 핸들러를 제거
- one() : 이벤트를 한 번만 연결

3. JQUERY & AJAX

❖ jQuery AJAX

- \$.ajax() method의 형태

- `$.ajax(options);`
- `$.ajax(url, options);`
- Ajax가 성공했을 때 자동으로 success 실행
- success handler의 첫 번째 매개 변수는 Ajax가 성공했을 때 서버로 부터 받은 데이터
- 예제

3. JQUERY & AJAX

❖ jQuery AJAX (cont.)

● \$.ajax() method 옵션

- async : 동기, 비동기 지정 (boolean)
- complete : Ajax 완료 event handler 지정 (function)
- data : 요청 매개 변수 지정 (일반적으로 JSON)
- error : Ajax 실패 event handler 지정 (function)
- success : Ajax 성공 event handler 지정 (function)
- timeout : 만료시간 지정 (number)
- type : “GET” , “POST” 지정 (string)
- url : 호출 대상 URL 지정 (string)
- dataType : 결과 데이터가 json 혹은 jsonp일 경우 설정 (string)
- jsonp : JSONP방식일 경우 서버로 보내질 hash값에 대한 Key (string)

3. JQUERY & AJAX

❖ jQuery AJAX (cont.)

● 예제

- 서버 프로그램에 인자 전달 (한글처리 포함)
- 서버로부터 JSON 데이터 받아서 처리
- dataType : “json”

3. JQUERY & AJAX

❖ jQuery AJAX (cont.)

● AJAX Cross Domain 통신

- XMLHttpRequest 객체는 보안상 이유로 JavaScript source를 가져온 서버하고만 통신할 수 있음.
- 불특정 다수의 도메인 서버에 접근 : cross domain 통신
- 가장 쉬운 방법은 JSONP를 이용하는 방식.
(단, JSON을 이용하는 경우에만 가능)
- 예제

3. JQUERY & AJAX

❖ jQuery AJAX (cont.)

- jQuery AJAX 연습문제를 위한 setting

- MySQL설치 및 설정
- 서버 프로그램 설치 (Eclipse Project 제공)
- 제목에 대한 keyword를 입력 받아서
DB로부터 해당 책들의 세부정보를 JSON형식으로
return하는 프로그램
- 프로그램 설치 후 실행결과 확인

3. JQUERY & AJAX

❖ jQuery AJAX (cont.)

● jQuery AJAX Book Search Screenshot

Library Management System

자바

Book 전체 List

Book List

표지	Title	ISBN	Price	Author	상세보기	삭제
	IT EXPERT, 모바일 자바 프로그래밍 출판일 : 2002년 09월 페이지수 : 528 출판사 : 한빛미디어(주) 부록 : CD 1장	89-7914-206-4	23000	권기경, 박용우	상세보기	삭제
	Head First Java: 노회로를 자극하는 자바 학습법(개정판)	89-7914-371-0	28000	케이시 시에라, 버트 베이즈	상세보기	삭제
	모던 웹을 위한 HTML5 프로그래밍 : 한 권으로 끝내는 자바스 크립트 프레임워크 Backbone.js, Angular.js, Knockout.js 활용 출판일 : 2013년 08월 페이지수 : 720 출판사 : 한빛미디어(주) 부록 : 없음	978-89-6848-041-6	30000	윤인성	상세보기	삭제
	모던 웹을 위한 JavaScript + jQuery 입문(개정판) : 자바스크 립트에서 제이쿼리, 제이쿼리 모바일까지 한 권으로 끝낸다	978-89-6848-042-3	32000	윤인성	상세보기	삭제

3. JQUERY & AJAX

- ❖ jQuery AJAX (cont.)
- 영화진흥위원회 Open API를 이용한 영화정보 출력



Thanks for Listening