

# jQuery

## I. 개발환경 설정과 시작하기 전에

### 1. 시작하기

javascript Framework 중 가장 많이 사용되는 프레임워크로 가장 중요한 아래사항을 꼭 익혀두시기 바랍니다.

설정 1 : 다운로드 후 설정

- 다운로드 URL : <https://jquery.com/download/>

설정 2 : CDN

```
<script src=https://code.jquery.com/jquery-3.6.1.js> </script>
```

- html 파일에 jQuery 로드 방법
- 선택자 사용방법
- 이벤트 메서드 사용방법
- DOM 생성 및 삭제

### 1) 선택자

#### ① 기본 선택자

선택자	설명
\$("태그")	태그명을 선택자로 사용
\$("#id")	css 의 id 를 선택자로 사용
\$(".class")	css 의 class 를 선택자로 사용

#### ② 사용자 정의 선택자

기본 선택자와 함께 사용하며 콜론(👉)을 붙여 사용합니다.

선택자	설명
:eq(n)	선택한 요소 중에서 인덱스가 n 인 요소를 선택함.
:odd	선택한 요소 중에서 인덱스가 홀수인 요소를 모두 선택함.
:even	선택한 요소 중에서 인덱스가 짝수인 요소를 모두 선택함.

선택자	설명
:gt(n)	선택한 요소 중에서 인덱스가 n 보다 큰 요소를 모두 선택함.
:lt(n)	선택한 요소 중에서 인덱스가 n 보다 작은 요소를 모두 선택함.
:first	선택한 요소 중에서 첫 번째 요소를 선택함.
:last	선택한 요소 중에서 마지막 요소를 선택함.
:not(선택자)	선택한 요소 중에서 지정한 선택자와 일치하지 않는 요소를 모두 선택함.
:root	선택한 요소 중에서 최상위 루트 요소를 선택함.
:has(선택자)	선택한 요소 중에서 지정한 선택자와 일치하는 자손 요소를 갖는 요소를 모두 선택함.
:empty	선택한 요소 중에서 자식 요소를 가지고 있지 않은 요소를 모두 선택함.
:parent	선택한 요소 중에서 자식 요소를 가지고 있는 요소를 모두 선택함.

### 3) input 요소의 선택

특정 <input>요소를 선택할 수 있는 선택자는 다음과 같습니다.

선택자	설명
:button	type 속성값이 "button"인 요소를 모두 선택함.
:checkbox	type 속성값이 "checkbox"인 요소를 모두 선택함.
:file	type 속성값이 "file"인 요소를 모두 선택함.
:image	type 속성값이 "image"인 요소를 모두 선택함.
:password	type 속성값이 "password"인 요소를 모두 선택함.
:radio	type 속성값이 "radio"인 요소를 모두 선택함.
:reset	type 속성값이 "reset"인 요소를 모두 선택함.
:submit	type 속성값이 "submit"인 요소를 모두 선택함.
:text	type 속성값이 "text"인 요소를 모두 선택함.
:input	<input>, <textarea>, <select>, <button> 요소를 모두 선택함.
:checked	type 속성값이 "checkbox" 또는 "radio"인 요소 중에서 체크되어 있는 요소를 모두 선택함.

선택자	설명
:selected	<option> 요소 중에서 선택된 요소를 모두 선택함.
:focus	현재 포커스가 가지고 있는 요소를 선택함.
:disabled	비활성화되어있는 요소를 모두 선택함.
:enabled	활성화되어있는 요소를 모두 선택함.

## 2. 이벤트 메서드의 활용 : .on() Method

```
$(선택자).on(event[, selector][, data], function(){ ... })
```

- events : 1개 또는 2개 이상의 공백으로 구분된 이벤트명
- selector : 이벤트를 바인딩할 매치드셋의 후손 선택자
- data : 이벤트핸들러에 전달할 데이터, 이벤트핸들러의 매개변수, event.data
- function : 이벤트가 발생했을 때 실행될 함수

### ① 마우스 이벤트 메서드

메소드	설명
.click()	자바스크립트의 "click" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "click" 이벤트를 발생시킴.
.dblclick()	자바스크립트의 "dblclick" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "dblclick" 이벤트를 발생시킴.
.hover()	자바스크립트의 "mouseenter"와 "mouseleave" 이벤트를 같이 이벤트 핸들러와 연결함.
.mousedown()	자바스크립트의 "mousedown" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "mousedown" 이벤트를 발생시킴.
.mouseenter()	해당 요소 위로 마우스가 진입할 때 발생하는 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 마우스 진입 이벤트를 발생시킴.
.mouseleave()	해당 요소에서 마우스가 빠져나갈 때 발생하는 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 마우스가 빠져나가는 이벤트를 발생시킴.
.mousemove()	자바스크립트의 "mousemove" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "mousemove" 이벤트를 발생시킴.
.mouseout()	자바스크립트의 "mouseout" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "mouseout" 이벤트를 발생시킴.
.mouseover()	자바스크립트의 "mouseover" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "mouseover" 이벤트를 발생시킴.

메소드	설명
.mouseup()	자바스크립트의 "mouseup" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "mouseup" 이벤트를 발생시킴.

## ② 키보드 이벤트 메서드

메소드	설명
.keydown()	자바스크립트의 "keydown" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "keydown" 이벤트를 발생시킴.
.keyup()	자바스크립트의 "keyup" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "keyup" 이벤트를 발생시킴.
.keypress()	자바스크립트의 "keypress" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "keypress" 이벤트를 발생시킴.

## ③ 입력양식과 관련된 이벤트 메서드

메소드	설명
.blur()	자바스크립트의 "blur" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "blur" 이벤트를 발생시킴.
.change()	자바스크립트의 "change" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "change" 이벤트를 발생시킴.
.select()	자바스크립트의 "select" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "select" 이벤트를 발생시킴.
.submit()	자바스크립트의 "submit" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "submit" 이벤트를 발생시킴.
.focus()	자바스크립트의 "focus" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "focus" 이벤트를 발생시킴.
.focusin()	"focusin" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "focusin" 이벤트를 발생시킴.
.focusout()	"focusout" 이벤트와 이벤트 핸들러를 연결하거나, 해당 요소에 "focusout" 이벤트를 발생시킴.

## 3. 요소 제어

### ① 요소의 추가

선택자	설명
-----	----

선택자	설명
.append()	선택한 요소의 마지막에 새로운 요소나 콘텐츠를 추가함.
.prepend()	선택한 요소의 처음에 새로운 요소나 콘텐츠를 추가함.
.appendTo()	선택한 요소를 해당 요소의 마지막에 삽입함.
.prependTo()	선택한 요소를 해당 요소의 처음에 삽입함.
.html()	해당 요소의 HTML 콘텐츠를 반환하거나 설정함.
.text()	해당 요소의 텍스트 콘텐츠를 반환하거나 설정함.
.before()	선택한 요소의 바로 앞에 새로운 요소나 콘텐츠를 추가함.
.after()	선택한 요소의 바로 뒤에 새로운 요소나 콘텐츠를 추가함.
.insertBefore()	선택한 요소를 해당 요소의 앞에 삽입함.
.insertAfter()	선택한 요소를 해당 요소의 뒤에 삽입함.

## ② 요소의 복제

선택자	설명
.clone()	선택한 요소를 복사하여 새로운 요소를 생성함.

## ③ 요소의 삭제

선택자	설명
.remove()	선택한 요소를 DOM 트리에서 삭제함. (삭제된 요소와 연관된 제이쿼리 데이터나 이벤트도 같이 삭제됨)
.detach()	선택한 요소를 DOM 트리에서 삭제함. (삭제된 요소와 연관된 제이쿼리 데이터나 이벤트는 유지됨)
.empty()	선택한 요소의 자식 요소를 모두 삭제함.
.unwrap()	선택한 요소의 부모 요소를 삭제함.

## 4. 요소의 설정

메소드	설명
.css()	css() 메소드는 선택한 요소 집합의 첫 번째 요소의 스타일 속성값을 반환하거나, 선택한 요소의 스타일 속성을 인수로 전달받은 값으로 설정함.

메소드	설명
<code>.attr()</code>	선택한 요소 집합의 첫 번째 요소의 지정된 속성(attribute)값을 반환하거나, 선택한 요소의 지정된 속성을 전달받은 값으로 설정함.
<code>.prop()</code>	선택한 요소 집합의 첫 번째 요소의 지정된 프로퍼티(property)값을 반환하거나, 선택한 요소의 지정된 프로퍼티를 전달받은 값으로 설정함.
<code>.removeAttr()</code>	선택한 요소에서 지정된 속성(attribute)을 제거함.
<code>.removeProp()</code>	선택한 요소에서 지정된 프로퍼티(property)를 제거함.

## 5. 클래스의 설정

메소드	설명
<code>.addClass()</code>	선택한 요소에 인수로 전달받은 클래스를 추가함.
<code>.removeClass()</code>	선택한 요소에서 인수로 전달받은 클래스를 제거함.
<code>.ToggleClass()</code>	선택한 요소에 클래스가 없으면 인수로 전달받은 클래스를 추가하고, 전달받은 클래스가 추가되어 있으면 제거함.
<code>.hasClass()</code>	인수로 전달받은 값이 선택한 요소의 클래스 이름과 일치하는지를 확인함.

## 6. AJAX

### 1) 형식

```
$.ajax({
    url: "파일", //dataType에 지정된 형식에 맞는 데이터를 지니고 있는 파일
    data:{ key:data, key:data ... },
    method:"GET", //GET | POST
    dataType: "text",/*json | text | html | xml | csv */
    success : function(jsonData) {
        jsonList=JSON.parse(jsonData);
    },
    error: function(request,status,error){
        console.log("code:"+request.status);
        console.log("message:"+request.responseText);
        console.log("error:"+error);});});
```

### ajax error code

Code	Description	의 미
100	Continue	Client는 Request를 Server는 Response를 계속적으로 해야한다는 것
101	Switching Protocols	Client의 요청에 따라 연결 프로토콜을 전환
200	OK	에러 없이 전송 성공
201	Created	POST 명령 실행 및 성공
202	Accepted	서버가 클라이언트 명령을 받음
203	Non-Authoritative Information	서버가 클라이언트 요구 중 일부만 전송
204	No Content	클라이언트 요구를 처리했으나 전송할 데이터가 없음
205	Reset Content	클라이언트 폼 전송후 폼 내용 삭제
206	Partial Content	
300	Multiple Choices	
301	Moved Permanently	



302	Moved Temporarily	요구한 데이터를 변경된 임시 URL에서 찾았음
303	See Other	요구한 데이터가 변경된 URL에 있음을 명시
304	Not Modified	요구한 데이터를 변경하지 않았기 때문에 문제가 있음
305	Use Proxy	
307	Temporary Redirect	
400	Bad Request	클라이언트의 잘못된 요청으로 처리할 수 없음
401	Unauthorized	클라이언트의 인증 실패
	401.1 - 권한 없음 : 로그인 실패	
	401.2 - 권한 없음 : 서버 구성으로 인한 로그인 실패	
	401.3 - 권한 없음 : 리소스의 ACL에 의해 액세스가 거부	
	401.4 - 권한 없음 : 필터에 의해 권한 부여가 거부	
	401.5 - 권한 없음 : ISAPI 또는 CGI 응용 프로그램으로 권한 부여가 실패	
403	Forbidden	접근이 거부된 문서를 요청함
	403.1 - 액세스 금지 : 액세스 실행 금지	
	403.2 - 액세스 금지 : 읽기 액세스 금지	
	403.3 - 액세스 금지 : 쓰기 액세스 금지	
	403.4 - 액세스 금지 : SSL 필요	
	403.5 - 액세스 금지 : SSL 128 필요	
	403.6 - 액세스 금지 : IP 주소 거부	
	403.7 - 액세스 금지 : 클라이언트 인증서 필요	
	403.8 - 액세스 금지 : 사이트 액세스 거부	
	403.9 - 액세스 금지 : 연결된 사용자가 너무 많음	
	403.10 - 액세스 금지 : 잘못된 구성	
	403.11 - 액세스 금지 : 암호 변경	
	403.12 - 액세스 금지 : 매퍼가 액세스를 거부	
	403.13 - 액세스 금지 : 클라이언트 인증서 해지	
	403.15 - 액세스 금지 : 클라이언트 액세스 라이선스 초과	
	403.16 - 액세스 금지 : 클라이언트 인증서가 유효하지 않거나 신뢰할 수 없음	
	403.17 - 액세스 금지 : 클라이언트 인증서가 만료되었거나 아직 유효하지 않음	
404	Not Found	문서를 찾을 수 없음
	404.1 - 웹 사이트를 찾을 수 없음	
405	Method Not Allowed	리소스를 허용안함
406	Not Acceptable	허용할 수 없음
407	Proxy Authentication Required	프록시 인증 필요
408	Request Timeout	요청시간이 지남

409	Conflict	리소스간 충돌로 인하여 서버가 응답할 수 없음
410	Gone	요청된 리소스를 영구적으로 사용할 수 없음
411	Length Required	Content-Length를 정의하지 않고 리소스를 요청하였음
412	Precondition Failed	전제조건 실패
413	Request Entity Too Large	Request Entity가 서버가 처리할 수 있는 것보다 큼
414	Request-URI Too Long	URI가 너무 김
415	Unsupported Media Type	지원되지 않은 미디어 형식
416	Requested Range Not Satisfiable	
417	Expectation Failed	
500	Internal Server Error	내부 서버 오류(잘못된 스크립트 실행시)
	500-11 : 서버 종료 중	
	500-12 : 응용 프로그램 다시 시작	
	500-13 : 서버 사용량이 많음	
	500-14 : 잘못된 응용 프로그램	
	500-15 : global.asa에 대한 요청이 허용되지 않음	
501	Not Implemented	클라이언트에서 서버가 수행할 수 없는 행동을 요구함
502	Bad Gateway	서버의 과부하 상태
503	Service Unavailable	외부 서비스가 죽었거나 현재 멈춤 상태
504	Gateway Timeout	
505	HTTP Version Not Supported	

---

## 11. DOM







