

4강 jQuery를 활용한 이벤트

0401_select_filter.html :

```
<script>
setTimeout("showTime()", 3000);

function showTime(){
    $("li:hidden").show(1000);
}
</script>
```

line0034 : `setTimeout("showTime()", 3000);`

3초의 시간이 지나서 showTime() 함수를 호출합니다.

line0037 : `$("li:hidden").show(1000);`

li 요소들 중에서 가려져 있는 (display:none) 요소를 1초의 애니메이션 시간으로 보여줍니다. : hidden

0402_select_filter.html :

```
<script>
$(function(){
    // $("li:first").css({"background-color":"#666", color:"#fff"});
    // $("li:first-child").css({"background-color":"#666", color:"#fff"});

    // $("li:last").css({"background-color":"#666", color:"#fff"});
    // $("li:last-child").css({"background-color":"#666", color:"#fff"});

    // $("li:eq(2)").css({"background-color":"#666", color:"#fff"});
    // $("li:nth-child(2)").css({"background-color":"#666", color:"#fff"});

    // $("li:even").css({"background-color":"#666", color:"#fff"});
    // $("li:nth-child(even)").css({"background-color":"#666", color:"#fff"});

    // $("li:odd").css({"background-color":"#666", color:"#fff"});
    // $("li:nth-child(odd)").css({"background-color":"#666", color:"#fff"});

    $("li:nth-child(3n+2)").css({"background-color":"#666", color:"#fff"});
});
</script>
```

line0035 : `$("li:first").css({"background-color":"#666", color:"#fff"});`

문서상에서 첫 번째 li 요소를 선택합니다.

line0036 : \$("li:first-child").css({"background-color":"#666", color:"#fff});

각각의 ul 요소의 첫 번째 li 요소를 선택합니다.

line0038 : \$("li:last").css({"background-color":"#666", color:"#fff});

문서상에서 마지막 li 요소를 선택합니다.

line0039 : \$("li:last-child").css({"background-color":"#666", color:"#fff});

각각의 ul 요소의 마지막 li 요소를 선택합니다.

line0041 : \$("li:eq(2)").css({"background-color":"#666", color:"#fff});

문서상에서 3번째 li 요소를 선택합니다.

line0042 : \$("li:nth-child(2)").css({"background-color":"#666", color:"#fff});

각각의 ul 요소의 2번째 li 요소를 선택합니다.

line0044 : \$("li:even").css({"background-color":"#666", color:"#fff});

문서상에서 짝수 li 요소를 선택합니다.

line0045 : \$("li:nth-child(even)").css({"background-color":"#666", color:"#fff});

각각의 ul 요소의 짝수 li 요소를 선택합니다.

line0047 : \$("li:odd").css({"background-color":"#666", color:"#fff});

문서상에서 홀수 li 요소를 선택합니다.

line0048 : \$("li:nth-child(odd)").css({"background-color":"#666", color:"#fff});

각각의 ul 요소의 홀수 li 요소를 선택합니다.

line0050 : \$("li:nth-child(3n+2)").css({"background-color":"#666", color:"#fff});

(3*0+2), (3*1+2), (3*3+2)와 같은 연산에 해당하는 요소를 선택합니다.

0403_select_filter.html :

```
<script>
$(function(){
    // $(".container").children("p").css({"background-color":"#666", color:"#fff});
    // $(".container").find("p").css({"background-color":"#666", color:"#fff});
    // $("#part1").prev().css({"background-color":"#666", color:"#fff});
    // $("#part1").next().css({"background-color":"#666", color:"#fff});
    // $("#part1").siblings().css({"background-color":"#666"});
    $("p#part1").parent().css({"background-color":"#666"});
});
</script>
```

line0055 : \$(".container").children("p").css({"background-color":"#666", color:"#fff});
container 클래스 하위 자식 요소인 p 요소를 선택합니다.

line0056 : \$(".container").find("p").css({"background-color":"#666", color:"#fff});
container 클래스 하위 p 요소를 모두 선택합니다.

line0057 : \$("#part1").prev().css({"background-color":"#666", color:"#fff});
part1 아이디의 이전 p 요소를 선택합니다.

line0058 : \$("#part1").next().css({"background-color":"#666", color:"#fff});
part1 아이디의 다음 p 요소를 선택합니다.

line0059 : \$("#part1").siblings().css({"background-color":"#666"});
part1 아이디의 주변 p 요소를 선택합니다.

line0060 : \$("#part1").parent().css({"background-color":"#666"});
part1 아이디의 상위 요소를 선택합니다.

0404_filtering.html :

```
<script>
$(function(){
    // $(".container p").first().css({"background-color":"#666", color:"#fff});
    // $(".container p").last().css({"background-color":"#666", color:"#fff});
    // $(".container p").eq(1).css({"background-color":"#666", color:"#fff});
    // $(".container p").filter(".middle").css({"background-color":"#666", color:"#fff});
    // $(".container p").not(".middle").css({"background-color":"#666", color:"#fff});
    $(".container div").has("p").css({"background-color":"#666", color:"#666"});
});
</script>
```

line0055 : \$(".container p").first().css({"background-color":"#666", color:"#fff});
container 클래스의 첫 번째 p 요소를 스타일합니다.

line0056 : \$(".container p").last().css({"background-color":"#666", color:"#fff});
container 클래스의 마지막 p 요소를 스타일합니다.

line0057 : \$(".container p").eq(1).css({"background-color":"#666", color:"#fff});
container 클래스의 두 번째 p 요소를 스타일합니다.

line0058 : \$(".container p").filter(".middle").css({"background-color":"#666", color:"#fff});
container 클래스 하위의 middle 클래스를 스타일합니다.

line0059 : \$(".container p").not(".middle").css({"background-color":"#666", color:"#fff});

container 클래스 하위의 middle 클래스가 아닌 p 요소를 스타일합니다.

line0060 : \$(".container div").has("p").css({"background-color":"#666", color:"#666});

container 클래스 하위 div 요소 중 p 요소를 가지고 있는 p 요소를 스타일합니다.