

과목명: 웹프로그래밍응용

교재: 모던웹을 위한 JavaScript JQuery 입문, 한빛미디어

Part2. jQuery / Ch13. 기본

Professor Seung-Hoon Choi

13.1 기본

□ jQuery 란?

- 모든 브라우저에서 동작하는 클라이언트 자바스크립트 라이브러리
- 2006년 John Resig이 개발
- 기능
 - 문서 객체 모델과 관련된 처리를 쉽게 구현
 - 일관된 이벤트 연결을 쉽게 구현
 - 시각적 효과를 쉽게 구현
 - Ajax 애플리케이션을 쉽게 개발
- 가장 많이 방문하는 웹사이트 10,000 개 중 jQuery 사용 비율
 - 67% (2013년 5월 기준, by BuildWith.com)
 - 22% (2014년 4월 9일 기준, by BuildWith.com)

13.2 다운로드

□ jQuery 사용 방법

- (1) 직접 다운로드 받아서 저장 후 사용
- (2) CDN 호스트 이용
 - CDN: Content Delivery Network
 - CDN 호스트로부터 jQuery를 요청하여 얻는 방식
 - 사용 예: 코드 13-1
 - `<script src=http://code.jquery.com/jquery-1.10.2.js> </script>`
 - 구글이나 마이크로소프트사도 jQuery CDN 호스트 역할을 제공함
- jQuery를 오프라인으로 사용할 경우에는 직접 다운로드 받아서 사용해야 함

13.2 다운로드

□ jQuery 다운로드

- <http://jquery.com> 접속
- Download jQuery 클릭
- [Download the uncompressed, development jQuery 1.11.0](#) 클릭
- 본인의 workspace의 jQuery 폴더에 저장

□ jQuery 종류

- uncompressed 버전: jquery.js (교재에서 사용함, 260KB)
- minified 버전: jquery.min.js (크기를 5분의 1로 zipping, 90KB)

13.2 다운로드

□ 다운로드 받은 jQuery 사용

- 코드 13-2
 - `<script src="jquery-1.10.2.js"> </script>`
- Visual Studio 제품군에서는 보조 기능 제공함
 - 예: "\$."을 입력하면 가능한 메소드 리스트와 메소드 프로토타입 등을 보여줌
 - 이를 위해서는 jquery-1.XX.X-**vsdoc**.js 파일이 필요함
 - jquery-1.XX.X.js 파일과 같은 폴더에 둘 것
 - <http://www.asp.net/ajaxlibrary/cdn.ashx> 의 화면 아래쪽에서 다운받을 수 있음

13.3 \$(document).ready()

□ jQuery를 사용한 모든 웹 페이지는 다음 코드로 시작함

– 코드 13-4

```
$(document).ready(function () {  
  
});
```

▪ 웹 문서가 준비되면 매개변수로 전달된 콜백 함수를 실행하라.

– window.onload = function() { } 과 비슷하지만, 여러 개의 콜백 함수를 등록할 수 있다.

▪ 코드 13-6

– 간단한 형태

```
$(function () {  
  
});
```

13.3 \$(document).ready()

□ \$ 의미

- jQuery 코드를 보면 다음 문장이 있다.
 - `jQuery = function() { ... }`
 - `window.jQuery = window.$ = jQuery;`
 - jQuery 식별자를 \$로 대체한 것임
 - \$ 대신에 jQuery 식별자를 사용해도 됨

13.4 기본 선택자

- jQuery 메소드의 가장 기본 형태

```
$('#h1').css('color', 'red');
```

\$ 선택자 메소드

13.4.1 전체 선택자

□ 전체 선택자

- HTML 페이지에 있는 모든 문서 객체를 선택하는 선택자
- '*' 를 사용함
- 코드 13-8
 - `$('*').css('color', 'red');`
 - 모든 문서 객체를 선택하여
 - 스타일 속성 color에 red 값을 설정하라.
 - `css()` 메소드: 스타일을 지정하는 메소드

13.4.2 태그 선택자

□ 태그 선택자

- 특정한 태그를 선택하는 선택자
- 코드 13-9, 13-10
 - `$('h1').css('color', 'orange');`

□ 여러 개의 태그를 동시에 선택하려면

- 코드 13-11
 - `$('h1, p').css('color', 'orange');`

13.4.3 아이디 선택자

□ 아이디 선택자

- 태그의 id 속성 값을 이용하여 문서 객체를 선택하는 선택자
- 코드 13-12, 13-13
 - `$('#target').css('color', 'orange');`
 - id 속성 값이 'target'인 문서 객체를 선택함
- 코드 13-14
 - `$(h1#target').css('color', 'orange');`
 - h1 태그 중에서 id 속성 값이 'target'인 문서 객체를 선택함
- id는 웹 페이지 내에서 고유해야 함

13.4.4 클래스 선택자

□ 클래스 선택자

- 태그의 class 속성 값을 이용하여 문서 객체를 선택하는 선택자
- 코드 13-15, 13-16
 - `$('.item').css('color', 'orange');`
 - class 속성 값이 'item'인 문서 객체를 선택함
 - `$(h1.item).css('color', 'orange');`
 - h1 태그 중에서 class 속성 값이 'item'인 문서 객체를 선택함
- 코드 13-17
 - `$('.item.select').css('color', 'orange');`
 - class 속성 값으로 'item'와 'select' 모두를 갖는 문서 객체를 선택함
- class 속성 값이 동일한 여러 태그가 존재할 수 있으며
- 하나의 태그는 여러 개의 class 속성 값을 가질 수 있다.

13.4.4 클래스 선택자

□ class 속성

- 하나의 웹 페이지에서
- class 속성 값이 동일한 여러 개의 태그가 존재할 수 있으며
- 하나의 태그는 여러 개의 class 속성 값을 가질 수 있다.

13.5 자손 선택자와 후손 선택자

- 문서 객체 모델은 트리 구조
 - 코드 13-18 -> 그림 13-13
 - 자손
 - 태그 바로 아래 있는 자식(들)
 - 후손
 - 태그 아래 있는 모든 문서 객체

13.5.1 자손 선택자

□ 자손 선택자

- 형태: 요소A > 요소B
- 코드 13-19
 - `$('body > *').css('color', red');`
 - body 태그의 모든 자식을 선택함
 - 요소 검사로 확인 가능함

13.5.2 후손 선택자

□ 후손 선택자

- 형태: 요소A 요소B
- 코드 13-20
 - `$('body *').css('color', red);`
 - body 태그의 아래에 있는 모든 후손을 선택함
 - 요소 검사로 확인 가능함

13.6 속성 선택자

□ 속성 선택자

- 속성 존재 여부 또는 속성의 값을 기준으로 태그를 선택하는 선택자

13.6 속성 선택자

□ 표 13-1

- 요소[속성]: 특정한 속성이 있는 태그를 선택
- 요소[속성=값]: 특정한 속성의 값이 특정 값과 같은 태그를 선택

13.6 속성 선택자

□ 표 13-1(계속)

- 요소[속성|=값]: 속성 안의 값이 특정 값을 **단어로 포함**하는 태그를 선택
 - 태그의 속성 값이 'kurien-blog'일 때 이것을 kurien과 blog, 두 단어로 인식함
 - 선택자가 '**img[name]="blog"**' 인 경우, 인 태그를 선택함
- 요소[속성~=값]: 속성 안의 값이 특정 값을 **단어로 포함**하는 태그를 선택
 - 태그의 속성 값이 'kurien-blog'일 때 이것을 'kurien-blog' 한 단어로 인식함
 - 선택자가 '**img[name~="blog"]**' 인 경우, 인 태그는 선택하지 않음

13.6 속성 선택자

□ 표 13-1(계속)

- 요소[속성^=값]: 속성 안의 값이 특정 값으로 시작하는 태그를 선택
 - 예: `img[src$="gif"]` => src의 속성 값이 gif로 끝나는 img 태그를 선택
- 요소[속성\$=값]: 속성 안의 값이 특정 값으로 끝나는 태그를 선택
 - 예: 선택자가 `'img[name*="log"]'` 인 경우, `` 인 태그를 선택함

13.6 속성 선택자

□ 코드 13-21, 13-22

- `$('input[type="text"]').val('Hello jQuery..!');`
 - type 속성의 값이 text인 input 태그를 선택하여
 - value 속성의 값으로 'Hello jQuery..!'을 설정하라.

13.7 필터 선택자

□ 필터 선택자

- 속성 선택자보다 편리하게 문서 객체를 선택할 수 있는 기능을 제공함
- ':' 기호를 사용함

13.7.1 입력 양식 필터 선택자

□ 입력 양식 필터 선택자

- 입력 양식 문서 객체를 간단하게 선택할 수 있게 해 줌
- 표 13-2
 - **input:button** - type이 button인 input 태그를 (또는 button 태그) 를 선택함
 - **input:checkbox** - type이 checkbox 인 input 태그를 선택
 - **input:file** - type이 file인 input 태그를 선택
 - **input:image** - type이 image인 input 태그를 선택 (이미지 submit 버튼)
 - **input:password** - type이 password인 input 태그를 선택
 - **input:radio** - type이 radio인 input 태그를 선택
 - **input:reset** - type이 reset인 input 태그를 선택
 - **input:submit** - type이 submit인 input 태그를 선택
 - **input:text** - type이 text인 input 태그를 선택

13.7.1 입력 양식 필터 선택자

□ 입력 양식 필터 선택자

– 표 13-3

- **요소:checked** - 체크되어 있는 입력 양식을 선택
- **요소:disabled** - 비활성화된 입력 양식을 선택
- **요소:enabled** - 활성화된 입력 양식을 선택
- **요소:focus** - 초점이 맞추어져 있는 입력 양식을 선택
- **요소:input** - 모든 입력 양식(input, textarea, select, button 태그)을 선택
- **요소:selected** - option 객체 중 선택된 태그를 선택

13.7.1 입력 양식 필터 선택자

❑ 코드 13-24

- setTimeout() 함수를 이용하여 5초 후에 사용자가 선택한 항목의 value를 출력하도록 한다.
- var value = \$('select > option:selected').val();
 - select 태그의 자손인 option 중에서 선택된 항목의 value 속성의 값을 value 변수에 할당해라.

❑ 실습

- 5초 후에, 사용자가 선택한 텍스트 박스에 노란색 배경색을 적용하는 웹 페이지를 작성하라.
 - <input type='text'>
 - \$('input:focus').css('background', 'yellow')

13.7.2 위치 필터 선택자

□ 위치 필터 선택자

- 위치와 관련된 필터 선택자
- 표 13-4
 - 요소:odd - 홀수 번째에 위치한 문서 객체를 선택
 - 요소:even - 짝수 번째에 위치한 문서 객체를 선택
 - 요소:first - 첫 번째에 위치한 문서 객체를 선택
 - 요소:last - 마지막에 위치한 문서 객체를 선택

13.7.2 위치 필터 선택자

□ 코드 13-26

- 표의 짝수 번째 행과 홀수 번째 행의 색을 다르게 함
 - `$(tr:odd).css('background', '#F9F9F9'); // 흰색`
 - `$(tr:even).css('background', '#9F9F9F'); // 회색`
 - `$(tr:first).css('background', '#000000').css('color', '#FFFFFF');`
- 행 번호는, 0 부터 시작함

13.7.3 함수 필터 선택자

□ 함수 필터 선택자(함수 형태의 필터 선택자)

– 표 13-5

- **요소:contains(문자열)** - 특정 문자열을 포함하는 모든 문서 객체 선택
- **요소:eq(n)** - n 번째 위치하는 문서 객체 선택(0이 첫번째 요소)
- **요소:gt(n)** - n번째 초과에 위치하는 모든 문서 객체 선택
- **요소:has(태그)** - 특정 태그가 있는 모든 문서 객체를 선택
 - 예: \$('div:has(p)') => <p> 태그를 갖는 모든 <div> 선택
- **요소:lt(n)** - n 번째 미만에 위치하는 모든 문서 객체 선택
- **요소:not(선택자)** - 선택자와 일치하지 않는 모든 문서 객체 선택
 - 예: \$('input:not(:empty)') => 비워있지 않은 모든 입력 객체 선택
- **요소:nth-child(3n+1)** - 3n+1번째에 위치하는 모든 문서 객체를 선택(1, 4, 7, ... 번째)

13.7.3 함수 필터 선택자

□ 코드 13-27

- 표의 행과 열마다 다른 색상으로 꾸미기
- 주의사항
- nth-child() 필터 사용 시에는
 - 표의 첫 행은 1부터 시작됨
 - 표의 첫 열은 1부터 시작됨

13.8 배열 관리

□ 배열 관리

- jQuery로 배열을 관리할 때는 **each() 메소드**를 사용함
- 두 가지 형태
 - 1) **\$.each(object, function(index, item) { })**
 - 자바스크립트 배열의 각 원소를 접근하거나
 - 객체의 각 속성을 접근할 때 사용함
 - 매개변수
 - object: 배열 객체
 - function: 원소를 처리하는 함수
 - 2) **\$(selector).each(function(index, item) { })**
 - selector로 선택된 모든 문서 객체의 각각을 접근할 때 사용함

13.8.1 자바스크립트 배열 관리

□ 자바스크립트 배열 관리

- 형식: `$.each(object, function(index, item) { });`
 - object: 배열 또는 객체
 - index: 배열의 인덱스 또는 객체의 키를 의미함
 - item: 인덱스나 키에 해당하는 값
- 주의: `forEach()` 메소드와 매개변수의 순서가 다름
 - `[].forEach(function(item, index) { });`

□ 코드 13-28, 13-29, 13-30

```
$.each(array, function(index, item) {  
    // item.link 접근  
    // item.name 접근  
    ... ..  
})
```

13.8.2 jQuery 배열 관리

□ jQuery 배열 관리

- jQuery 배열은, 선택자로 여러 개의 문서 객체를 선택할 때 생성됨
- 형식: `$(selector).each(function(index, item) { })`
 - 선택된 모든 문서 객체의 각각에 대해 함수 안의 내용을 실행함

13.8.2 jQuery 배열 관리

❑ 코드 13-31, 13-32

- `$('h1').addClass('high-light');`
 - `<h1>` 태그를 모두 골라서 `class` 속성의 값을 `'high-light'`로 지정하라.

❑ 코드 13-33, 13-34, 13-35

- `$('h1').each(function(index, item)) { })`
 - `<h1>` 태그 각각에 대해서 함수 안의 내용을 실행하라.

❑ 코드 13-36

- 원소 각각에 대해 실행할 함수 내부에서
 - `$(item)` 대신에 `$(this)`를 사용해도 됨

13.9 객체 확장

□ 객체 확장

- 두 객체를 합쳐서 확장할 때 사용함
- 형식: `$.extend(object, addedObject1, addedObject2, ...)`
 - object 객체에 나머지 객체의 속성을 추가하라

□ 코드 13-40

- `$.extend(object, { ... })`
 - object 객체에 뒤에 오는 객체를 추가함

13.10 jQuery 충돌 방지

□ jQuery 충돌 방지

- jQuery 이외의 자바스크립트 프레임워크를 함께 사용할 때 충돌이 발생할 수 있다.
- 예:
 - jQuery와 Prototype 프레임에서, 식별자 \$를 다 같이 사용함

□ \$.noConflict()

- 이 메소드 실행 후에는 , jQuery의 식별자 \$를 더 이상 사용할 수 없음
- jQuery 객체를 다른 변수에 저장하면 계속해서 간단하게 사용할 수 있다.
 - 코드 13-42