

자바스크립트 DOM요소를 쉽게 사용할 수 있는

**Jquery**

7driety



JavaScript:  
The Definitive Guide

Flanagan O'REILLY

# Jquery 필요성 & 사용 전 준비사항

HTML 문서 탐색, 이벤트 처리, 애니메이션, AJAX 통신 간편 처리 라이브러리

제이쿼리는 자바스크립트 DOM 을 간편하게 사용하기 위해서 사용하는 라이브러리로 중소기업에서 주로 이용합니다.

- 바닐라 자바스크립트 프로그래밍 : 집에서 재료공수부터 완성까지 모두 직접하여 원하는 요리 만들기
- 제이쿼리 프로그래밍 : 식당에서 '메뉴판(CDN)'을 받고 원하는 메뉴 주문하고 완성된 요리 받기

사용 전 필수준비사항 CDN 준비 <https://cdnjs.com/libraries/jquery>

- 자바스크립트를 베이스로 만들어진 언어로 제이쿼리를 사용하기 위해선 script CDN 연결이 필수입니다. 연결 후에 제이쿼리를 작성하세요! 아래 Copy Script Tag를 클릭 복사 후 head 안에 붙여넣기합니다.



# Jquery CDN과 기본 연결, 개념

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
```

- 제이쿼리가 시작은 모두 \$ 으로 시작합니다. \$ == 제이쿼리
- \$() 라고 작성하면 () 괄호 안을 DOM을 제어할 수 있는 함수영역이라고 인식하여 다양한 요소를 쉽게 호출할 수 있습니다.
- script태그는 head 태그 안 기존 외부 script 태그들보다 먼저 작성합니다.



# Jquery 아이디, 클래스, 태그 선택 방법

id, class, tag(element) 선택자를 jquery로 쉽게 선택할 수 있습니다.

- 아이디선택자    `$('#아이디명');`      예) `<h1 id="name1"></h1>`      `$('#name1')`
- 클래스선택자    `$('.클래스명');`      예) `<h1 class="name2"></h1>`      `$('.name2')`
- 태그선택자      `$('태그이름');`      예) `<h1></h1>`      `$('h1')`

**DOM 변수 선언 법** \*변수명 앞 제이쿼리 변수 표시를 위해 `$변수명`을 작성하기도 합니다.

`const 변수명 = $('#아이디명');`    //제이쿼리로 아이디 선택 후 `const`로 선언된 변수에 대입한다

`let $변수명 = $('.클래스명');`    //제이쿼리로 클래스 선택 후 `let`으로 선언된 변수에 대입한다

`var 변수명 = $('태그명');`        //제이쿼리로 태그 선택 후 `var`로 선언된 변수에 대입한다

# Jquery 속성과 메서드 및 함수 문법

`$(‘DOM’).속성;`

`$(‘DOM’).함수();`

`$(‘DOM’).함수(‘속성’, ‘값’);`

`$(‘DOM’).메서드();`

`$(‘DOM’).메서드(‘속성’, ‘값’);`

# Jquery 이벤트 관리

`$(대상).이벤트(실행함수(){ });`

- `$(대상).click(function(){ alert('클릭 완료'); })`
- `$(대상).click(function(){ $(대상).css('background-color', 'lime'); })`
- `$(대상).mouseover( ()=>{ console.log('마우스 오버 완료'); })`
- `$(대상).mouseout( ()=>{ console.log('마우스 아웃 완료'); })`
- `$(대상).mouseover(function(){ $(대상).addClass('design1'); })`

# Jquery 가족관계 선택

parent, children, find, siblings 사용하여 DOM 가족 관계 선택 가능

- `$(대상).parent().css('border', '1px solid black');` 부모 요소 선택
- `$(대상).parent().parent().addClass('design');` 부모의 부모요소 선택
- `$(대상).children('.item').hide();` 자식 선택
- `$(대상).children('a').toggleClass('highlight');` 자식 선택
- `$(대상).find('li.active').css('color', 'aqua');` 자식, 자손 선택
- `$(대상).find('span').text('새로운 내용');` 자식, 자손 선택
- `$(대상).siblings('div').text('새로운 내용');` 형제 선택

# Jquery 자주 이용하는 속성&메서드&함수

- 요소 숨기기/보이기      `$(대상).hide();`
  - `$(대상).show();`
- 요소 스타일 적용하기 `$(대상).css('background-color', 'yellow');`
- 요소 내 글자 입력하기      `$(대상).text('삽입 글자');`
- 요소 내 태그 입력하기      `$(대상).html('<span>삽입 태그와 내용</span>');`
- 요소 클래스 적용하기 `$(대상).addClass('적용클래스명');`
- 요소 클래스 해제, 토글      `$(대상).removeClass('제거클래스명');`
  - `$(대상).toggleClass('반복클래스명');`
- 홀수, 짝수 스타일 적용      `$(대상:odd).css('color', 'red');` // 짝수 even
- 다중 선택자 적용하기 `$(그룹대상, 그룹대상).css('font-size', '3rem');`
- input값 가져오기      `$(대상).val();`
- 요소 상태 참, 거짓 구분      `$(대상).is("클래스");` // :hidden, :visible 작성가능



# Jquery 애니메이션 효과

- `$(대상).hide();` 대상 숨기기 ex) `display = 'none'` 동일함
- `$(대상).show();` 대상 보이기 ex) `display = 'block'` 동일함

## slide~fade 지속시간 설정 가능 500 == 0.5초, `fadeOut(500)`

- `$(대상).slideUp();` 점점 위로 접히며 숨기기 ex) `height max-content -> 0` 동일함
- `$(대상).slideDown();` 점점 아래로 펼쳐지며 보이기 ex) `height 0 -> max-content` 동일함
- `$(대상).fadeIn();` 점점 나타나기 ex) `opacity 0 -> opacity 1` 동일함
- `$(대상).fadeOut();` 점점 사라지기 ex) `opacity 1 -> opacity 0` 동일함
- `$(대상).stop(true).애니메이션함수명();`
  - 애니메이션 연속 재생 방지 `stop(true)`

# Jquery 애니메이션 효과

animate 함수는 다양한 CSS 속성을 부드럽게 변화시켜 애니메이션 효과 부여 기능입니다.

**`$(대상).animate({ 속성: '값', 속성: '값' }, 지속시간, 가속도, 콜백함수());`**

`$(대상).animate({`

    height: '200px',       //대상의 높이값 1초 동안 200으로 변경하기

    opacity:0.5,           //대상의 투명도 1초 동안 0.5로 변경하기

`}, 1000, 'ease', function(){`

    console.log('애니메이션 완료'); //1초 애니메이션이 끝나면 함수 내 console.log 실행하기

`});`

# Jquery 애니메이션 효과 + animate 체이닝기법

animate 함수는 다양한 CSS 속성을 부드럽게 변화시켜 애니메이션 효과 부여 기능입니다.

**`$(대상).animate().animate().animate();`**

`$(대상)`

`//애니메이션 대상 설정(변수 대체가능)`

`.animate({ width: '100px' }, 1000)`

`//1초동안 대상 너비값 100px 변경(기본속도 ease)`

`.animate({ height: '200px' }, 500, 'ease')` `//0.5초동안 점점 느리게 대상 높이값 200px로 변경`

`.animate({ color: '#000' }, 1500, 'linear');` `//1.5초동안 일정속도로 대상 글자색 변경`