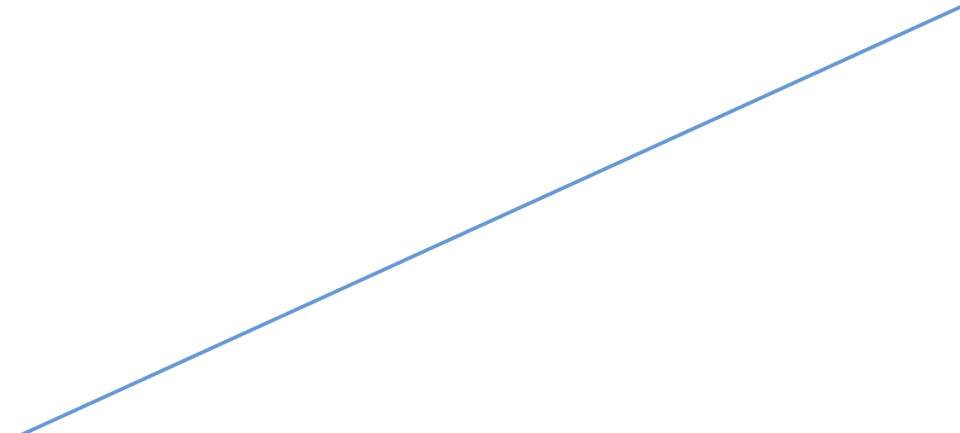


jQuery

Front-End 개발자 양성 과정



DinoWorks
DIGITAL NOMAD

- Basic
 - Event
 - HTML
 - Animation
 - 기타 Method
- 

➤ jQuery 개요

- 자바스크립트 라이브러리
- 자바스크립트 프로그래밍을 단순화할 수 있게 함
- 배우기 쉬움

```
$(document).ready(function(){  
    $('p').click(function(){  
        $(this).hide();  
    });  
});
```

➤ jQuery 기능

- HTML/DOM 변경
- CSS 변경
- HTML 이벤트 메소드
- 효과(Effect) 및 애니메이션
- AJAX
- 부가기능(플러그인)

➤ jQuery 설치

```
<head>  
  <script src="../../js/jquery-3.3.1.min.js"></script>  
</head>
```

- 3.x 버전은 IE 6,7,8을 지원하지 않는 버전
- Download : www.jquery.com

➤ jQuery 문법

- HTML 요소를 선택하고 특정 행동을 수행하는 방식으로 구성
- 기본문법 : `$(selector).action()`
- ‘\$’ 표시는 jQuery를 정의하고 접근하게 하는 문자
- ‘(selector)’는 HTML 요소를 선택하기 위한 요소로 CSS 선택자 문법과 동일

```
$(this).hide();  
$("p").hide();  
$(".test").hide();  
$("#test").hide().css({'display': 'none'}); // chaining
```

➤ Document Ready 이벤트

```
$(document).ready(function(){  
    // HTML 문서(DOM)이 다 불러진 이후에 실행될 jQuery문 삽입  
});
```

➤ jQuery 선택자(Selector)

- 요소(element) 선택자
- #id 선택자
- .class 선택자
- * : 전체 선택자
- this : 현재 요소 선택자
- p:first, p:first-child, li:eq(3), li:gt(2), li:lt(1) : 가상 선택자
- a[href], a[target= "_blank"], a[target != '_blank'] : 속성 선택자
- \$("button")
- \$("tr:even"), \$("tr:odd")

➤ jQuery API

- `$('#header').on()` → 객체의 메소드(method) 형태를 가짐
- 메소드 내부에 파라미터를 조작하는 형태로 구성
- 포함가능한 파라미터의 형태
 - 숫자 : 100, 300 등
 - 문자열 : 'click', 'mouseenter focus' 등
 - 객체 : `{left: '100px', top: '50px'}`, `{src: '../img/visual.jpg', alt: '최고의 가성비 하이셈'}` 등
 - 함수 : `function(e) { ... }` → 동작을 정의
 - 기타 : `$(selector).index($(this).parent())` → 다른 선택 요소의 그룹이 포함되기도 함
- 주요 API
 - Event : `on()`, `trigger()`, `off()`
 - HTML(Get/Set, Add/Remove) : `text()`, `html()`, `val()`, `attr()`, `append()`, `prepend()`, `after()`, `before()`, `empty()`, `remove()`
 - HTML(CSS) : `addClass()`, `removeClass()`, `toggleClass()`, `css()`
 - HTML(Traversing) : `parent()`, `find()`, `prev()`, `next()`
 - HTML(Dimension) : `width()`, `height()`, `innerWidth()`, `innerHeight()`, `outerWidth()`, `outerHeight()`, `offset()`, `position()`
 - Animation : `animate()`, `stop()`
 - Etc : `index()`, `each()`, `this`

➤ jQuery 이벤트(일부)

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

```
$('#p').click();
```

```
$('#p').click(function(){  
    alert('This is a paragraph');  
});  
$('#p').on('click', function(){  
    alert('clicked');  
});
```


➤ jQuery Event

- on() : Event 정의
- off() : Event 제거
- trigger() : 이벤트 발생

```
$('#p').on('mouseenter focus', function(){
    $(this).css({background-color: 'red', border: '3px solid blue'});
});
$(document).on('click', 'a[href="#"]', function(e) {
    e.preventDefault();
}); // 이벤트 위임
$('#p').trigger('focus');
$('#p').off('click');
```

➤ jQuery GET

- `text()` : 선택한 요소의 텍스트 콘텐츠를 리턴하거나 설정
- `html()` : 선택한 요소의 HTML 콘텐츠를 리턴하거나 설정
- `val()` : Form 요소의 값(value)을 리턴하거나 설정
- `attr()` : 선택한 요소의 값을 리턴하거나 설정

```
$('#btn1').click(function(){  
    alert('TEXT : ' + $('#test').text());  
});  
  
$('#btn2').click(function(){  
    alert('HTML : ' + $('#test').html());  
});  
  
$('#btn3').click(function(){  
    alert('VALUE : ' + $('#input').val());  
});  
  
$('#btn4').click(function(){  
    alert('ATTR : ' + $('#a:eq(0)').attr('href'));  
});
```

➤ jQuery SET

```
$('#btn1').click(function(){  
    $('#test').text('Hello World');  
});  
  
$('#btn2').click(function(){  
    $('#text li').html('<li>Inserted 0' + i + '</li>');  
});  
  
$('#btn4').click(function(){  
    $('#text a:eq(0)').attr('href', 'http://naver.com');  
});  
  
$('#btn4').click(function(){  
    $('#text a:eq(0)').attr({  
        'href': 'http://naver.com',  
        'target': '_blank'  
    });  
});
```

➤ jQuery ADD

- `append()` : 선택된 요소의 자식요소 중 가장 끝에 콘텐츠 삽입
- `prepend()` : 선택된 요소의 자식요소 중 시작부분에 콘텐츠 삽입
- `after()` : 선택된 요소의 바로 뒤(형제요소)에 콘텐츠 삽입
- `before()` : 선택된 요소의 바로 앞(형제요소)에 콘텐츠 삽입

← **before()** `$('ul').before(.....);`

← **prepend()** `$('ul').prepend(.....);`

`List 1`
`List 2`
`List 3`

← **append()** `$('ul').append(.....);`

← **after()** `$('ul').after(.....);`

➤ jQuery Remove

- `remove()` : 선택된 요소와 그 하위 요소까지 모두 제거
- `empty()` : 선택된 요소의 하위 요소만 제거

```
<ul> ← remove()    $('ul').remove();    $('ul li').remove('.first');
```



```
  <li class="first">List 1</li>  
  <li>List 2</li> ← empty()    $('ul').empty();  
  <li>List 3</li>
```



```
<ul>
```

➤ jQuery CSS Classes

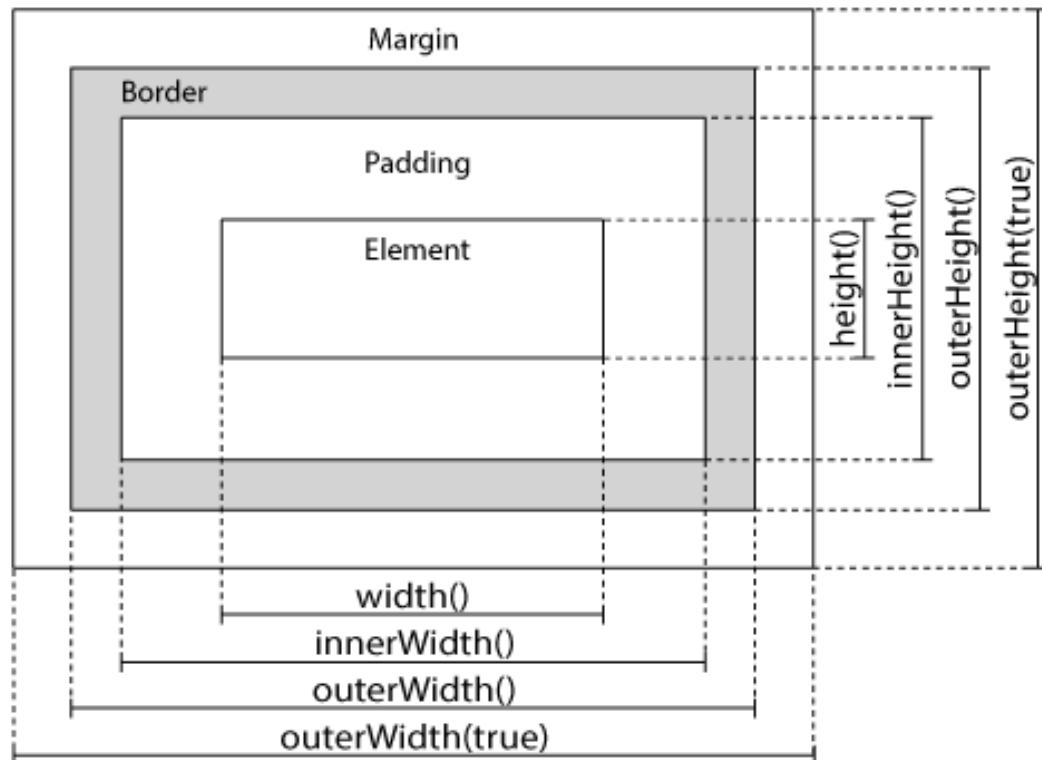
- `addClass()`
- `removeClass()`
- `toggleClass()`
- `css()`

```
$('#menu li a').on('mouseenter', function(){  
    $(this).addClass('over');  
});
```

```
$('p').on('mouseenter focus', function(){  
    $(this).css({'background-color': 'red', 'border': '3px solid blue'});  
});
```

➤ jQuery Dimension Methods

- `width()`
- `height()`
- `innerWidth()`
- `innerHeight()`
- `outerWidth()`
- `outerHeight()`



➤ jQuery Dimension Methods

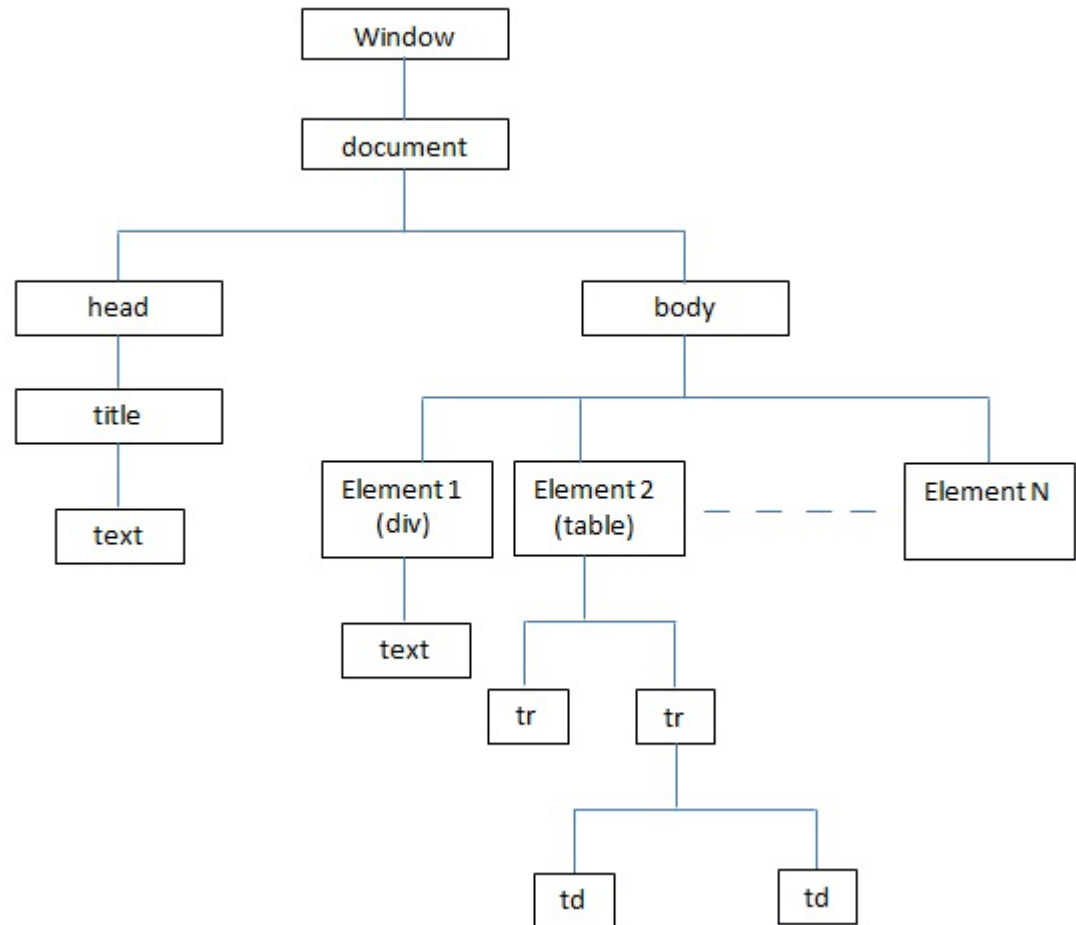
- offset()
- position()

```
// document(문서) 기준의 위치값 반환  
var offsetLeft = $('#header h1').offset().left;  
var offsetTop = $('#gnb').offset().top;
```

```
// offset parent(위치 기준을 제공하는 요소) 기준의 위치값 반환  
var offsetLeft = $('#header h1').position().left;  
var offsetTop = $('#gnb').position().top;
```


➤ DOM Traversing

- `parent()`
- `parents()`
- `parentUntil()`
- `children()`
- `find()`
- `siblings()`
- `next()`, `nextAll()`, `nextUntil()`
- `prev()`, `prevAll()`, `prevUntil()`
- `first()`
- `last()`
- `eq()`
- `filter()`
- `not()`



➤ jQuery Animation

- 변화시킬 css 속성, 속도(ms), 종료 후 호출내용(callback 함수)에 대한 설정 가능

```
$('#button').click(function(){  
    $('#div.animation').animate({'left': '500px'}, 3000, function(){  
        alert('Animation is finished');  
    });  
});
```

- 연속된 애니메이션(Chaining)

```
$('#button').click(function(){  
    $('#div.animation').animate({'left': '500px'}, 3000).animate({'left': 0}, 1000);  
});
```

➤ jQuery Animation

○ 애니메이션의 정지

```
$('#anibox').on('mouseenter', function(){  
    $('#div.animation').stop();  
});
```

- stop() 에는 stopAll(true/false), gotoEnd(true, false) 2개의 파라미터가 존재
- stopAll : stop(true)인 경우 Que에 있는 모든 애니메이션이 중지되며 false(default 값)인 경우 현재 진행 중인 애니메이션만 정지됨
- gotoEnd : stop(false, true)인 경우 현재 상태로 정지되는 것이 아니라 현재 애니메이션의 가장 최종 상태로 이동하여 정지됨

➤ .index(), this

```
$('#ul li a').on('click', function(){  
    var index = $('#ul li').index($(this).parent());  
    alert(index);  
});
```

➤ .each()

```
$('#button').on('click', function(){  
    $('#ul li').each(function(){  
        var index = $('#ul li').index($(this));  
        if (index === 3) {  
            $(this).remove();  
        }  
    });  
});
```