

# JQuery 이벤트 처리

출처 : Do it! 자바스크립트+제이쿼리입문(이지스퍼블리싱, 정인용)  
모던웹을 위한 javascript+jQuery입문(한빛미디어, 윤인성)  
javascript+jQuery입문+실전북(앤써북, 김규태)  
속깊은 jQuery(루비페이퍼, 원형섭)  
<http://tcpschool.com/>

## 4. jQuery 이벤트 처리

### ❖ jQuery 이벤트

- 이벤트란? 사용자가 웹 페이지에서 하는 동작, 즉 마우스를 움직이거나 요소 클릭, 입력창에 값 입력 등 다양한 동작을 이벤트라고 함
- jQuery는 자체적으로 추상화된 jQuery 이벤트 객체를 제공
- jQuery 이벤트 객체는 각 브라우저의 서로 다른 이벤트 객체를 추상화하여 제공하므로, 어느 브라우저에서나 동일한 코드로 이벤트 처리가 가능 - 크로스 브라우저 지원
- jQuery의 이벤트에는 기존 자바스크립트의 이벤트가 모두 존재

### ❖ jQuery 이벤트 핸들러

- 이벤트 발생시 특정 요소에 지정된 이벤트의 발생을 확인하기 위해서 이벤트 핸들러를 사용(연결)
- 특정 요소에 지정한 타입의 이벤트가 발생하면 웹 브라우저는 그 요소에 연결된 이벤트 핸들러를 실행



## 4. jQuery 이벤트 처리

### ❖ jQuery 이벤트 객체

- 이벤트 핸들러는 이벤트 객체를 매개변수로 전달 받을 수 있음
- 이렇게 전달된 이벤트 객체를 이용하여 프로그램을 제어하거나, 이벤트의 기본 동작을 제어할 수 있음

```
$("#mouseArea").bind("mouseenter mouseleave", function (e) {  
    if (e.type == 'mouseenter') {  
        countEnter++;  
        $("#result1").text(countEnter + " MouseEnter");  
    }  
});
```

```
$(document).on("contextmenu", function(e) {  
    e.preventDefault();  
});
```

### ❖ jQuery 이벤트 설정

- jQuery는 이벤트 연결을 위해 메서드를 사용
- 연결 메서드를 호출할 때 이벤트 핸들러 함수를 파라미터로 전달하며, 전달된 함수는 이벤트에 연결되고, 지정한 엘리먼트에서 이벤트 상황이 발생하면, 연결된 함수가 호출됨



## 4. jQuery 이벤트 처리

### ❖ 이벤트 연결

- jQuery 이벤트 연결(binding) 메서드는 하나의 이벤트만 등록할 수 있는 단독 이벤트 메소드와, 하나 이상의 이벤트를 설정할 수 있는 그룹 이벤트 메소드가 있음

- 단독 이벤트 메서드 : `$( "#btn" ).click(function() {실행할 코드 });`
- 그룹 이벤트 메서드 : `$("#btn").bind("click", function() {실행할 코드 });` //jQuery1.7부터는 미 권장  
`$("#btn").on("click", function() {실행할 코드 });`  
`$(ul).on("click", "li", function() {실행할 코드 });` //이벤트 위임

### ■ 단독 이벤트 종류

마우스 이벤트	
메소드	설명
<code>click()</code>	마우스를 클릭했을 때
<code>dblclick()</code>	마우스를 더블 클릭했을 때
<code>mousedown()</code>	마우스 버튼을 눌렀을 때
<code>mouseenter()</code>	마우스 커서가 해당 태그로 들어갔을 때
<code>mouseleave()</code>	마우스 커서가 해당 태그에서 나갔을 때
<code>mousemove()</code>	마우스가 움직일 때
<code>mouseup()</code>	마우스 버튼을 땄을 때



## 4. jQuery 이벤트 처리

### ❖ 이벤트 연결

#### ■ 단독 이벤트 종류

##### 키보드 이벤트

메소드	설명
keydown()	키보드 키를 눌렀을 때
keypress()	키가 입력되었을 때
keyup()	키보드 키를 떼었을 때

##### 입력 양식 이벤트

메소드	설명
blur()	입력 양식에 값 입력을 종료할 때
change()	입력 양식의 값이 변경될 때
focus()	입력 양식에 값 입력을 시작할 때
select()	type 속성이 select인 입력 양식의 목록에서 값을 선택했을 때
submit()	type 속성이 submit인 입력 양식을 클릭했을 때

##### 웹 브라우저 이벤트

메소드	설명
resize()	웹 브라우저의 크기를 변경할 때
scroll()	웹 브라우저를 스크롤할 때



## 4. jQuery 이벤트 처리

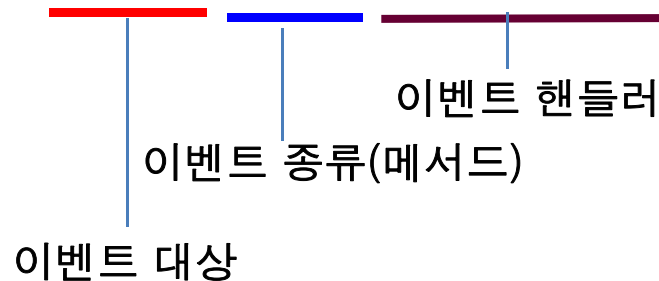
### ❖ 이벤트 설정 – 이벤트 핸들러 등록

- **단독 이벤트 메서드**는 한 동작에 대한 이벤트를 설정할 때 사용하는 메서드로, "요소를 클릭했을 때", "요소를 더블 클릭했을 때", "요소에 마우스를 올렸을 때" 등 하나의 이벤트에만 대응하여 이벤트 핸들러의 실행문이 실행됨
- 내부적으로 단독 메서드는 모두 on() 메서드를 호출함

```
$("#inquiry").click(function() { });  
→ $("#inquiry").on("click", function() { });
```

- 이벤트 사용의 기본 형식

`$("#btn").click(function() { ....});`

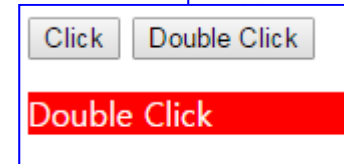
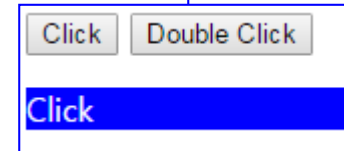
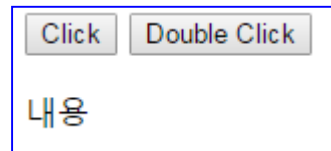


## 4. jQuery 이벤트 처리

### ❖ 마우스 이벤트 메소드 – click() / dblclick()

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4 <meta charset="utf-8">
5 <title>jQuery Event</title>
6 <script src="js/jquery-3.1.1.min.js"></script>
7 <script>
8 $(document).ready(function () {
9     $("#btn1").click(function() { // <button id="btn1">을 클릭했을 때...
10         var $txt = $(this).text();
11         $("p").text($txt).css({"color":"white", "background-color":"blue"});
12     });
13
14     $("#btn2").dblclick(function() { // <button id="btn2">을 더블클릭했을 때...
15         var $txt = $(this).text();
16         $("p").text($txt).css({"color":"white", "background-color":"red"});
17     });
18 });
19 </script>
20 </head>
21 <body>
22 <button id="btn1">Click</button>
23 <button id="btn2">Double Click</button>
24
25 <p>내용</p>
26 </body>
27 </html>
```

jquery03.html



## 4. jQuery 이벤트 처리

### ❖ 마우스 이벤트 메소드 – mouseover() / mouseout() / hover()

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>jQuery Event</title>
6   <script src="js/jquery-3.1.1.min.js"></script>
7   <script>
8     $(document).ready(function () {
9       $("#btn1")
10        .mouseover(function(){
11          var $txt = $(this).text();
12          $("p").text($txt).css({"color":"white", "background-color":"blue"});
13        })
14        .mouseout(function(){
15          var $txt = $(this).text();
16          $("p").text($txt).css({"color":"white", "background-color":"red"});
17        });
18
19      $("#btn2").hover(function(){
20        $("p").text("hover-in").css("background-color", "green");
21      }, function(){
22        $("p").text("hover-out").css("background-color", "pink");
23      });
24    });
25  </script>
26 </head>
27 <body>
28   <button id="btn1">mouseover/out</button>
29   <button id="btn2">hover</button>
30
31   <p>내용</p>
32 </body>
33 </html>
```

Jquery04.html

mouseover/out

hover

mouseover/out





## 4. jQuery 이벤트 처리

### ❖ 마우스 이벤트 메소드 – mouseenter() / mouseleave()

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <style>
8       #nations > thead > tr { background-color:purple; color:yellow; }
9       #nations > thead > tr > th { padding :0px 1px 0px 0px; width:150px; }
10      #nations > tbody > tr > td { border-bottom: solid 1px gray; }
11      .sel { background-color : aqua; }
12    </style>
13    <script>
14      $(document).ready(function () {
15        $("#nations > tbody > tr").mouseenter(function() {
16          $(this).addClass("sel");
17        }).mouseleave(function() {
18          $(this).removeClass("sel");
19        });
20      });
21    </script>
22  </head>
23  <body>
24    <table id="nations">
25      <thead>
26        <tr><th>번호</th><th>국가명</th><th>수도</th></tr>
27      </thead>
28      <tbody>
29        <tr><td>1</td><td>미국</td><td>워싱턴DC</td></tr>
30        <tr><td>2</td><td>프랑스</td><td>파리</td></tr>
31        <tr><td>3</td><td>영국</td><td>런던</td></tr>
32        <tr><td>4</td><td>중국</td><td>베이징</td></tr>
33        <tr><td>5</td><td>태국</td><td>방콕</td></tr>
34        <tr><td>6</td><td>모로코</td><td>라바트</td></tr>
35        <tr><td>7</td><td>라오스</td><td>비엔티안</td></tr>
36      </tbody>
37    </table>
38  </body>
39 </html>
```

Jquery04\_2.html

```
<table id="nations">
  <thead>...</thead>
  <tbody>
    <tr class="...">...</tr>
    <tr class="...">...</tr>
    <tr class="sel">
      <td>3</td>
      <td>영국</td>
      <td>런던</td>
    </tr>
    <tr class="...">...</tr>
    <tr class="...">...</tr>
    <tr class="...">...</tr>
    <tr class="...">...</tr>
  </tbody>
</table>
```

mouseover()/mouseout()은 버블링이 발생하지만  
mouseenter()/mouseleave() 발생하지 않기 때문에 사용빈도가 높음

번호	국가명	수도
1	미국	워싱턴DC
2	프랑스	파리
3	영국	런던
4	중국	베이징
5	태국	방콕
6	모로코	라바트
7	라오스	비엔티안

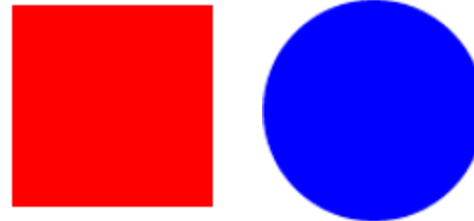


## 4. jQuery 이벤트 처리

### ❖ 마우스 이벤트 메소드 – hover()

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>jQuery Event</title>
6   <script src="js/jquery-3.1.1.min.js"> </script>
7   <style>
8     #box {
9       width: 100px; height: 100px;
10      background-color: red;
11    }
12    #box.hover{
13      background-color: blue;
14      border-radius: 50px;
15    }
16  </style>
17  <script>
18    $(document).ready(function () {
19      $('#box').hover(function () { // 마우스가 올라갈 경우 hover클래스 추가
20        $('#box').addClass('hover');
21      }, function () { // 마우스가 올라간 상태에서 내려가면 hover클래스 제거
22        $('#box').removeClass('hover');
23      });
24    });
25  </script>
26 </head>
27 <body>
28   <div id="box"></div>
29 </body>
30 </html>
```

Jquery05.html



## 4. jQuery 이벤트 처리

### ❖ 키보드 이벤트 메소드 – keydown() / keyup() / keypress()

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <script>
8       $(document).ready(function () {
9         $("#user_id")
10          .keydown(function(e){ //키보드 자판을 눌렀을 때
11            $(this).css("background-color","skyblue");
12            $("p").text(e.keyCode); //고유 키 코드 값을 출력
13          })
14          .keyup(function(){ //자판을 눌렀다 떼었을 때
15            $(this).css("background-color","white");
16          });
17      });
18    </script>
19  </head>
20  <body>
21    <div>
22      <input type="text" name="user_id" id="user_id" />
23    </div>
24
25    <h3> key-code </h3>
26    <p></p>
27  </body>
28 </html>
```

Jquery06.html

key-code

65

## 4. jQuery 이벤트 처리

### ❖ 키보드 이벤트 메소드 – keyup()

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <style>
8       #nations > thead > tr { background-color:purple; color:yellow; }
9       #nations > thead > tr > th { padding :0px 1px 0px 0px; width:150px; }
10      #nations > tbody > tr > td { border-bottom: solid 1px gray; }
11    </style>
12    <script>
13      $(document).ready(function () {
14
15      </script>
16    </head>
17    <body>
18      <div style="width:460px;">
19        국가명 : <input type="text" id="keyword" /><br />
20      </div>
21      <table id="nations">
22        <thead>
23          <tr><th>번호</th><th>국가명</th><th>수도</th></tr>
24        </thead>
25        <tbody>
26          <tr><td>1</td><td>미국</td><td>워싱턴DC</td></tr>
27          <tr><td>2</td><td>프랑스</td><td>파리</td></tr>
28          <tr><td>3</td><td>영국</td><td>런던</td></tr>
29          <tr><td>4</td><td>중국</td><td>베이징</td></tr>
30          <tr><td>5</td><td>태국</td><td>방콕</td></tr>
31          <tr><td>6</td><td>모로코</td><td>라바트</td></tr>
32          <tr><td>7</td><td>라오스</td><td>비엔티안</td></tr>
33          <tr><td>8</td><td>베트남</td><td>하노이</td></tr>
34          <tr><td>9</td><td>피지</td><td>수바</td></tr>
35          <tr><td>10</td><td>솔로몬 제도</td><td>호니아라</td></tr>
36          <tr><td>11</td><td>자메이카</td><td>킹스턴</td></tr>
37          <tr><td>12</td><td>나미비아</td><td>빈트후크</td></tr>
38          <tr><td>13</td><td>동티모르</td><td>딜리</td></tr>
39          <tr><td>14</td><td>멕시코</td><td>멕시코시티</td></tr>
40          <tr><td>15</td><td>베네수엘라</td><td>카라카스</td></tr>
41          <tr><td>16</td><td>서사모아</td><td>아피아</td></tr>
42        </tbody>
43      </table>
44    </body>
45  </html>
```

Jquery06\_2.html

국가명 :

번호	국가명	수도
1	미국	워싱턴DC
2	프랑스	파리
3	영국	런던
4	중국	베이징
5	태국	방콕
6	모로코	라바트
7	라오스	비엔티안
8	베트남	하노이
9	피지	수바
10	솔로몬 제도	호니아라
11	자메이카	킹스턴
12	나미비아	빈트후크
13	동티모르	딜리
14	멕시코	멕시코시티
15	베네수엘라	카라카스
16	서사모아	아피아

## 4. jQuery 이벤트 처리

### ❖ 키보드 이벤트 메소드 – keyup()

```
12 <script>
13     $(document).ready(function () {
14         //국가명 입력창(#keyword)에 keyup() 이벤트 정의
15         $("#keyword").keyup(function () {
16             //사용자가 입력한 값을 inputStr에 저장
17             var inputStr = $(this).val();
18             //테이블의 모든 행에 숨겨지도록 hide() 메서드 처리
19             $("#nations>tbody>tr").hide();
20             //테이블 행의 2번째 td에서 특정 텍스트를 포함한 요소를 선택해주는 선택자(contains() 메서드)
21             // 즉, 입력한 국가명을 포함한 요소를 선택
22             var temp = "#nations>tbody>tr>td:nth-child(3n+2):contains('" + inputStr + "')";
23             console.log(temp);
24             //temp를 td의 부모 요소인 tr 요소에 표시
25             $(temp).parent().show();
26         });
27     });
28 </script>
```

Jquery06\_2.html-스크립트

국가명 :

번호	국가명	수도
1	미국	워싱턴DC
12	나미비아	빈트후크

국가명 :

번호	국가명	수도
1	미국	워싱턴DC
12	나미비아	빈트후크

Elements Console Sources Network Timeline

top Preserve log

#nations>tbody>tr>td:nth-child(3n+2):contains(' ')  
#nations>tbody>tr>td:nth-child(3n+2):contains('미')



## 4. jQuery 이벤트 처리

### ❖ 키보드 이벤트 메소드 – keyup(), keypress(), keydown()

- 한글과 숫자만 입력하도록 이벤트 처리

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <style>
8       .focused { border: solid 3px gray; background-color: aqua; }
9     </style>
10    <script>
11      $(document).ready(function () {
12
13      });
14    </script>
15  </head>
16  <body>
17    한글입력 : <input type="text" class="korean effect" /> <br />
18    숫자입력  : <input type="text" class="number effect" /> <br />
19  </body>
20 </html>
```

Jquery06\_3.html





## 4. jQuery 이벤트 처리

### ❖ 키보드 이벤트 메소드 – keyup(), keypress(), keydown()

- 한글과 숫자만 입력하도록 이벤트 처리

```
10 <script>
11 $(document).ready(function () {
12     //입력필드 focus, blur 이벤트 처리
13     $("input.effect").on("focus blur", function(e) {
14         $(this).toggleClass("focused"); //focused를 토글 처리
15     });
16
17     // 한글만 입력하도록 이벤트 처리
18     $("input.korean").on("keyup keypress", function(e) {
19         //한글 패턴에 부합하는 모든 문자열을 찾기 위해 정규식 사용
20         var pat = /^[^ㄱ-힣\s\,\.\\"'\"]/g;
21         var v = $(this).val();
22         //한글 패턴에 맞지 않는 문자를 입력하면 ""으로 치환
23         $(this).val(v.replace(pat, ""));
24     });
25
26     //숫자만 입력하도록 이벤트 처리
27     $("input.number").keydown(function(e) {
28         //입력 가능한 키-코드 패턴을 저장
29         var chrs = [48,49,50,51,52,53,54,55,56,57]; //키보드 위쪽 0~9
30         chrs = chrs.concat([96,97,98,99,100,101,102,103,104,105]); //숫자 키패드
31         chrs = chrs.concat([37,38,39,40]); //방향키(커서키)
32         chrs = chrs.concat([13,8,9]); //엔터, 백스페이스, 탭 키
33         chrs = chrs.concat([45,46,36,35]); //ins, del, home, end
34         chrs = chrs.concat([16,17,18,19]); //shift, ctrl, alt, 소수점(.)
35
36         //사용자가 입력한 키와 등록된 키-코드 패턴을 비교
37         var k = e.keyCode; //사용자가 입력한 키코드는 이벤트객체(e)를 통해서 받음
38
39         var isValid = false;
40         //입력한 키값과 등록된 키-코드 패턴을 비교
41         for(var i=0; i < chrs.length; i++) {
42             if (k == chrs[i]) {
43                 isValid = true;
44                 break;
45             }
46         }
47         /* 입력한 키와 등록된 키-코드가 틀리면
48         유효하지 않은 문자열이므로 기본 이벤트 실행을 중지 */
49         if (isValid == false) {
50             e.preventDefault();
51             alert("숫자만 입력가능합니다.");
52         }
53     });
54 }
55 </script>
```

Jquery06\_3.html-스크립트



## 4. jQuery 이벤트 처리

### ❖ change() 이벤트 메소드

- change() 이벤트 메소드는 선택한 입력 요소의 값이 바뀌고 포커스가 다른 요소로 이동되었을 때 이벤트가 발생

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"> </script>
7   </script>
8     $(document).ready(function () {
9       $('#site').change(function(e){
10         $('#p').text($(this).val());
11       });
12     });
13   </script>
14 </head>
15 <body>
16   <div>
17     <select id="site">
18       <option value="">선택</option>
19       <option value='www.naver.com'>네이버</option>
20       <option value='www.google.com'>구글</option>
21       <option value='www.daum.net'>다음</option>
22     </select>
23   </div>
24
25   <h3> select-url </h3>
26   <p></p>
27
28 </body>
29 </html>
```

Jquery07.html

네이버 ▾

select-url

www.naver.com





## 4. jQuery 이벤트 처리

### ❖ index() 인덱스 반환 메소드

- 이벤트가 발생한 요소의 인덱스 값을 반환

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4 <meta charset="utf-8">
5 <title>jQuery Event</title>
6 <script src="js/jquery-3.1.1.min.js"></script>
7 </script>
8 $(document).ready(function () {
9     $("#menu a").mouseover(function() { //<a>에 마우스를 올렸을 때...
10         $("#menu a").css("background", "none"); //<a>에 배경 속성을 지웁니다.
11
12         //이벤트 발생한 리스트 <a>만 배경이 노란색으로
13         $(this).css("background", "yellow");
14
15         //이벤트 발생한 리스트의 인덱스값을 가져와서 tetx 표시
16         var num=$("#menu a").index(this);
17         $("#p").text(num);
18     });
19 });
20 </script>
21 </head>
22 <body>
23 <ul id="menu">
24 <li><a href="#">내용1</a></li>
25 <li><a href="#">내용2</a></li>
26 <li><a href="#">내용3</a></li>
27 <li><a href="#">내용4</a></li>
28 </ul>
29
30 <h3> index </h3>
31 <p></p>
32 </body>
33 </html>
```

Jquery08.html

- [내용1](#)
- [내용2](#)
- [내용3](#)
- [내용4](#)

index

0



## 4. jQuery 이벤트 처리

- ❖ 그룹 이벤트 등록 메소드 – on() 메서드를 이용한 이벤트(핸들러)등록
  - jQuery 이벤트를 연결하는 가장 기본적인 방법

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>jQuery Event</title>
6   <script src="./js/jquery-3.2.1.min.js"></script>
7 </script>
8 $(document).ready(function () {
9   $("input:disabled").css("border", "dashed 2px gray");
10
11   $("#inquiry").on("click", function() {
12     var v = $("#food option:selected").val();
13     if (v == "") {
14       alert("음식 분류를 선택하셔야 합니다.");
15     } else {
16       $("#result").val(v).css("text-align", "center");
17     }
18   });
19 });
20 </script>
21 </head>
22 <body>
23   음식 분류 :
24   <select id="food">
25     <option value="" selected>-----음식 분류를 선택하세요-----</option>
26     <option value="한식">한식</option>
27     <option value="중식">중식</option>
28     <option value="양식">양식</option>
29     <option value="일식">일식</option>
30     <option value="분식">분식</option>
31   </select>
32
33   <button id="inquiry">조회</button>
34
35   <hr>
36   선택 결과 : <input type="text" id="result" value="" disabled />
37 </body>
38 </html>
```

음식 분류 :

선택 결과 :

## 4. jQuery 이벤트 처리

### ❖ 그룹 이벤트 등록 메소드 – on()

- 그룹 이벤트 등록 메소드를 사용하면 2개 이상의 이벤트를 등록 할 수 있음

Jquery10.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <script>
8       $(document).ready(function () {
9         $("div#cont").on({
10          "mouseover focus" : function(){
11            $(this).css("color","red");
12          },
13          "mouseout blur" : function(){
14            $(this).css("color","blue");
15          }
16        });
17      });
18    </script>
19  </head>
20  <body>
21    <div>
22      <div id="cont"><h1> 내용 </h1></div>
23    </div>
24  </body>
25 </html>
```

내용



## 4. jQuery 이벤트 처리

### ❖ 그룹 이벤트 등록 메소드 – on()

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="../js/jquery-3.2.1.min.js"></script>
7 </head>
8 <body>
9     <div id="mouseArea">
10         <span>mouseArea</span>
11     </div>
12     <div id="result1"></div>
13     <div id="result2"></div>
14 </body>
15 </html>
16
17 Jquery11.html
18
19 <script>
20     var countEnter = 0;
21     var countLeave = 0;
22     $(document).ready(function () {
23         $("#mouseArea").on("mouseenter mouseleave", function (e) {
24             if (e.type == 'mouseenter') {
25                 countEnter++;
26                 $("#result1").text(countEnter + " MouseEnter");
27             }
28             else {
29                 countLeave++;
30                 $("#result2").text(countLeave + " MouseLeave ");
31             }
32         });
33     });
34 </script>
35 </head>
36 <body>
37     <div id="mouseArea"><span> mouseArea </span></div>
38
39     <div><span id="result1"></span></div>
40     <div><span id="result2"></span></div>
41 </body>
42 </html>
```

mouseArea

2 MouseEnter  
2 MouseLeave



## 4. jQuery 이벤트 처리

### ❖ 그룹 이벤트 등록 메소드 – on()

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      #box { width:200px; height:200px; border:2px solid blue;
11            margin-left:300px; margin-top:200px; background-color:blue; }
12      #box.mouseover { background-color: red; }
13      #box.mouseout { background-color: ''; }
14      #box.mousedown { background-color: green }
15      #box.mouseup { background-color: orange; }
16      #box > p { padding:60px; color:white; }
17    </style>
18    <script>
19
20    </script>
21  </head>
22  <body>
23    <div id="box">
24      <p> eventType </p>
25    </div>
26  </body>
27 </html>
28
```

Jquery12.html



## 4. jQuery 이벤트 처리

### ❖ 그룹 이벤트 등록 메소드 – on()

```
17 <script>
18 $(document).ready(function () {
19     var $box = $("#box");
20
21     // 그룹으로 이벤트 등록
22     $box.on({
23         mouseover : function(){//이벤트 등록
24             var $eventElement = $(this); //event 발생 엘리먼트 저장
25             $eventElement.removeClass().addClass("mouseover");
26             $eventElement.find("> p").text("mouseover");
27         },
28         mouseout : function(){
29             var $eventElement = $(this);
30             $eventElement.removeClass().addClass("mouseout");
31             $eventElement.find("> p").text("mouseout");
32         },
33         mousedown : function(){
34             var $eventElement = $(this);
35             $eventElement.removeClass().addClass("mousedown");
36             $eventElement.find("> p").text("mousedown");
37         },
38         mouseup : function(){
39             var $eventElement = $(this);
40             $eventElement.removeClass().addClass("mouseup");
41             $eventElement.find("> p").text("mouseup");
42         }
43     });
44 }
45 </script>
```

Jquery12.html

id="box"

mouseout

▼ <body>  
▼ <div id="box" class="mouseout">  
    <p>mouseout</p>  
    </div>  
</body>



## 4. jQuery 이벤트 처리

### ❖ 그룹 이벤트 등록 메소드 – off()

- on()메서드로 연결한 이벤트 제거

Jquery14.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>jQuery Event</title>
6   <script src="./js/jquery-3.2.1.min.js"></script>
7   <style>
8     #box { width:200px; height:200px; border:2px solid blue;
9           margin-left:300px; margin-top:200px; background-color:blue; }
10    #box.mouseover { background-color: red; }
11    #box.mouseout { background-color: ''; }
12    #box.mousedown { background-color: green; }
13    #box.mouseup { background-color: orange; }
14    #box > p { padding:60px; color:white; }
15  </style>
16
17  <script>
18    $(document).ready(function () {
19
20    });
21  </script>
22 </head>
23 <body>
24   <input id="btn1" type="button" value="mouseout 이벤트 제거" />
25   <input id="btn2" type="button" value="모든 이벤트 제거" />
26
27   <div id="box" class="out">
28     <p> eventType </p>
29   </div>
30
31 </body>
32 </html>
```

mouseout 이벤트 제거

모든 이벤트 제거

eventType



## 4. jQuery 이벤트 처리

### ❖ 그룹 이벤트 등록 메소드 – off()

- on()메서드로 연결한 이벤트 제거

Jquery14.html – script

```
<script>
$(document).ready(function () {
    var $box = $('#box');//id="box"요소를 선택자로 선택

    $box.on({//이벤트 소스인 box에 이벤트 등록
        mouseover:function(){
            var $eventElement = $(this);//event 발생 엘리먼트 저장
            $eventElement.removeClass().addClass('mouseover');
            $eventElement.find('> p').text('mouseover');
        },
        mouseout:function(){
            var $eventElement = $(this);
            $eventElement.removeClass().addClass('mouseout');
            $eventElement.find('> p').text('mouseout');
        },
        mousedown:function(){
            var $eventElement = $(this);
            $eventElement.removeClass().addClass('mousedown');
            $eventElement.find('> p').text('mousedown');
        },
        mouseup:function(){
            var $eventElement = $(this);
            $eventElement.removeClass().addClass('mouseup');
            $eventElement.find('> p').text('mouseup');
        }
    });

    var $btn1 = $('#btn1');
    var $btn2 = $('#btn2');

    $btn1.on('click', function(){
        $box.off('mouseout');//mouseout 이벤트 제거
    });

    $btn2.on('click', function(){
        $box.off();// 모든 이벤트 제거
    });
});
</script>
```

mouseout 이벤트 제거

모든 이벤트 제거

eventType

off()

off(eventtype)

off(eventtype, function)





## 4. jQuery 이벤트 처리

### ❖ 동적으로 생성되는 엘리먼트에 이벤트 생성

- bind() 메소드를 이용한 이벤트 연결 - v1.7부터 미 권장

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      #wrap { margin-left:300px; }
11      #nav > input { margin-left:300px; }
12      .box { width:150px; height:150px; border:2px solid blue;}
13      .box.over { background-color: red; }
14      .box.out { background-color: green; }
15      .box.click { background-color: blue; }
16      .box > p { padding:30px; color:white; }
17      #nav { margin:20px; }
18    </style>
19    <script>
20
21    </script>
22  </head>
23  <body>
24    <div id="nav">
25      <input type="button" value="box추가" />
26    </div>
27
28    <div id="wrap">
29      <div class="box out">
30        <p> eventType </p>
31      </div>
32    </div>
33  </body>
34 </html>
```

Jquery15.html

box추가

eventType



## 4. jQuery 이벤트 처리

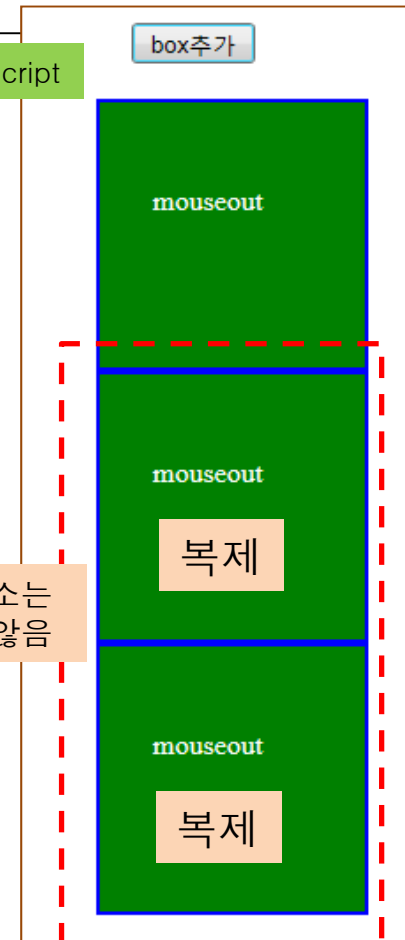
### ❖ 동적으로 생성되는 엘리먼트에 이벤트 생성

- “box추가” 버튼을 클릭하면 clone() 메소드로 box(0) 복사해서 “wrap”에 추가
- bind()는 동적으로 추가된 요소에는 이벤트가 발생하지 않음

```
19<script>
20$(document).ready(function(){
21    var $box = $('.box');
22
23    $box.bind({
24        mouseover:function(){
25            var $tg = $(this);
26            $tg.removeClass().addClass('box over');
27            $tg.find('> p').text('mouseover');
28        },
29        mouseout:function(){
30            var $tg = $(this);
31            $tg.removeClass().addClass('box out');
32            $tg.find('> p').text('mouseout');
33        },
34        click:function(){
35            var $tg = $(this);
36            $tg.removeClass().addClass('box click');
37            $tg.find('> p').text('click');
38        }
39    });
40
41    $('#nav > :input').bind('click', function(){
42        //clone()메소드를 사용해 box를 복제해서 wrap에 추가
43        $('#wrap').append($box.eq(0).clone());
44    });
45
46</script>
```

Jquery15.html - script

```
<div data-brackets-id="888" id="wrap">
  <div data-brackets-id="889" class="box out">
    <p data-brackets-id="890">mouseout</p>
  </div>
  <div data-brackets-id="889" class="box out">
    <p data-brackets-id="890">mouseout</p>
  </div>
  <div data-brackets-id="889" class="box out">
    <p data-brackets-id="890">mouseout</p>
  </div>
</div>
```



동적으로 생성된 요소는  
이벤트가 동작하지 않음



## 4. jQuery 이벤트 처리

### ❖ 동적으로 생성되는 엘리먼트에 이벤트 생성

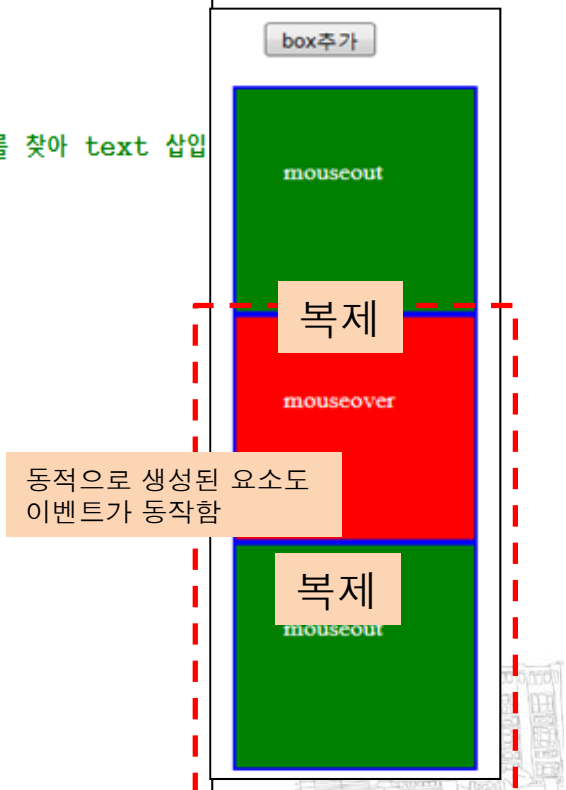
- 부모에게 이벤트를 위임하여 부모가 이벤트를 처리하도록 이벤트 위임

```
<script>
```

```
$(document).ready(function () {  
    var $box = $('#box'); //id="box"요소를 선택자로 선택  
  
    //on() 메소드를 이용하여 동적으로 생성된 요소에도 이벤트가 발생하도록 등록 - 이벤트 위임  
  
    //$("#상위요소(부모)명").on({ 리터럴객체(이벤트종류, 이벤트핸들러), "선택요소(자식)명" });  
    $("#wrap").on({ //on() 메소드로 이벤트를 등록(첫번째 인자로 리터럴객체를 넘겨줌)  
        mouseover: function() {  
            var $tg = $(this); //현재 선택한요소  
            $tg.removeClass().addClass('box over');  
            $tg.find('> p').text('mouseover'); //현재 선택한요소에서 p태그를 찾아 text 삽입  
        },  
        mouseout: function() {  
            var $tg = $(this); //현재 선택한요소  
            $tg.removeClass().addClass('box out');  
            $tg.find('> p').text('mouseout');  
        },  
        click: function() {  
            var $tg = $(this); //현재 선택한요소  
            $tg.removeClass().addClass('box click');  
            $tg.find('> p').text('click');  
        }  
    }, ".box"); //on() 메소드의 두 번째 인자값으로 "선택요소(자식)명"인 '.box'를 넘겨줌  
  
    $('#nav > :input').on('click', function() {  
        //clone() 메소드를 사용해 box를 복제해서 wrap에 추가  
        $('#wrap').append($box.eq(0).clone());  
    });  
});
```

```
</script>
```

Jquery16.html - script



## 4. jQuery 이벤트 처리

### ❖ 이벤트 위임

- 부모에게 이벤트를 위임하여 부모가 이벤트를 처리하도록 이벤트 위임

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>jQuery Event</title>
5   <script src="../js/jquery-3.2.1.min.js"></script>
6   <script>
7     // 웹 페이지를 모두 불러오면
8     $(document).ready(function () {
9       /* body 태그에 h1 태그의 click 이벤트를 위임
10        - body 태그 내부의 h1태그에서 click 이벤트가 발생했을 때
11          h1태그를 생성하여 body에 추가
12        */
13       $('body').on('click', 'h1', function() {
14         $('<h1></h1>').text($(this).text()).appendTo('body');
15       });
16     });
17   </script>
18 </head>
19 <body>
20   <h1>Header</h1>
21 </body>
22 </html>
```

Jquery16\_2.html

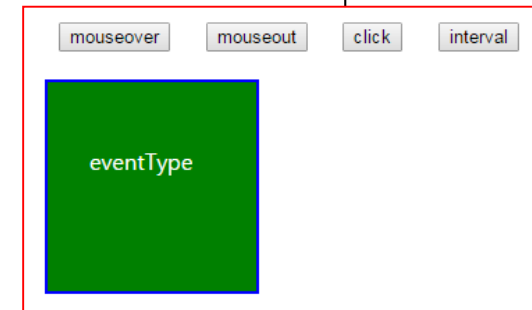
## 4. jQuery 이벤트 처리

### ❖ 강제로 이벤트 발생

- 액션이 취해지지 않아도 이벤트를 발생시켜야 할 때 가 있음. 예를 들어 마우스가 오버되지 않은 상태지만 오버 된 것처럼 효과를 적용하고 싶을 때와 같이 강제로 이벤트를 발생할 때는 **trigger()** 메소드를 사용

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      #wrap { margin-left:30px; }
11      #nav > input { margin-left:20px; }
12      #box { width:150px; height:150px; border:2px solid blue;}
13      #box.over { background-color: red; }
14      #box.out { background-color: green; }
15      #box.click { background-color: blue; }
16      #box > p { padding:30px; color:white; }
17      #nav { margin:20px; }
18    </style>
19    <script>
20
21    </script>
22  </head>
23  <body>
24    <div id="nav">
25      <input class="mouseover" type="button" value="mouseover" />
26      <input class="mouseout" type="button" value="mouseout" />
27      <input class="click" type="button" value="click" />
28      <input class="interval" type="button" value="interval" />
29    </div>
30    <div id="wrap">
31      <div id="box" class="out">
32        <p> eventType </p>
33      </div>
34    </div>
35  </body>
36 </html>
```

jquery17.html



## 4. jQuery 이벤트 처리

### ❖ 강제로 이벤트 발생

jquery17.html

```
19 <script>
20     $(document).ready(function(){
21         var $box = $('#box');//id="box" 요소를 선택하여 변수에 저장
22
23         $box.on({
24             mouseover:function(){
25                 var $eventElement = $(this);
26                 $eventElement.removeClass().addClass('box over');
27                 $eventElement.find('> p').text('mouseover');
28             },
29             mouseout:function(){
30                 var $eventElement = $(this);
31                 $eventElement.removeClass().addClass('box out');
32                 $eventElement.find('> p').text('mouseout');
33             },
34             click:function(){
35                 var $eventElement = $(this);
36                 $eventElement.removeClass().addClass('box click');
37                 $eventElement.find('> p').text('click');
38             }
39         });
40     });
```

## 4. jQuery 이벤트 처리

### ❖ 강제로 이벤트 발생

jquery17.html

```
41      var $nav = $('#nav');//id="nav" 요소를 선택하여 변수에 저장
42
43      //find()메서드는 전체 요소가 아닌 선택된 요소안에서 지정한 선택자를 찾기 때문에 속도가 빠름
44      $nav.find('> .mouseover').on('click', function(){//class="mouseover"에 이벤트 등록
45          //trigger()메소드를 호출하여 box에 mouseover 이벤트를 강제로 발생시킴
46          $box.trigger('mouseover', $(this));
47      });
48
49      $nav.find('> .mouseout').on('click', function(){
50          $box.trigger('mouseout', $(this));
51      });
52
53      $nav.find('> .click').on('click', function(){
54          $box.click();//'click'이벤트가 발생하면 box에 수동으로 'click'를 발생시킴
55      });
```



## 4. jQuery 이벤트 처리

### ❖ 강제로 이벤트 발생

jquery17.html

```
56 //interval 버튼을 클릭하면 2초마다 myTimer()함수를 호출하여 강제로 이벤트 발생
57 var myVar;
58 var cnt = 0;
59
60 $nav.find('> .interval').on('click', function(){
61     myVar = setInterval(myTimer, 2000);
62     cnt = 0;
63 });
64
65 function myTimer() {
66     if (cnt < 3) {
67         cnt++;
68         if (cnt == 1) {
69             $box.trigger('mouseover', $(this));
70         }
71         else if (cnt == 2) {
72             $box.trigger('mouseout', $(this));
73         }else {
74             $box.trigger('click', $(this));
75         }
76     }else {
77         clearInterval(myVar);// setInterval() 해제
78     }
79 };
80 });
81 </script>
```



## 4. jQuery 이벤트 처리

### ❖ 이벤트 발생시 콜백 함수로 넘어오는 event 객체

- Event 객체를 사용하면 다양한 값을 가져올 수 있음

jquery18.html

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      html * { margin:0; padding:0; }
11      body span { color:red; font-size: 30px; }
12      #wrap { width:1024px; height:1024px; }
13      #box { width:400px; height:400px; border:2px solid blue;
14            margin-left:300px; margin-top:100px; position:
15      #box > p { margin-top:60px; margin-left:150px; }
16      p > span { margin:0; }
17
18    </style>
19    <script>
20
21    </script>
22  </head>
23  <body>
24    <div id="wrap">
25      <div id="box">
26        <p>
27          clientX = <span id="clientx">0</span><br>
28          clientY = <span id="clienty">0</span><br>
29          offsetX = <span id="offsetx">0</span><br>
30          offsetY = <span id="offsety">0</span><br>
31          pageX = <span id="pagex">0</span><br>
32          pageY = <span id="pagey">0</span><br>
33          screenX = <span id="screenx">0</span><br>
34          screenY = <span id="screeny">0</span>
35        </p>
36      </div>
37    </body>
</html>
```

clientX = 0  
clientY = 0  
offsetX = 0  
offsetY = 0  
pageX = 0  
pageY = 0  
screenX = 0  
screenY = 0

## 4. jQuery 이벤트 처리

### ❖ 이벤트 발생시 콜백 함수로 넘어오는 event 객체

```
19 <script>
20     $(document).ready(function(){
21         var clientx = $('#clientx');
22         var clienty = $('#clienty');
23         var offsetx = $('#offsetx');
24         var offsety = $('#offsety');
25         var pagex = $('#pagex');
26         var pagey = $('#pagey');
27         var screenx = $('#screenx');
28         var screeny = $('#screeny');
29
30         //이벤트 발생시 콜백함수의 매개변수로 event객체를 넘겨줌
31         //event.property를 사용하면 다양한 값을 가져올 수 있음
32         $('#box').mousemove(function(e){
33             clientx.text(e.clientX); //x좌표값(스크롤 이동값 제외)
34             clienty.text(e.clientY); //y좌표값(스크롤 이동값 제외)
35             offsetx.text(e.offsetX); //이벤트발생시 이벤트 대상으로부터 x좌표값
36             offsety.text(e.offsetY); //이벤트발생시 이벤트 대상으로부터 y좌표값
37             pagex.text(e.pageX); //x좌표값(스크롤 이동값 포함)
38             pagey.text(e.pageY); //y좌표값(스크롤 이동값 포함)
39             screenx.text(e.screenX); //x좌표값(모니터 화면 기준)
40             screeny.text(e.screenY); //y좌표값(모니터 화면 기준)
41         });
42     });
43 </script>
```

jquery18.html

```
clientX = 311
clientY = 303
offsetX = 9
offsetY = 201
pageX = 311
pageY = 303
screenX = 2658
screenY = 407
```



## 4. jQuery 이벤트 처리

### ❖ 이벤트 발생시 콜백 함수에 데이터 전달

- 이벤트 발생시 콜백 함수에 데이터를 전달해 줄 수 있음
- 이벤트를 생성한 후 매개변수로 데이터를 전달하면, 이벤트 핸들러 콜백 함수가 호출되고, 매개변수로 전달된 event객체의 data 속성을 사용하면 값을 전달받을 수 있음

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="../js/jquery-3.2.1.min.js"></script>
7     <script>
8       $(document).ready(function () {
9         function calcHandler(e) { //매개변수로 전달된 데이터는 이벤트객체 타입
10          var x = parseInt($("#x").val());
11          var y = parseInt($("#y").val());
12          var result = 0;
13          //전달된 이벤트 객체는 이벤트핸들러 함수에서 객체의 data 속성으로 사용
14          if (e.data == "+") {
15            result = x+y;
16          } else if (e.data == "*") {
17            result = x*y;
18          } else {
19            throw new Error("처리 불가능한 형식!!");
20          }
21          $("#result").html(result);
22        }
23      }
24
25      $("#add").click("+", calcHandler); //이벤트핸들러에 데이터('+') 전달
26      $("#multiply").click("*", calcHandler);
27    });
28  </script>
29 </head>
30 <body>
31   X : <input type="text" id="x" value="5" /><br />
32   Y : <input type="text" id="y" value="3" /><br />
33   <hr/>
34   <button id="add">덧셈</button>
35   <button id="multiply">곱셈</button><br /><br />
36   처리 결과 : <span id="result">0</span>
37 </body>
38 </html>
```

jquery19.html

X :	<input type="text" value="5"/>
Y :	<input type="text" value="3"/>
<div>덧셈    곱셈</div>	
처리 결과 : 8	

## 4. jQuery 이벤트 처리

### ❖ 이벤트 네임 스페이스

- 이벤트 네임스페이스를 사용하면, 하나의 요소에 여러 개의 이벤트를 연결하면서 식별력을 갖도록 할 수 있으며, 이벤트 연결을 해지할 때 편리함

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>jQuery Event</title>
6   <script src="js/jquery-3.1.1.min.js"></script>
7 </script>
8 $(document).ready(function () {
9   $("#first").on({//#first에 3개의 click이벤트를 각기 다른 네임스페이스로 바인딩
10    "click.one" : function(){//event.namespaces]
11      console.log("첫번째클릭이벤트");
12    },
13    "click.two" : function(){
14      console.log("두번째클릭이벤트");
15    },
16    "click.three" : function(){
17      console.log("세번째클릭이벤트");
18    }
19  });
20
21  // #first의 click이벤트중 "one"이라는 네임스페이스를 가진 이벤트를 제거
22  $("#second").click(function() {
23    $("#first").off("click.one");
24  });
25 });
26 </script>
27 </head>
28 <body>
29   <button id="first">3개의 click이벤트를 각기 다른 네임스페이스로 바인딩 버튼 </button><br>
30
31   <button id="second">첫번째 click 이벤트제거 버튼</button>
32
33 </body>
34 </html>
```

jquery20.html

첫번째클릭이벤트	jquery20.html:11
두번째클릭이벤트	jquery20.html:14
세번째클릭이벤트	jquery20.html:17

두번째클릭이벤트	jquery20.html:14
세번째클릭이벤트	jquery20.html:17

3개의 click이벤트를 각기 다른 네임스페이스로 바인딩 버튼  
첫번째 click 이벤트제거 버튼



## 4. jQuery 이벤트 처리

### ❖ 기본 이벤트

- 이미 HTML 문서나 요소에 어떤 기능을 실행하도록 정의되어 있는 기능
- 대표적인 기본 이벤트
  - <a> 요소를 클릭했을 때 href 특성의 경로로 페이지가 이동함
  - 브라우저 화면을 마우스 오른쪽 클릭했을 때 내장 컨텍스트 메뉴가 나타남
  - <form> 요소 내부의 submit 버튼을 클릭했을 때 <form> 요소의 action 특성에 지정된 경로로 method 특성에 지정된 방식으로 전송함.
  - <input type="text" ... /> 요소에 키보드를 누르면 입력한 문자가 텍스트박스에 나타남
- 기본 이벤트는 HTML 마크업만으로도 미리 정의된 기능을 수행할 수 있기 때문에 기본 이벤트가 편리하긴 하지만 때로는 기본 이벤트 실행을 막아야 할 때도 있음



## 4. jQuery 이벤트 처리

### ❖ 기본 이벤트

- 이벤트 객체가 제공하는 메서드

메서드명	설명
preventDefault( )	기본 이벤트 실행을 중지시킨다.
stopPropagation( )	상위 요소로의 이벤트 전파를 중단시킨다.
stopImmediatePropagation( )	상위 요소로의 이벤트 전파를 중단시키고 현재 DOM의 HTML 요소의 다른 이벤트로의 전파도 중단시킨다.
isDefaultPrevented( )	preventDefault( ) 메서드가 호출되었는지를 true/false로 나타낸다.
isPropagationStopped( )	stopPropagation( ) 메서드가 호출되었는지를 true/false로 나타낸다.
isImmediatePropagationStopped( )	stopImmediatePropagation( ) 메서드가 호출되었는지를 true/false로 나타낸다.



## 4. jQuery 이벤트 처리

### ❖ 기본 이벤트 실행 중지

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>jQuery Event</title>
6   <script src="./js/jquery-3.2.1.min.js"></script>
7 </script>
8 $(document).ready(function () {
9   $("#google").on("click", function(e) {
10    //컨폼 창에서 "취소" 버튼을 클릭하면 하이퍼링크 기본 이벤트를 중지
11    if (confirm("정말로 구글로 이동할까요?") == false) {
12      e.preventDefault();
13    }
14  });
15
16  //오른쪽 마우스 클릭시 contextmenu 기본 이벤트를 중지
17 $(document).on("contextmenu", function(e) {
18   e.preventDefault();
19 });
20 //컨텍스트 메뉴 기본 이벤트를 중지
21 </script>
22 </head>
23 <body>
24 <div>
25   <a id="google" href="http://www.google.com">구글(google.com)</a>
26 </div>
27
28 </body>
29 </html>
```

jquery21.html

[구글\(google.com\)](http://google.com)

127.0.0.1:8020 내용:

정말로 구글로 이동할까요?

☐ 이 페이지가 추가적인 대화를 생성하지 않도록 차단합니다.

확인

취소



## 4. jQuery 이벤트 처리

### ❖ 이벤트를 한번만 연결 - one() 메소드

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="../js/jquery-3.2.1.min.js"></script>
7 </script>
8     $(document).ready(function () {
9         $("button").one("click", function() {
10             // <button>요소가 처음 클릭됐을 때에만 실행됨.
11             $("#text").append("첫 번째 클릭 이벤트, 한 번만 실행됨<br>");
12         });
13     });
14 </script>
15 </head>
16 <body>
17 <div>
18     <h1>one() 메소드</h1>
19     <button>마우스를 클릭해 보세요!</button>
20
21     <p id="text"></p>
22 </div>
23 </body>
24 </html>
```

jquery22.html



## 4. jQuery 이벤트 처리

### ❖ 이벤트의 버블링 단계 멈추기

- 이벤트가 발생하면 이벤트 모델은 DOM 구조를 두 번 순회함
- 이벤트가 발생하면 DOM구조의 최상위 루트에서 자식 요소를 타고 이벤트가 발생한 요소까지 내려가는 **캡처 단계**가 일어나고, 다시 반대로 이벤트 요소로부터 최상위 루트까지 올라가는 **버블 단계**가 일어남
- 이 이벤트 전파과정에서 부모 요소와 자식 요소에 각각 click이벤트를 생성한 후 자식 요소를 클릭하면, 클릭하지 않은 부모 요소에서 클릭이 발생함
- 즉, 이벤트 모델의 전파과정 중 버블 단계에서 이벤트가 발생함
- event객체의 stopPropagation() 메소드를 사용하면 사전에 차단 할 수 있음



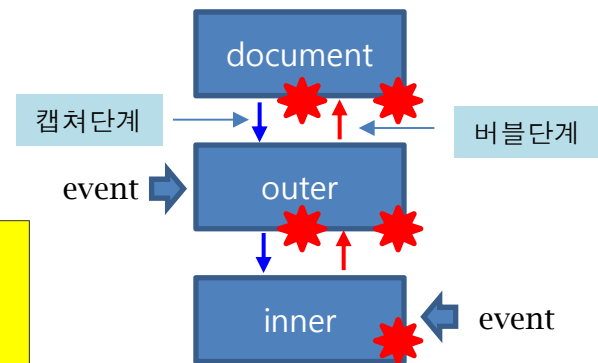
## 4. jQuery 이벤트 처리

### ❖ 이벤트의 버블링

- 자식 요소를 클릭하면, 클릭하지 않은 부모 요소까지 클릭 이벤트 발생

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <style>
8       div#outer { border:solid 1px black; background-color: yellow;
9         width:600px; height:400px; padding: 10px 10px 10px 10px;
10        position:absolute; top: 100px; left:50px; }
11      div#inner { border:solid 1px black; background-color: orange;
12        width:100px; height:100px; position:relative; top:100px; left:50px; }
13    </style>
14    <script>
15      $(document).ready(function () {
16        $("#inner").on("click", function(e) {
17          console.log("div#inner click!!");
18          console.dir(e);
19        });
20        $("#outer").on("click", function(e) {
21          console.log("div#outer click!!");
22          console.dir(e);
23        });
24        $(document).on("click", function(e) {
25          console.log("document click");
26          console.dir(e);
27        });
28      });
29    </script>
30  </head>
31  <body>
32    <div id="outer">
33      <div id="inner"></div>
34    </div>
35  </body>
</html>
```

jquery23.html



div#outer click!!	jquery23.html:21
▶ r.Event	jquery23.html:22
document click	jquery23.html:25
▶ r.Event	jquery23.html:26
div#inner click!!	jquery23.html:17
▶ r.Event	jquery23.html:18
div#outer click!!	jquery23.html:21
▶ r.Event	jquery23.html:22
document click	jquery23.html:25
▶ r.Event	jquery23.html:26

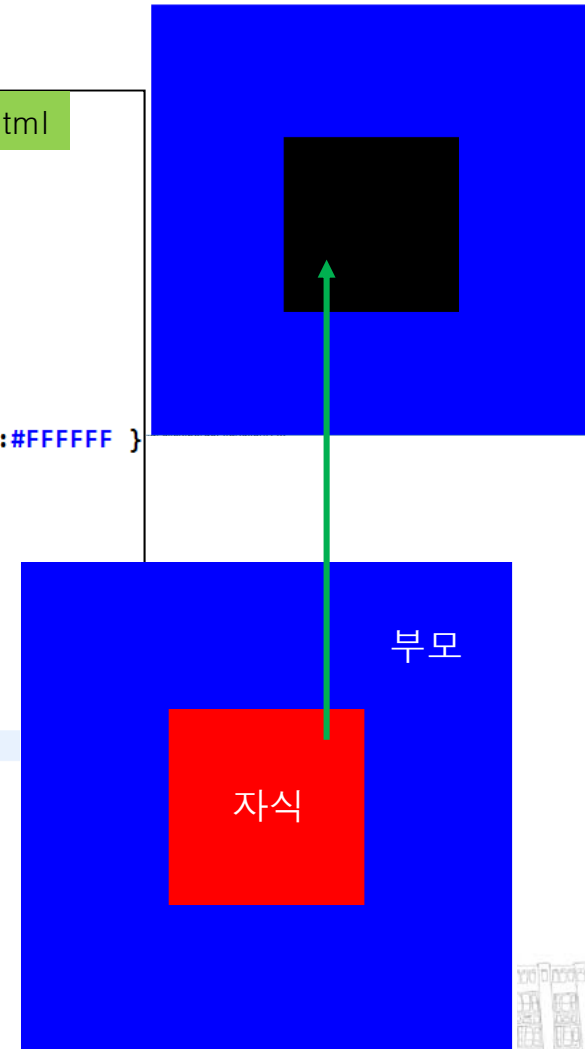
## 4. jQuery 이벤트 처리

### ❖ 이벤트의 버블링 단계 멈추기

#### ■ stopPropagation()

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7   </head>
8   <style>
9     html * { margin:0; padding:0; }
10    body { margin-left:300px; margin-top:60px; }
11    #wrap { width:200px; height:200px; border:1px solid blue;
12            background-color: blue; padding:150px;}
13    #box { width:200px; height:200px; background-color: red; color:#FFFFFF }
14  </style>
15  <script>
16    $(document).ready(function () {
17      $('#wrap').on('click', function(event){
18        $(this).css('background-color', 'yellow');
19      });
20      $('#box').on('click', function(event){
21        //이벤트가 부모 요소에 전달되지 않도록 버블링 차단
22        event.stopPropagation();
23        $(this).css('background-color', 'black');
24      });
25    });
26  </script>
27 </head>
28 <body>
29   <div id="wrap">
30     <div id="box"></div>
31   </div>
32 </body>
</html>
```

jquery24.html



부모

자식

## 4. jQuery 이벤트 처리

### ❖ 이벤트 위임(Event Delegation)

- jQuery는 이벤트의 위임을 통해 여러 요소에 공통으로 적용되는 이벤트 핸들러를, 여러 요소의 부모 요소에 한 번만 등록하여 이벤트를 처리할 수 있도록 지원
- 이벤트 위임을 이용하면 현재 존재하는 자손 요소뿐만 아니라, 동적으로 추가되는 자손요소까지 이벤트를 연결할 수 있음

jquery25.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7   </head>
8   <script>
9     $(document).ready(function () {
10       $("ul a").on("click", function(event) { // <ul>요소의 자식 요소 중 모든 <a>요소에 click 이벤트 등록,
11         event.preventDefault(); // <a>요소의 클릭시 기본 동작인 링크의 작동을 중지시킴.
12         $("#text").append("이 링크는 동작하지 않습니다!<br>");
13       });
14
15       //<ul>요소에 자식 요소 <a> 를 추가
16       $("button").one("click", function() {
17         $("ul").append('<li><a href="http://jquery.com">jQuery</a></li>');
18       });
19       /* 각 요소마다 직접 등록된 이벤트 핸들러는 현재 존재하는 <a>요소에만 등록되며,
20        * 동적으로 추가되는 요소에는 적용되지 않음.
21        * 따라서 모든 <a> 요소에 등록한 preventDefault() 메서드가 새로 추가된 <a>요소에는 적용되지 않음 */
22     });
23   </script>
24   <body>
25     <div id="wrap">
26       <h1>이벤트 핸들러의 직접 등록</h1>
27       <p>아래 버튼을 눌러 링크를 추가하고, 링크를 클릭해 보세요!</p>
28       <ul>
29         <li><a href="http://www.google.com">GOOGLE</a></li>
30         <li><a href="http://www.naver.com">NAVER</a></li>
31       </ul>
32       <button>리스트 아이템 추가</button>
33
34       <p id="text"></p>
35     </div>
36   </body>
37 </html>
```

#### 이벤트 위임

아래 버튼을 눌러 링크를 추가하고, 링크를 클릭해 보세요!

- [GOOGLE](#)
- [NAVER](#)

리스트 아이템 추가

#### 이벤트 위임

아래 버튼을 눌러 링크를 추가하고, 링크를 클릭해 보세요!

- [GOOGLE](#)
- [NAVER](#)
- [jQuery](#)

리스트 아이템 추가

이 링크는 동작하지 않습니다!  
이 링크는 동작하지 않습니다!

## 4. jQuery 이벤트 처리

### ❖ 이벤트 위임

- 형식 : `$("부모요소").on("이벤트명", "하위선택자", 이벤트 핸들러)`
  - 부모요소 : 이벤트를 위임 처리할 상위 요소
  - 하위선택자 : 이벤트를 발생시키는 자식 선택자

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7   </head>
8   <script>
9     $(document).ready(function () {
10      /* 이벤트 위임처리
11       * $("부모요소").on("이벤트명", "이벤트를 발생시키는 자식요소 선택자", "이벤트처리함수"
12       * 이벤트를 위임하면 on()메서드는 첫 번째 인수로 전달된 이벤트가 전파되었을 때,
13       * 그 이벤트를 발생한 요소가 두 번째 인수로 전달된 선택자와 같은가를 검사하여
14       * 같으면 이벤트를 처리함 */
15      $("ul").on("click", "a", function(event) { // 모든 <a>요소의 click 이벤트를 부모 요소 <ul>에 위임
16        event.preventDefault();
17        $("#text").append("이 링크는 동작하지 않습니다!<br>");
18      });
19
20      //<ul>요소에 자식 요소 <a> 를 추가
21      $("button").one("click", function() {
22        $("ul").append('<li><a href="http://jquery.com">jQuery</a></li>');
23      });
24    });
25  </script>
26 </head>
27 <body>
28   <div id="wrap">
29     <h3>이벤트 위임</h3>
30     <p>아래 버튼을 눌러 링크를 추가하고, 링크를 클릭해 보세요!</p>
31     <ul>
32       <li><a href="http://www.google.com">GOOGLE</a></li>
33       <li><a href="http://www.naver.com">NAVER</a></li>
34     </ul>
35     <button>리스트 아이템 추가</button>
36
37     <p id="text"></p>
38   </div>
39 </body>
40 </html>
```

jquery26.html

#### 이벤트 위임

아래 버튼을 눌러 링크를 추가하고, 링크를 클릭해 보세요!

- [GOOGLE](#)
- [NAVER](#)
- [jQuery](#)

리스트 아이템 추가

이 링크는 동작하지 않습니다!  
이 링크는 동작하지 않습니다!  
이 링크는 동작하지 않습니다!

## 4. jQuery 이벤트 처리

### ❖ 이벤트 위임-delegate() 메서드

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7   </head>
8   <style>
9     #wrap { width: 500px; margin: 0 auto; }
10    #nations > thead > tr { background-color:purple; color:yellow; }
11    #nations > thead > tr > th { padding :0px 1px 0px 0px; width:150px; }
12    #nations > tbody > tr > td { border-bottom: solid 1px gray; }
13  </style>
14  <script>
15    $(document).ready(function () {
16      /* 국가 리스트 배열 */
17    });
18  </script>
19  <body>
20    <div id="wrap">
21      지역 선택 :
22      <select id="region">
23        <option value="all" selected>전체 정보 보기</option>
24        <option value="asia">아시아</option>
25        <option value="america">미주</option>
26        <option value="europe">유럽</option>
27        <option value="oceania">대양주</option>
28        <option value="africa">아프리카</option>
29      </select>
30      <button id="inquiry">조회</button>
31
32      <table id="nations">
33        <thead>
34          <tr><th>국가명</th><th>수도</th><th>상세보기</th></tr>
35        </thead>
36        <tbody id="list">
37
38        </tbody>
39      </table>
40    </div>
41  </body>
42 </html>
```

jquery28.html

지역 선택 : 전체 정보 보기 ▼ 조회

국가명

수도

상세보기



## 4. jQuery 이벤트 처리

### ❖ 이벤트 위임

- 아래와 같이 정보를 조회할 수 있는 스크립트 작성

지역 선택 :

국가명	수도	상세보기
대한민국	서울	<input type="button" value="상세보기"/>
미국	워싱턴DC	<input type="button" value="상세보기"/>
프랑스	파리	<input type="button" value="상세보기"/>
영국	런던	<input type="button" value="상세보기"/>
중국	베이징	<input type="button" value="상세보기"/>
태국	방콕	<input type="button" value="상세보기"/>
모로코	라바트	<input type="button" value="상세보기"/>
라오스	비엔티안	<input type="button" value="상세보기"/>
베트남	하노이	<input type="button" value="상세보기"/>
피지	수바	<input type="button" value="상세보기"/>
솔로몬 제도	호니아라	<input type="button" value="상세보기"/>
자메이카	킹스턴	<input type="button" value="상세보기"/>
나미비아	빈트후크	<input type="button" value="상세보기"/>
동티모르	딜리	<input type="button" value="상세보기"/>
멕시코	멕시코시티	<input type="button" value="상세보기"/>
베네수엘라	카라카스	<input type="button" value="상세보기"/>
서사모아	아피아	<input type="button" value="상세보기"/>

지역 선택 :

국가명	수도	상세보기
대한민국	서울	<input type="button" value="상세보기"/>
중국	베이징	<input type="button" value="상세보기"/>
태국	방콕	<input type="button" value="상세보기"/>
라오스	비엔티안	<input type="button" value="상세보기"/>
베트남	하노이	<input type="button" value="상세보기"/>
동티모르	딜리	<input type="button" value="상세보기"/>





## 4. jQuery 이벤트 처리

### ❖ 이벤트 위임

jquery28.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>jQuery Event</title>
6   <script src="../js/jquery-3.2.1.min.js"></script>
7 </head>
8   <style>
9     #wrap { width: 500px; margin: 0 auto; }
10    #nations > thead > tr { background-color:purple; color:yellow; }
11    #nations > tbody > tr > th { padding :0px 1px 0px 0px; width:150px; }
12    #nations > tbody > tr > td { border-bottom: solid 1px gray; }
13 </style>
14 <script>
15   |
16 </script>
17 </head>
18 <body>
19   <div id="wrap">
20     지역 선택 :
21     <select id="region">
22       <option value="all" selected>전체 정보 보기</option>
23       <option value="asia">아시아</option>
24       <option value="america">미주</option>
25       <option value="europe">유럽</option>
26       <option value="oceania">대양주</option>
27       <option value="africa">아프리카</option>
28     </select>
29     <button id="inquiry">조회</button>
30
31     <table id="nations">
32       <thead>
33         <tr><th>국가명</th><th>수도</th><th>상세보기</th></tr>
34       </thead>
35       <tbody id="list">
36
37       </tbody>
38     </table>
39   </div>
40 </body>
</html>
```

## 4. jQuery 이벤트 처리

### ❖ 이벤트 위임

jquery28.html- 스크립트

```
14 ▼ $(document).ready(function () {
15     /* 국가 리스트 배열 */
16 ▼     var nationData = [
17         { no:1, name : "대한민국", capital : "서울", region:"asia" },
18         { no:2, name : "미국", capital : "워싱턴DC", region:"america" },
19         { no:3, name : "프랑스", capital : "파리", region:"europe" },
20         { no:4, name : "영국", capital : "런던", region:"europe" },
21         { no:5, name : "중국", capital : "베이징", region:"asia" },
22         { no:6, name : "태국", capital : "방콕", region:"asia" },
23         { no:7, name : "모로코", capital : "라바트", region:"africa" },
24         { no:8, name : "라오스", capital : "비엔티안", region:"asia" },
25         { no:9, name : "베트남", capital : "하노이", region:"asia" },
26         { no:10, name : "피지", capital : "수바", region:"oceania" },
27         { no:11, name : "솔로몬 제도", capital : "호니아라", region:"oceania" },
28         { no:12, name : "자메이카", capital : "킹스턴", region:"america" },
29         { no:13, name : "나미비아", capital : "빈트후크", region:"africa" },
30         { no:14, name : "동티모르", capital : "딜리", region:"asia" },
31         { no:15, name : "멕시코", capital : "멕시코시티", region:"america" },
32         { no:16, name : "베네수엘라", capital : "카라카스", region:"america" },
33         { no:17, name : "서사모아", capital : "아피아", region:"oceania" }
34     ];
35
36     /* "#list"에 출력할 HTML 마크업 템플릿 정의 */
37     var template = "<tr><td>${name}</td><td>${capital}</td>" +
38         "<td><button class='detail' data-no='${no}'>상세보기</button></td></tr>";
39
40     //국가 배열에서 지역별로 필터링된 데이터(배열)를 만들기 위한 함수
41 ▼     function filterData(region) {
42         //arr.filter(function(배열의 요소값, 배열 요소의 인덱스번호)
43 ▼         var result = nationData.filter(function(v, index) {
44 ▼             if (v.region == region) {
45                 return true;
46             }
47         });
48         return result; //필터링된 배열 리턴
49     };
```



## 4. jQuery 이벤트 처리

### ❖ 이벤트 위임

jquery28.html- 스크립트

```
51  /* 이벤트 위임처리- "button.detail"에서 발생한 "click" 이벤트를 "#list"에 위임
52  * $('부모요소').on("이벤트명", "이벤트를 발생시키는 자식요소", "이벤트처리함수" */
53  ▼  $("#list").on("click", "button.detail", function(e) {
54      var cno = $(this).attr("data-no");
55      console.dir(e);
56      alert("상세보기 : " + cno);
57  })
58
59  //조회 버튼 클릭시 조건에 맞는 데이터를 가져와 처리
60  ▼  $("#inquiry").click(function() {
61      var arr, str, s;
62      // 선택한 지역의 value 값을 sel 변수에 저장
63      var sel = $("#region option:selected").val();
64      console.log(sel);
65      // "전체정보보기" 선택한 경우 모든 국가 데이터를 arr에 저장
66  ▼  if (sel == "all") {
67      arr = nationData;
68  ▼  } else { //"전체정보보기"가 아닌경우
69      arr = filterData(sel); //필터링된 국가 데이터를 arr에 저장
70  }
71  //필터링된 데이터에서 각 국가의 요소값을 가져와 template에 저장
72  ▼  $.each(arr, function(index, item) {
73      s = template;
74      s = s.replace("${no}", item.no);
75      s = s.replace("${name}", item.name);
76      s = s.replace("${capital}", item.capital);
77
78      str += s;
79  });
80  //"조회" 버튼을 클릭할 때마다 "#list" 내용을 비우고, 다시 추가
81  $("#list").empty();
82  $("#list").append(str);
83
84  ▼  /* 각 국가의 "상세보기" 버튼은 항상 새롭게 만들어지지만 부모요소인 "#list"에
85      이벤트를 위임했기 때문에 이벤트가 동작함
86      */
87  });
88
89  </script>
```



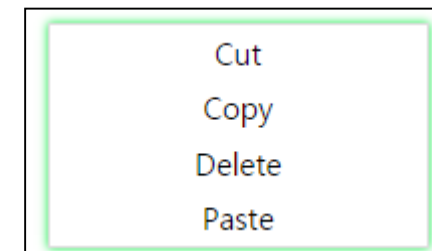
## 4. jQuery 이벤트 처리

### ❖ 사용자 정의 컨텍스트 메뉴

- 마우스 이벤트를 사용한 사용자 정의 컨텍스트 메뉴 만들기

지역 선택 : 전체 정보 보기 ▾ 조회

국가명	수도	상세보기
미국	워싱턴DC	상세보기
프랑스	파리	상세보기
영국	런던	상세보기
중국	베이징	상세보기
태국	방콕	상세보기
모로코	라바트	상세보기
라오스	万象	상세보기
베트남	하노이	상세보기
피지	수바	상세보기
솔로몬 제도	호니아라	상세보기
자메이카	킹스턴	상세보기
나미비아	빈트후크	상세보기
동티모르	딜리	상세보기
멕시코	멕시코시티	상세보기
베네수엘라	카라카스	상세보기
서사모아	아피아	상세보기



## 4. jQuery 이벤트 처리

### ❖ 사용자 정의 컨텍스트 메뉴

- 마우스 이벤트를 사용한 사용자 정의 컨텍스트 메뉴

jquery29.html

```
138<body>
139  <div id="wrap">
140      지역 선택 :
141      <select id="region">
142          <option value="all" selected>전체 정보 보기</option>
143          <option value="asia">아시아</option>
144          <option value="america">미주</option>
145          <option value="europe">유럽</option>
146          <option value="oceania">대양주</option>
147          <option value="africa">아프리카</option>
148      </select>
149      <button id="inquiry">조회</button>
150
151      <table id="nations">
152          <thead>
153              <tr><th>국가명</th><th>수도</th><th>상세 보기</th></tr>
154          </thead>
155          <tbody id="list">
156          </tbody>
157      </table>
158
159      <div id="popup">
160          <div data-cmd="cut">Cut</div>
161          <div data-cmd="copy">Copy</div>
162          <div data-cmd="delete">Delete</div>
163          <div data-cmd="paste">Paste</div>
164      </div>
165  </div>
166 </body>
167 </html>
```

jquery28.html을 복사해서 이 부분 삽입





## 4. jQuery 이벤트 처리

### ❖ 사용자 정의 컨텍스트 메뉴

- 마우스 이벤트를 사용한 사용자 정의 컨텍스트 메뉴

```
92      /* ===== 사용자 정의 컨텍스트 메뉴 ===== */
93      var rightClicked;
94      $(document).on("contextmenu",function(e) {
95          //마우스 오른쪽 클릭된 요소 파악
96          rightClicked = e.target;
97          //컨텍스트 메뉴 요소 선택
98          var menu = $("#popup");
99          //오른쪽 클릭에 대한 default event 실행 취소
100         e.preventDefault();
101         //마우스 오른쪽 클릭한 좌표에 컨텍스트 메뉴 배치
102         var pageX = e.pageX;
103         var pageY = e.pageY;
104         menu.css({top: pageY, left: pageX});
105         //컨텍스트 메뉴 크기,브라우저 화면 영역(viewport) 파악
106         var mwidth = menu.width();
107         var mheight = menu.height();
108         var screenWidth = $(window).width();
109         var screenHeight = $(window).height();
110         //화면이 스크롤 된 경우를 위해 스크롤한 높이를 파악
111         var scrTop = $(window).scrollTop();
112         //화면에서 오른쪽 경계선, 아래쪽 경계선 근처에서 클릭했을 때
113         //컨텍스트 메뉴를 안정적으로 보이도록 좌표를 다시 설정한 후 출력
114         if(pageX+mwidth > screenWidth){
115             menu.css({left:pageX-mwidth});
116         }
117         if(pageY+mheight > screenHeight+scrTop){
118             menu.css({top:pageY-mheight});
119         }
120         menu.show();
121     });
122
123     //문서에서 클릭이벤트가 발생할 경우 컨텍스트 메뉴 닫음
124     $(document).on("click", function(){
125         $("#popup").hide();
126     });
127
128     //컨텍스트 메뉴를 클릭하면 커맨드 실행
129     $("#popup > div").click(function(e) {
130         var cmd = $(this).data("cmd");
131         $(this).parent().hide();
132         alert("실행 커맨드 : " + cmd + "\n\n" +
133             "   마우스 오른쪽 클릭된 태그 : " + rightClicked.tagName);
134     });
```

jquery29.html

jquery28.html을 복사해서 스크립트 이 부분 삽입



## 4. jQuery 이벤트 처리

### ❖ 브라우저, 문서, 입력폼 관련 이벤트

이벤트	설명
resize	브라우저의 크기가 변경될 때의 이벤트
scroll	브라우저, 문서, 요소에서 스크롤링했을 때의 이벤트
ready	DOM 요소들이 완전히 로드되었을 때의 이벤트
error	문서 요소의 완전한 로드 실패했을 때의 이벤트, ex) <img /> 요소의 src 특성에서 부여된 이미지를 로딩하지 못한 경우
load	문서의 로드가 완료될 때의 이벤트
unload	현재 보고 있는 페이지에서 다른 페이지로 이동할 때의 이벤트(deprecated)
focus	요소가 포커스를 획득했을 때의 이벤트
blur	요소가 포커스를 잃었을 때의 이벤트
change	입력폼 요소의 value 속성이 변경될 때의 이벤트, select박스, 체크박스, 라디오버튼에서는 마우스로 선택만 변경해도 즉시 이벤트가 발생하지만, 텍스트박스과 같은 경우는 포커스를 잃었을 때 발생한다.
select	사용자가 요소 내부의 텍스트를 선택했을 때의 이벤트, <input type="text" ... />, <textarea /> 요소에서만 이벤트가 발생한다.
submit	<form /> 요소 내부의 submit 버튼이나 이미지 버튼을 클릭할 때 발생하는 이벤트이다.





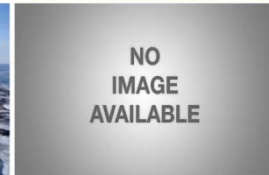
## 4. jQuery 이벤트 처리

### ❖ error 이벤트

- 문서 요소의 완전한 로드 실패했을 때 발생하는 이벤트
- 동적으로 이미지 등을 로딩하는 경우에 주로 사용

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <style>
8       img.travel { width: 300px; height:200px; background-color: yellow; overflow:auto; }
9     </style>
10    <script>
11      $(document).ready(function () {
12        var images = ["machu.jpg", "chichenitza.jpg",
13                     "capetown.jpg", "abc.jpg", "tajmahal.jpg" ];
14
15        $("img.travel").attr("src", function(i) {
16          console.log(i);
17          return "img/" + images[i];
18        }).on("error", function() {
19          // "abc.jpg" 파일이 존재하지 않기 때문에 문서로드 실패 error 이벤트 발생
20          $(this).attr("src", "img/noimage.jpg");
21        });
22      });
23    </script>
24  </head>
25  <body>
26    <div id="container">
27      <img class="travel" />
28      <img class="travel" />
29      <img class="travel" />
30      <img class="travel" />
31      <img class="travel" />
32    </div>
33  </body>
34 </html>
```

jquery31.html



# 4. jQuery 이벤트 처리

## ❖ 요소의 이동

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <style>
8       img.scape_hidden { display: none; }
9       img.scape_visible { display: block; }
10    </style>
11    <script>
12      $(document).ready(function () {
13        var handle;
14        function viewSlide() {
15          var slider = $("#slider");
16          slider.find("img.scape_visible").removeClass("scape_visible");
17          //첫번째 이미지를 마지막에 추가함으로써 이미지를 회전시킴
18          slider.find("img:first").appendTo(slider).addClass("scape_visible");
19        }
20
21        $("#slideStart").click(function() {
22          handle = setInterval(viewSlide, 2000);
23        });
24        $("#slideStop").click(function() {
25          clearInterval(handle);
26        });
27
28        viewSlide();
29      });
30    </script>
31  </head>
32  <body>
33    <div id="container">
34      <button id="slideStart">Slide Show!</button>
35      <button id="slideStop">Stop Slide Show!</button>
36      <h3>꼭 가보고 싶은 곳 5</h3>
37      <div id="slider">
38        
39        
40        
41        
42        
43      </div>
44    </div>
45  </body>
46 </html>
```

jquery32.html

Slide Show! Stop Slide Show!

꼭 가보고 싶은 곳 5



# 4. jQuery 이벤트 처리

## ❖ 요소의 복제

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7   </head>
8   <body>
9     #addForm { width:650px; }
10    .pkey { width:250px; }
11    .pname { width:150px; }
12    .ptel { width:150px; }
13    .pdel { width:100px; }
14  </body>
15  <script>
16    // 동적으로 행 추가
17  </script>
18 </html>
```

### 동적으로 행 추가

행 추가

식별번호	이름	전화번호	삭제버튼
334d53e9-dde7-4c14-8029-5c45093df567	홍길동	010-123-1234	삭제
75b4b3ff-2765-462d-8499-01dfid23516cb	최아라	010-111-1111	삭제

저장

### 동적으로 행 추가

행 추가

식별번호	이름	전화번호	삭제버튼
------	----	------	------

저장

```
19 <body>
20   <div id="container">
21     <h2>동적으로 행 추가</h2>
22     <button id="add" class="pno">행 추가 </button>
23     <div id="layout1">
24       <table id="addform">
25         <thead>
26           <tr>
27             <th class="pkey">식별번호</th>
28             <th class="pname">이름</th>
29             <th class="ptel">전화번호</th>
30             <th class="pdel">삭제버튼</th>
31           </tr>
32         </thead>
33         <tbody id="rowbody">
34           <tr>
35             <td><input type="text" class="pkey" name="key" disabled /></td>
36             <td><input type="text" class="pname" name="name" /></td>
37             <td><input type="text" class="ptel" name="tel" /></td>
38             <td><input type="button" class="pdel" value="삭제" /></td>
39           </tr>
40         </tbody>
41       </table>
42       <br />
43       <button id="save">저장</button>
44     </div>
45
46     <table id="temptable" style="display:none">
47       <tr id="temprow">
48         <td><input type="text" class="pkey" name="key" disabled /></td>
49         <td><input type="text" class="pname" name="name" /></td>
50         <td><input type="text" class="ptel" name="tel" /></td>
51         <td><input type="button" class="pdel" value="삭제" /></td>
52       </tr>
53     </table>
54   </div>
55 </body>
56 </html>
```

# 4. jQuery 이벤트 처리

## ❖ 요소의 복제

jquery33.html- 스크립트

```
15<script>
16    //랜덤 UUID값을 생성
17    function generateUUID() {
18        var d = new Date().getTime();
19        var uuid = 'xxxxxxxx-xxxx-4xxx-xxxxxxxxxxxx'.replace(/[xy]/g, function(c) {
20            var r = (d + Math.random()*16)%16 | 0;
21            d = Math.floor(d/16);
22            return (c=='x' ? r : (r&0x3|0x8)).toString(16);
23        });
24        return uuid;
25    };
26
27    $(document).ready(function () {
28        //삭제버튼을 클릭하면 "삭제"버튼을 포함하고 있는 <tr>(부모요소)을 삭제
29        $("#temprow input.pdel").click(function() {
30            $(this).parents("tr").remove();
31        });
32
33        //행추가"버튼을 클릭하면
34        $("#add").click(function() {
35            //generateUUID() 메서드를 호출하여 랜덤한 UUID값을 생성하여 KEY값으로 부여
36            var uuid= generateUUID();
37            //clone(true): 선택한요소("#temprow")를 복사(true:이벤트정보까지 복사)하여 "#rowbody"의 마지막에 추가
38            $("#temprow").clone(true).removeAttr("id")
39                .find("input:first").val(uuid).end().appendTo("#rowbody");
40            //end() : chain을 종료하고 선택집합의 이전 요소를 스택에서 찾아 반환(#temprow)
41        });
42
43        //저장버튼을 클릭하면 각 입력필드의 값을 배열로 모아서 한번에 처리
44        $("#save").click(function() {
45            var save_data = [];
46            //"#addform" 모든 <tr> 후손중에서 0보다 큰 <tr>을 선택하여 each()문으로 반복처리
47            $("#addform tr:gt(0)").each(function(i, item) {
48                var key = $(item).find("input.pkey").val();
49                var name = $(item).find("input.pname").val();
50                var tel = $(item).find("input.ptel").val();
51                save_data.push({key:key, name:name, tel:tel}); //배열에 push
52            });
53
54            //저장할 데이터 구조의 확인을 위해 콘솔로 출력
55            console.log("저장할 데이터 : ");
56            console.log(JSON.stringify(save_data)); //객체 또는 배열을 인자로 받아서 JSON 문자열로 변환
57        });
58    });
59</script>
```



## 4. jQuery 이벤트 처리

### ❖ 메뉴 만들기

Menu1

Menu2

Menu3

Menu4

Jquery34.html

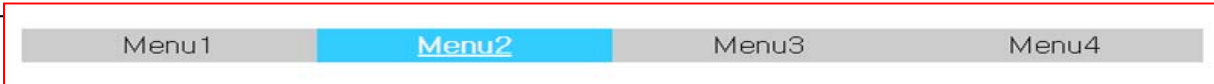
```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9
10    <style>
11      html{width:100%;height:100%;overflow-y:scroll}
12      html * {margin:0px;padding:0px;font:inherit;vertical-align:baseline;}
13      body{font-size:12px;font-family:Dotum,Arial;color:#74767a;line-height: 120%;}
14      a,a:link,a:visited{color:#74767a;text-decoration:none}
15      ul,ol{list-style:none}
16      table,fieldset,th,td,img{border:none}
17      img,input,select{vertical-align:middle}
18
19      #nav {width:800px; margin:100px auto 0 auto;}
20      #nav ul li {float:left;}
21      #nav > ul > li > a {display:block; width:200px; height:30px;
22                          line-height:30px; background:#ccc;
23                          text-align:center; color:#000; font-size:20px;}
24      #nav > ul > li.on > a {font-weight:bold; text-decoration:underline;
25                          background:#33ccff; color:#fff}
26    </style>
27    <script>
28
29  </script>
30  </head>
31  <body>
32    <div id="nav">
33      <ul>
34        <li><a href="#">Menu1</a></li>
35        <li><a href="#">Menu2</a></li>
36        <li><a href="#">Menu3</a></li>
37        <li><a href="#">Menu4</a></li>
38      </ul>
39    </div>
40  </body>
41</html>
```



## 4. jQuery 이벤트 처리

### ❖ 네비게이션 메뉴 만들기

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      html{width:100%;height:100%;overflow-y:scroll}
11      html * {margin:0px;padding:0px;font:inherit;vertical-align:baseline;}
12      body{font-size:12px;font-family:Dotum,Arial;color:#74767a;line-height: 120%;}
13      a,a:link,a:visited{color:#74767a;text-decoration:none}
14      ul,ol{list-style:none}
15      table,fieldset,th,td,img{border:none}
16      img,input,select{vertical-align:middle}
17
18      #nav {width:800px; margin:100px auto 0 auto;}
19      #nav ul li {float:left;}
20      #nav > ul > li > a {display:block; width:200px; height:30px;
21                          line-height:30px; background:#ccc;
22                          text-align:center; color:#000; font-size:20px;}
23      #nav > ul > li.on > a {font-weight:bold; text-decoration:underline;
24                          background:#33ccff; color:#fff}
25    </style>
26    <script>
27
28    </script>
29  </head>
30  <body>
31    <div id="nav">
32      <ul>
33        <li><a href="#">Menu1</a></li>
34        <li><a href="#">Menu2</a></li>
35        <li><a href="#">Menu3</a></li>
36        <li><a href="#">Menu4</a></li>
37      </ul>
38    </div>
39  </body>
40 </html>
```



Jquery34.html





## 4. jQuery 이벤트 처리

### ❖ 네비게이션 메뉴 만들기

Jquery34.html-스크립트

```
26 <script>
27 $(document).ready(function () {
28     var $menu = $('#nav > ul > li'); //이벤트를 설정할 요소 선택
29
30     // 배열에 index에 맞게 링크주소를 할당
31     var urls = ['http://www.naver.com',
32                 'http://www.daum.net',
33                 'http://www.nate.com',
34                 'http://www.google.net' ];
35
36     $menu.on({ // 요소에 이벤트 설정
37         mouseover:function(){ // 마우스오버 이벤트
38             var $tg = $(this); // this를 통해서 타겟에 접근
39             $tg.addClass('on'); // 클래스 on을 추가
40         },
41         mouseout:function(){ // 마우스아웃 이벤트
42             var $tg = $(this);
43             $tg.removeClass('on'); // 클래스 on을 삭제 합니다.
44         },
45         click:function(){
46             var $tg = $(this);
47             var i = $tg.index(); //index()메서드로 현재 클릭한 요소의 index번호를 가져옴
48             var linkURL = urls[i]; // index번호로 배열에서 링크주소를 가져옴
49             location.href = linkURL; // 링크
50         }
51     });
52 });
53 </script>
```



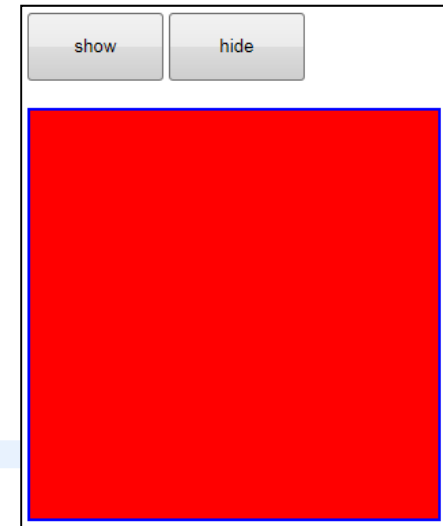


## 5. jQuery 애니메이션 처리

### ❖ show() / hide() 메소드

jquery35.html

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      html * { margin:0; padding:0; }
11      body { margin-top:150px; margin-left:500px; }
12      #wrap { margin-top:20px; }
13      #box { width:300px; height:300px; background-color:red; border:2px solid blue; }
14      input { width:100px; height:50px; }
15    </style>
16    <script>
17      $(document).ready(function () {
18        var $control = $('#control');//id="control" 요소를 control에 저장
19        var $showBtn = $control.find('> input:eq(0)');//이벤트를 설정할 요소 선택
20        var $hideBtn = $control.find('> input:eq(1)');//이벤트를 설정할 요소 선택
21        var $box = $('#box');//id="box" 요소를 box에 저장
22
23        $showBtn.click(function(){//showBtn에 이벤트등록
24          $box.show(1000);//1초만에 보여줌(커지고)
25        });
26        $hideBtn.click(function(){//hideBtn에 이벤트등록
27          $box.hide(1000);//1초만에 숨김(작아짐)
28        });
29      });
30    </script>
31  </head>
32  <body>
33    <div id="control"><!-- control button -->
34      <input type="button" value="show" />
35      <input type="button" value="hide" />
36    </div>
37    <div id="wrap">
38      <div id="box"></div><!-- 애니메이션 box -->
39    </div>
40  </body>
41 </html>
```

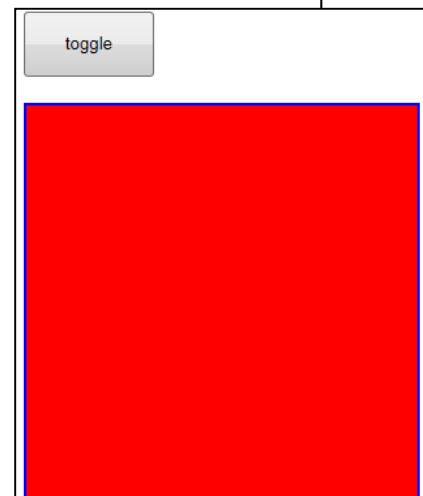


# 5. jQuery 애니메이션 처리

## ❖ toggle() 메소드

jquery36.html

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      html * { margin:0; padding:0; }
11      body { margin-top:150px; margin-left:500px; }
12      #wrap { margin-top:20px; }
13      #box { width:300px; height:300px; background-color:red; border:2px solid blue; }
14      input { width:100px; height:50px; }
15    </style>
16    <script>
17      $(document).ready(function () {
18        var $control = $('#control'); //id="control" 요소를 control에 저장
19        var $toggleBtn = $control.find('> input'); //이벤트를 설정할 요소 선택
20        var $box = $('#box'); //id="box" 요소를 box에 저장
21
22        $toggleBtn.click(function(){ //toggleBtn에 이벤트등록
23          $box.toggle(1000); //1초만에 토글 처리
24        });
25      });
26    </script>
27  </head>
28  <body>
29    <div id="control"><!-- control button -->
30      <input type="button" value="toggle" />
31    </div>
32    <div id="wrap">
33      <div id="box"></div><!-- 애니메이션 box -->
34    </div>
35  </body>
36 </html>
```

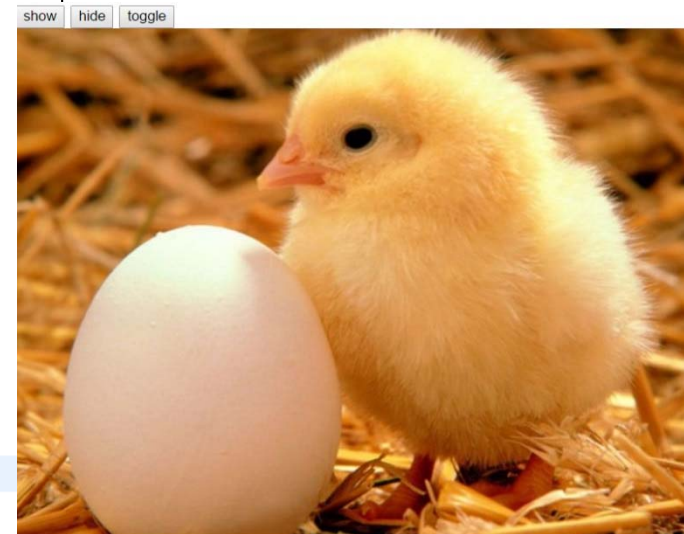


# 5. jQuery 애니메이션 처리

## ❖ toggle() 메소드

Jquery36\_2.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7     <script>
8       $(document).ready(function () {
9         $("#show").click(function() {
10           $("#a").show(1000);
11         });
12        $("#hide").click(function() {
13          $("#a").hide(1000);
14        });
15        $("#toggle").click(function() {
16          $("#a").toggle(1000);
17        });
18      });
19    </script>
20  </head>
21  <body>
22    <button id="show">show</button>
23    <button id="hide">hide</button>
24    <button id="toggle">toggle</button>
25    <br />
26    
27  </body>
28 </html>
```



# 5. jQuery 애니메이션 처리

## ❖ fadeIn() / fadeOut() 메소드

jquery37.html

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      html * { margin:0; padding:0; }
11      body { margin-top:150px; margin-left:500px; }
12      #wrap { margin-top:20px; }
13      #box { width:300px; height:300px; background-color:red; border:2px solid blue; }
14      input { width:100px; height:50px; }
15    </style>
16    <script>
17      $(document).ready(function () {
18        var $control = $('#control');
19        var $fadeInBtn = $control.find('> input:eq(0)');
20        var $fadeOutBtn = $control.find('> input:eq(1)');
21        var $box = $('#box');
22
23        $fadeInBtn.click(function(){
24          $box.fadeIn(2000);
25        });
26        $fadeOutBtn.click(function(){
27          $box.fadeOut(2000);
28        });
29      });
30    </script>
31  </head>
32  <body>
33    <div id="control"><!-- control button -->
34      <input type="button" value="fadeIn" />
35      <input type="button" value="fadeOut" />
36    </div>
37    <div id="wrap">
38      <div id="box"></div><!-- 애니메이션 box -->
39    </div>
40  </body>
41 </html>
```

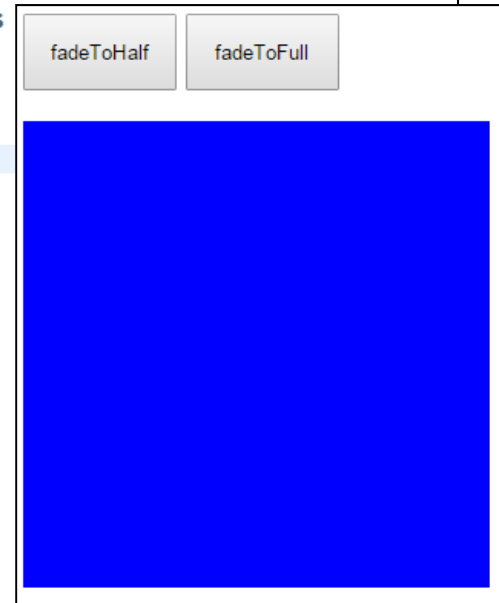


# 5. jQuery 애니메이션 처리

## ❖ fadeTO 메소드

jquery38.html

```
2< <html>
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <title>jQuery Event</title>
7
8      <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9  <style>
10     html * { margin:0; padding:0; }
11     body { margin-top:150px; margin-left:500px; }
12     #wrap { margin-top:20px; }
13     #box { width:300px; height:300px; background-color:blue; border:2px solid blue; }
14     input { width:100px; height:50px; }
15 </style>
16 <script>
17     $(document).ready(function () {
18         var $control = $('#control');
19         var $fadeToHalfBtn = $control.find('> input:eq(0)');
20         var $fadeToFullBtn = $control.find('> input:eq(1)');
21         var $box = $('#box');
22
23         $fadeToHalfBtn.click(function(){
24             $box.fadeTo(1000, 0.5); //투명도 지정
25         });
26         $fadeToFullBtn.click(function(){
27             $box.fadeTo(1000, 1);
28         });
29     });
30 </script>
31 </head>
32 <body>
33     <div id="control"><!-- control button -->
34         <input type="button" value="fadeToHalf" />
35         <input type="button" value="fadeToFull" />
36     </div>
37     <div id="wrap">
38         <div id="box"></div><!-- 애니메이션 box -->
39     </div>
40 </body>
41 </html>
```



# 5. jQuery 애니메이션 처리

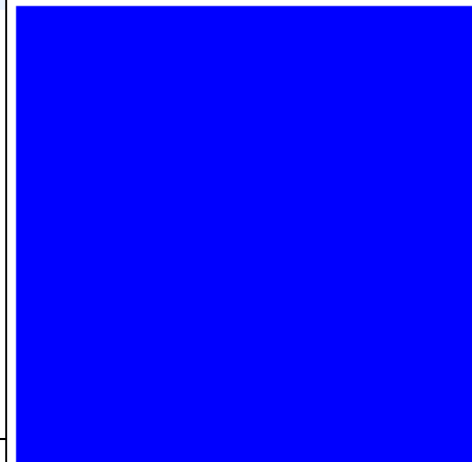
## ❖ slideUp() / slideDown() 메소드

jquery39.html

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      html * { margin:0; padding:0; }
11      body { margin-top:150px; margin-left:500px; }
12      #wrap { margin-top:20px; }
13      #box { width:300px; height:300px; background-color:blue; border:2px solid blue; }
14      input { width:100px; height:50px; }
15    </style>
16    <script>
17      $(document).ready(function () {
18        var $control = $('#control');
19        var $slideUpBtn = $control.find('> input:eq(0)');
20        var $slideDownBtn = $control.find('> input:eq(1)');
21        var $box = $('#box');
22
23        $slideUpBtn.click(function(){
24          $box.slideUp(1000); //1초안에 높이값을 0으로 줄이고
25        });
26        $slideDownBtn.click(function(){
27          $box.slideDown(1000); //1초안에 높이값을 높여주고
28        });
29      });
30    </script>
31  </head>
32  <body>
33    <div id="control"><!-- control button -->
34      <input type="button" value="slideUp" />
35      <input type="button" value="slideDown" />
36    </div>
37    <div id="wrap">
38      <div id="box"></div><!-- 애니메이션 box -->
39    </div>
40  </body>
41 </html>
```

slideUp

slideDown



## 5. jQuery 애니메이션 처리

### ❖ animate() / stop() 메소드 – 사용자 정의 애니메이션

```
1  <!DOCTYPE HTML>
2  <html>
3    <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <title>jQuery Event</title>
7
8      <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9      <style>
10         html * { margin:0; padding:0; }
11         #wrap { margin-top:140px; margin-left: 330px;}
12         #control { padding: 10px;}
13         #box { width:150px; height:150px; background-color:blue;
14               border:2px solid blue; position:absolute; left:340px; }
15      </style>
16      <script>
17
18      </script>
19    </head>
20    <body>
21      <div id="wrap">
22        <div id="control">
23          <input class="open" type="button" value="슬라이드 Down" />
24          <input class="close" type="button" value="슬라이드 Up" />
25          <input class="stop" type="button" value="슬라이드 Stop" />
26        </div>
27        <div id="box"></div>
28      </div>
29    </body>
30  </html>
```

jquery40.html

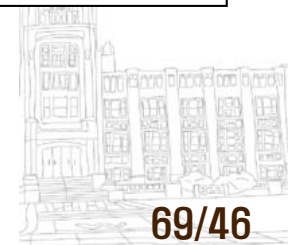
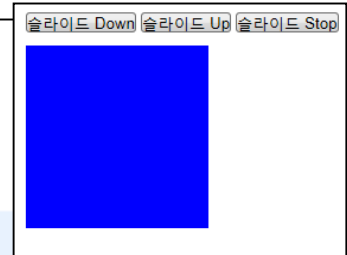




## 5. jQuery 애니메이션 처리

### ❖ animate() / stop() 메소드 – 사용자 정의 애니메이션

```
16<script>
17    $(document).ready(function () {
18        var $box = $('#box');
19        var $openBtn = $('#control > .open');
20        var $closeBtn = $('#control > .close');
21        var $stopBtn = $('#control > .stop');
22
23        $openBtn.click(function(){//1초안에 높이 500px만큼 애니메이션
24            $box.animate({height:500}, {duration:1000});
25        });
26        $closeBtn.click(function(){//1초안에 높이 0px만큼 애니메이션
27            $box.animate({height:0}, {duration:1000});
28        });
29        $stopBtn.click(function(){
30            $box.stop(); //애니메이션을 stop시킴
31        });
32    });
33</script>
```

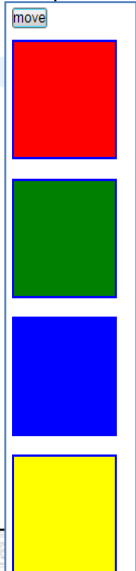


## 5. jQuery 애니메이션 처리

### ❖ animate() / delay() 메소드 – 사용자 정의 애니메이션

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <title>jQuery Event</title>
7
8     <script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
9     <style>
10      html * { margin:0; padding:0; }
11      #control { padding: 10px;}
12      #red_box { width:100px; height:100px; background-color:red;
13                border:2px solid blue; position:absolute; left:10px; }
14      #green_box { width:100px; height:100px; background-color:green;
15                  border:2px solid blue; position:absolute; left:10px; top:160px; }
16      #blue_box { width:100px; height:100px; background-color:blue;
17                  border:2px solid blue; position:absolute; left:10px; top:280px; }
18      #yellow_box { width:100px; height:100px; background-color:yellow;
19                    border:2px solid blue; position:absolute; left:10px; top:400px; }
20    </style>
21    <script>
22
23    </script>
24  </head>
25  <body>
26    <div id="wrap">
27      <div id="control">
28        <input class="left" type="button" value="move" />
29      </div>
30
31      <div id="red_box"></div>
32      <div id="green_box"></div>
33      <div id="blue_box"></div>
34      <div id="yellow_box"></div>
35    </div>
36  </body>
37 </html>
```

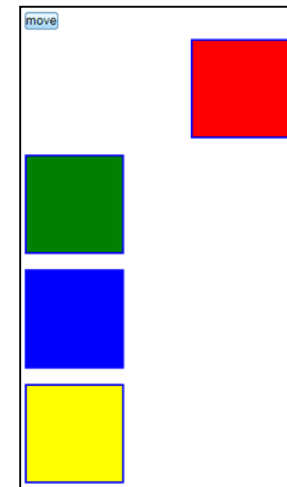
jquery41.html



## 5. jQuery 애니메이션 처리

### ❖ animate() / delay() 메소드 – 사용자 정의 애니메이션

```
21<script>
22$(document).ready(function () { //이벤트 대상요소 저장
23    var $redBox = $('#red_box');
24    var $greenBox = $('#green_box');
25    var $blueBox = $('#blue_box');
26    var $yellowBox = $('#yellow_box');
27
28    $('#wrap #control > :input').click(function(){
29        $redBox.css('left', 10)//이벤트가 발생하면 left 마진을 10px로 지정
30        .delay(100)//0.1초 지연
31        .animate({left:'90%'});
32
33        $greenBox.css('left', 10)
34        .delay(200)//0.2초 지연
35        .animate({left:'90%'});
36
37        $blueBox.css('left', 10)
38        .delay(300)//0.3초 지연
39        .animate({left:'90%'});
40
41        $yellowBox.css('left', 10)
42        .delay(400)//0.4초 지연
43        .animate({left:'90%'});
44    });
45    });
46</script>
```

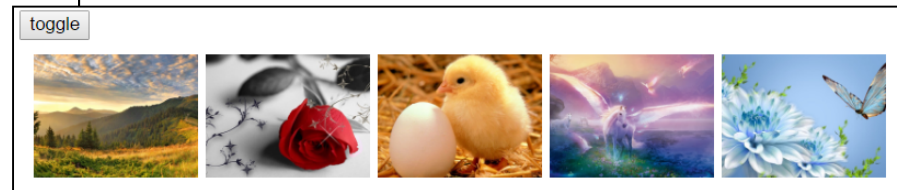


# 5. jQuery 애니메이션 처리

## ❖ 재귀함수를 통한 이미지 효과 주기

```
1 <!DOCTYPE html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title>jQuery Event</title>
6     <script src="js/jquery-3.1.1.min.js"></script>
7   </head>
8   <style>
9     div#container { margin: 10px 10px 10px 10px; }
10    img.pic { width:120px; height: 90px; }
11  </style>
12  <script>
13    $(document).ready(function () {
14      function toggleSeq(elem) {
15        elem.toggle(1000, function() {
16          var next = elem.next();
17          if (next.length > 0) {
18            toggleSeq(next);
19          }
20        });
21      }
22      $("#toggle_img").click(function() {
23        toggleSeq($("#img.pic").first());
24      });
25    });
26  </script>
27 </head>
28 <body>
29   <button id="toggle_img">toggle</button>
30   <div id="container">
31     
32     
33     
34     
35     
36   </div>
37 </body>
38 </html>
```

jquery42.html



# 5. jQuery 애니메이션 처리

## ❖ 이미지 슬라이드 효과

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <title>jQuery Event</title>
6   <style>
7
8   </style>
9
10  <script src="./js/jquery-3.2.1.min.js"></script>
11  <script>
12
13  </script>
14 </head>
15 <body>
16   <!-- 애니메이션 캔버스 구성
17     - slider_panel : 이미지를 담는 패널, 움직이며 이미지 전환 효과
18     - slider_text_panel : 글자를 담는 패널
19     - control_panel : 컨트롤 버튼을 담는 패널
20   -->
21   <div class="animation_canvas">
22     <div class="slider_panel">
23       
24       
25       
26       
27       
28     </div>
29
```

jquery43.html



Image Slider



## 5. jQuery 애니메이션 처리

### ❖ 이미지 슬라이드 효과

```
30     <div class="slider_text_panel">
31         <div class="slider_text">
32             <h1>Lorem ipsum</h1>
33             <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
34         </div>
35         <div class="slider_text">
36             <h1>Lorem ipsum</h1>
37             <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
38         </div>
39         <div class="slider_text">
40             <h1>Lorem ipsum</h1>
41             <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
42         </div>
43         <div class="slider_text">
44             <h1>Lorem ipsum</h1>
45             <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
46         </div>
47         <div class="slider_text">
48             <h1>Lorem ipsum</h1>
49             <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
50         </div>
51     </div>
52
53     <div class="control_panel">
54         <div class="control_button"></div>
55         <div class="control_button"></div>
56         <div class="control_button"></div>
57         <div class="control_button"></div>
58         <div class="control_button"></div>
59     </div>
60 </div>
61 <h1>Image Slider</h1>
62 </body>
63 </html>
```

jquery43.html



# 5. jQuery 애니메이션 처리

## ❖ 이미지 슬라이드 효과

```
6      <style>
7          * { margin:0px; padding:0px; }
8          h1 { text-align: center; }
9
10         /* Animation Canvas */
11         .animation_canvas {
12             overflow:hidden;
13             position:relative;
14             width:600px; height:400px;
15             margin:0 auto;
16         }
17
18         /* Slider Panel */
19         .slider_panel { width:3000px; position:relative; }
20         .slider_image { float:left; width:600px; height:400px; }
21
22         /* Slider Text Panel */
23         .slider_text_panel { position:absolute; top:200px; left:50px; color: white;}
24         .slider_text { position:absolute; width:250px; height:150px; }
25
26         /* Control Panel */
27         .control_panel {
28             position:absolute; top:380px;
29             left:270px; height:13px;
30             overflow:hidden;
31         }
32
33         .control_button {
34             width:12px; height:46px;
35             position:relative;
36             float:left; cursor:pointer;
37             background:url('../img/point_button.png');
38         }
39         /* point_button.png 그림에 표시부분을 제한하고 마우스 hover/active시
40            이미지가 변하는 효과를 위해 top 속성을 지정 */
41         .control_button:hover { top:-16px; }
42         .control_button.active { top:-31px; }
43     </style>
```

jquery43.html - CSS Style





## 5. jQuery 애니메이션 처리

### ❖ 이미지 슬라이드 효과

```
46     <script>
47         $(document).ready(function () {
48             // 슬라이더를 움직여주는 함수
49             function moveSlider(index) {
50                 // 슬라이더를 이동합니다.
51                 var willMoveLeft = -(index * 600);
52                 $('.slider_panel').animate({ left: willMoveLeft }, 'slow');
53
54                 // control_button에 active클래스를 부여/제거합니다.
55                 $('.control_button[data-index=' + index + ']').addClass('active');
56                 $('.control_button[data-index!=' + index + ']').removeClass('active');
57
58                 // 글자를 이동합니다.
59                 $('.slider_text[data-index=' + index + ']').show().animate({
60                     left: 0
61                 }, 'slow');
62
63                 $('.slider_text[data-index!=' + index + ']').hide('slow', function () {
64                     $(this).css('left', -300);
65                 });
66             }
67
68             // 초기 텍스트 위치 지정 및 data-index 할당
69             $('.slider_text').css('left', -300).each(function (index) {
70                 $(this).attr('data-index', index);
71             });
72
73             // 컨트롤 버튼의 클릭 핸들러 지정 및 data-index 할당
74             $('.control_button').each(function (index) {
75                 $(this).attr('data-index', index);
76             }).click(function () {
77                 var index = $(this).attr('data-index');
78                 moveSlider(index);
79             });
80
81             // 초기 슬라이더 위치 지정
82             var randomNumber = Math.round(Math.random() * 5);
83             moveSlider(randomNumber);
84         });
85     </script>
```

jquery43.html - Script