

3.10

- bxslider.com (슬라이더 사이트)
- 제이쿼리 다운로드 방법

① 제이쿼리.com

② 다운로드

③ (인터넷 연결시) 구글cdn -> 3.x snippet:에 있는거 복사

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
```

(인터넷 연결x시 다운로드)

Download the compressed, production jQuery 3.4.1 랑 uncompressed 두개
클릭 -> 우클릭 -> 다른이름저장 -> js폴더에 저장

- min.js는 압축한것

(볼 때는 압축안한거 보고 실제 서버에 올릴때는 압축한걸로 올림)

- \$.noConflict();
 - 다른 프레임웍과 충돌을 방지
 - jquery의 경우 \$를 jQuery로 변경해줘야함
 - 더이상 \$식별자를 사용할 수 없음
- \$(document).ready(function(){ });
 - 자바스크립트의 window.onload와 동일
 - 생략가능
- css 적용

\$("#box").css("background","#efefef");	(id)
\$("#p").css("color","orange");	(태그네임)
\$(".p").css("color","blue");	(클래스(p)태그)

\$("body p"); //자식 포함한 후손 모두 변경
\$("body > p"); //자식만 변경

- 메소드 체이닝 방식
 - 이어붙이기ex) \$(".sample").css("color","blue").css("background","green");
- 클릭시 함수 실행
\$("선택자").click(function(){
});
- 숨기기
\$(this).hide(); //display:none과 동일

- 선택자 옵션

p:odd 홀수번째 위치한 객체선택 (index는 0부터 시작)

p:even 짝수번째 위치한 객체선택

p:first 첫번째 객체 선택

p:last 마지막 객체 선택

- 텍스트 선택

ex)

- \$("#pTag").html("<p>aaa</p>"); //p태그가 적용되어 aaa만 출력됨
- \$("#pTag").text("<p>aaa</p>"); //<p></p>도 텍스트로 인식해서 전체출력

- 버튼선택

ex)

```
$("#button").click(function(){
    var elem = $("<p>새로운 p태그 입니다.</p>");
    elem.appendTo("body");

    $("<div/>",{
        "text" : "클릭하세요.",
        "click": function(){
            $(this).css("color","red");
        }
    }).appendTo("body");
});
```

버튼클릭시
1) p태그의 텍스트 생성
body에 붙임

2) div생성
text생성
클릭시 메서드 실행

body에 붙임

- delegate (selector, eventType, handler)

selector : 이벤트를 발생시킬 엘리먼트를 필터링할 선택자

eventType : click, mouseover 등과 같은 자바스크립트의 이벤트 타입이름

handler : 이벤트가 발생했을 때 실행 할 이벤트 핸들러 함수

delegate 메서드는 아직 문서를 만들지 않은 DOM 엘리먼트를 대상으로

이벤트 핸들러가 실행되게 된다.

해당 메서드로 이벤트 핸들러 함수를 연결하면 이벤트가 발생했을 때

대상 엘리먼트의 자손 엘리먼트 가운데 셀렉터 표현식과 일치하는 엘리먼트로

이벤트를 전달한다.

(jquery 1.4.2버전부터 가능)

ex)

//동적으로 생성된 p태그를 제어

```
$("#body").delegate("p","click",function(){
    $(this).hide();
});
```

//p태그를 클릭하면 함수실행

3.11

- html메서드 & text메서드

html()메서드와 text()메서드

- 지정된 엘리먼트(선택자)를 대상으로 값을 가져온다 - getter의 의미

html(값)메서드와 text(값)메서드

- 지정된 엘리먼트(선택자)를 대상으로 값을 출력, 변경한다 - setter의 의미

- html : 태그를 포함한 문장을 가져온다
text : 태그를 포함하지 않는 문자만 가져온다

ex)

- 1) var vtext = \$("p").text(); //getter
\$("#textResult").text(vtext); //setter
//p태그 전체를 가져옴
- 2) var vHTML = \$("p").html();
\$("#htmlResult").html(vHTML);
//p태그들 중 첫번째만 가져옴
- 3) var vHTML = "";
\$("p").each(function(){ //each는 for문과 유사
vHTML += \$(this).html();
});
\$("#htmlResult").html(vHTML);
//html로 p태그 전체 가져옴(each 사용)

- 선택자

\$("#p[title]")

\$("#input[type=x]") //type은 따옴표 제외하고 type=x의 형태로!

= (속성과 값이 같음)
!= (특정값과 같음)....?
~= (특정값을 단어로 시작)
^= (특정값으로 시작)
\$= (특정값으로 끝남)
*= (특정값을 포함)

- 내용필터

- 포함하고 있는 문자나 선택자에 의해서 선택

- contains() 문자열 포함하는지

- empty() 비었는지

- parent() 부모인지

- has(selector) 태그 포함하는지

- * 문자열 있는지 검색은 contains()

- 태그가 있는지 검색은 has()

- 내용 보이게 : `css("display","block");` 또는 아무것도x

- ” 안보이게 : `css("display","none");`

- 슬라이더 (slider)

- `slideUp("slow");`

- `slideDown(5000);`

- `slideToggle();`

- 괄호 안에는 슬라이드 속도

- 이미지 슬라이드 플러그인

- mouseover & mouseout

- 호버링(버블링) 현상이 나타난다

- 경계선 마다 계속 호출이 된다

- 자식요소위에 마우스 오버해도 이벤트가 발생

- mouseenter & mouseleave

- 호버링(버블링) 현상이 없다

- 한번만 호출된다

- 자신에게 마우스 오버했을때만 이벤트가 발생

- 이클립스 자동완성

- help - eclipse marketplace - tern검색 - 설치

- 프로젝트 우클릭 - configure - convert to Tern - jquery어쩌구 4개랑 크롬extension
체크

3.12

- 순서 선택자
- eq(index) 메서드는 해당하는 index번호를 선택하는 객체
eq(0)은 first와 동일

ex)

- 1) \$("li:eq(0)") 바로 인덱스 선택
- 2) \$("div:odd").eq(0) 붙여서 인덱스 선택
- 3) \$("div").filter(":even").eq(0) 다 분리해서 선택

- end메서드
- 선택을 한단계 앞으로 돌리는 역할을 한다
기존 선택에 새로운 필터링을 할 수 있도록 한다

ex)

```
$("div:even").eq(0).css("background","red")  
.end().eq(1).css("background","blue")  
.end().eq(2).css("background","#efeeff");  
(메서드 체이닝 방식)
```

- 아닌것을 선택
ex) \$("p").not(".intro")
- form 선택자 (form selector)

val()메서드 : 입력하거나 선택된 값을 가져오기
radio 선택된 값 가져오기 : \$("radio:checked").val();
select 선택한 값 (option) : \$("#work :selected").val();
(select만 스페이스하고 :selected!!)
radio랑 select도 사실 val()로 바로 가져올 수 있음

- alert으로 찍을땐 br말고 \n으로
ex) var str += \$("#id").val() + "\n";

- 태그 생성하기
\$("<태그>",{속성:"값"});

ex)

```
var img = $("<img/>",{  
    "src":"../img/"+txt+".jpg",  
    "width":"100px",  
    "height":"100px"  
});
```

- 태그에 붙이기

- 태그 뒤에 붙이기

```
$("#p").append("<div>append로 붙인 태그</div>");
```

(p태그 안쪽으로 해서 뒤에 붙는다)

```
$("#p").after("<div>after로 붙인 태그</div>");
```

(감싸고 있는 p태그 밖으로 해서 뒤에 붙는다)

- 태그 앞에 붙이기

```
$("#p").prepend("<div>prepend로 붙인 태그</div>");
```

(p태그 안쪽으로 해서 앞에 붙는다)

```
$("#p").before("<div>before로 붙인 태그</div>");
```

(감싸고 있는 p태그 밖으로 앞에 붙는다)

- `$("#body").append($("#p").clone());` (선택자를 앞에 선택)
`$("#p").clone().appendTo("body");` (" 뒤에 선택)

- `$("#img").text();` 텍스트 가져옴
`$("#img").attr("title");` //속성값(타이틀) 가져옴

- 삭제하는 메서드
`empty();` div는 지우지 않고 안에 자식들만 지운다 (태그, 텍스트포함)
`remove();` div 자체와 안에 자식들을 모두 지운다
- 삭제하면서 붙였을 경우,
`remove`는 이벤트도 삭제
`detach`는 이벤트가 유지

3.13

- 이벤트 발생
\$("태그네임").on("이벤트",function(){실행문});

- 체크된 값 가져오기

1) 처음 체크된 값만 가져오기

- 체크박스

```
var chTxt = $("input[type=checkbox]:checked").val();
```

- 라디오 버튼

```
result += $("label[for="+$("input[type=radio]:checked").attr("id")+"]').text();
```

(라벨이 = 라디오 체크된거의 속성이 아이디인것의 값인 텍스트값)

(라벨과 ID값은 동일해야함!!)

2) 체크된 모든 값 가져오기

- 체크박스

```
$.each($("input[type=checkbox]:checked"),function(i){
```

```
    if(i > 0) result += ","; //첫번째는 안붙고 시작, 마지막은 안붙고 끝남
```

```
    result += $(this).val();
```

```
});
```

- 클래스 추가삭제

1) 클래스 추가 (addClass)

```
$("태그네임").on("click", function(){
```

```
    $("p").addClass("pClass"); //p태그에 클래스 붙임
```

```
});
```

2) 클래스 삭제 (removeClass)

//클래스 값만 사라짐

```
$("태그네임").click(function(){
```

```
    $("p").removeClass("pClass");
```

```
});
```

3) 클래스 추가삭제 (toggleClass)

//한번 클릭시 실행, 재클릭시 실행취소

```
$("태그네임").click(function(){
```

```
    $("p").toggleClass("pClass");
```

```
});
```

4) 클래스 여부 확인 (hasClass)

//있으면 true, 없으면 false리턴

```
$("#has").click(function(){
```

```
    alert($("p").hasClass("pClass"));
```

```
});
```

- 속성에 문자열 잘라서 넣기

ex) title속성을 이미지명과 동일하게 넣기

```

$("img").each(function(){
    var srcTitle = "";
    srcTitle = $(this).attr("src");    //전체 src주소 넣기

    var startIdx = $(this).attr("src").lastIndexOf("/");    //시작인덱스 번호
    var endIdx = $(this).attr("src").lastIndexOf(".");    //끝인덱스 번호

    $(this).attr("title",srcTitle.substring(startIdx+1, endIdx));    //src잘라서 속성추가
});

```

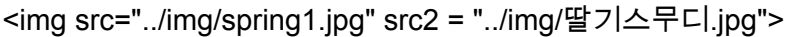
- xy좌표

```

$(document).mousemove(function(event){
    event.pageX        (페이지 좌표 x)
    event.pageY        (페이지 좌표 y)
    event.screenX      (스크린 좌표 x)
    event.screenY      (스크린 좌표 y)
});

```

- 이미지 변경

1. 이미지에 src를 두개 넣어놓는다.

2. ex) 이미지 마우스 오버시 src를 src2로 바꾼다.

```

$("img").mouseover(function(){
    $(this).attr("src",$(this).attr("src2"));
});

```
3. 임시 속성에 원래이미지(src)를 담아준다.

```

$("img").attr("temp",function(){
    return $(this).attr("src")
});

```
4. 마우스 아웃시 원래 이미지로 변경한다.

```

$("img").mouseout(function(){
    $(this).attr("src",$(this).attr("temp"));
});

```


- 마우스 오버시 이미지 확대

1. 마우스 오버시 동일 이미지 담은 p태그를 body에 생성

```
$("#태그네임").mouseover(function(event){  
    $("#body").append("<p id='preview'>  
        <img src='"+$(this).attr("src")+"'/></p>");  
    });
```
2. 마우스 무브시 이미지 보임

```
$(document).on("mousemove", ".thumbnail", function(event){  
    $("#preview").css("top", (event.pageY+10) + "px")  
        .css("left", (event.pageX+30) + "px").fadeIn("fast");  
    });
```
3. 마우스 아웃시 사라짐

```
$(document).on("mouseout", ".thumbnail", function(){  
    $("#preview").remove();  
    })
```

- 더블클릭 = dblclick

3.16

- width height
 - width()
height()
(border제외 padding제외)
 - innerWidth()
innerHeight()
(border제외 padding포함)
 - outerWidth()
outerHeight()
(border포함 padding포함)
 - innerWidth와 outerWidth는 margin과는 상관없고 padding값은 적용됨
- width height 값 적용
 - 1) .width(100).height(100)
 - 2) .css("width","300px").css("height","300px")
 - 1과 2는 동일

- .on()

- 선택한 요소에 어떤 타입의 이벤트라도 연결가능
- 하나의 이벤트 핸들러에 여러개의 이벤트를 동시에 연결 가능
ex) `.on("click mouseover dblclick",function(){ });`
- 선택한 요소에 여러개의 이벤트 핸들러와 여러개의 이벤트를 연결 가능
- on 하나에 이벤트별로 내용 다르게 설정
ex) `.on({
 click : function(){},
 dblclick : function(){}
});`

- 정적함수 & 동적함수

- 1) 태그에 이미지 바인딩 후 동적으로 태그 생성시
기존의 태그에만 적용
(동적으로 생성된 태그는 적용x)
- 2) 동적함수(delegate()) 사용하면 동적으로 태그 생성시
기존 + 뉴 태그 모두 적용

ex)

- 1) `$("#h1").on("click", function(){
 $(this).css("background-color","pink");
 $("#<h1>새로운 제목태그 입니다.</h1>").insertAfter("h1");
});`
- 2) `.delegate("h1", "click", function(){
 $(this).css("background-color","green");
 $("#<h1>새로운 제목태그 입니다.</h1>").insertBefore("h1");
});`

- key이벤트

- keypress(글자가 입력될때)
(입력할 수 있는 키보드(영어, 숫자 등)가 눌렸을 때에만 이벤트 발생)
keydown(키를 눌렀을때)
keyup(키를 눌렀다 뗄 때)
- keypress는 대소문자 구분, keydown은 대소문자 구분x
- focusin : 커서가 필드 안에 있을때
focusout : 필드 밖으로 나왔을때

3.17~19

- 대검찰청 개인 프로젝트
 - 슬라이드
 - 롤링배너