



Seoul
Software
ACademy

웹 개발자 부트캠프 과정

SeSAC x CODINGOn

With. 팀 리쳐드

jQuery 란?

jQuery



자바스크립트 언어를 간편하게 사용할 수 있도록 단순
화시킨 오픈 소스 기반의 자바스크립트 라이브러리

라이브러리란?

자주 사용하는 기능들을 모아두었다가 필요할 때
가져다 쓸 수 있도록 한 자원들의 묶음

jQuery 장점

- 주요 웹 브라우저의 구버전을 포함한 대부분의 브라우저에서 지원 가능
- HTML DOM를 손쉽게 조작 가능하며, CSS 스타일도 간단히 적용 가능
- 애니메이션 효과나 대화형 처리를 간단하게 적용
- 같은 동작을 하더라도 더욱 짧게 구현 가능
- 오픈 라이선스를 적용해 누구나 자유롭게 이용 가능

jQuery 사용

1. 파일 다운로드
2. CDN(Content Delivery Network) 사용

<https://releases.jquery.com/>

Code Integration

```
<script
  src="https://code.jquery.com/jquery-3.6.0.min.js"
  integrity="sha256-/xUj+30JU5yExlq6GSYGSHk7tPXikynS7ogEvDej/m4="
  crossorigin="anonymous"></script>
```



The `integrity` and `corsorigin` attributes are used for [Subresource Integrity \(SRI\) checking](#). This allows browsers to ensure that resources hosted on third-party servers have not been tampered with. Use of SRI is recommended as a best-practice, whenever libraries are loaded from a third-party source. Read more at srihash.org

jQuery 기초

```
$ ( 선택자 ) . 동작함수 ( ) ;
```

\$ 는 jQuery() 함수의 축약 형태로, 식별자 역할

jQuery 메서드

값 가져오기 & 설정하기

- `$(선택자).val();`
 - Input 과 같은 value 값을 가져올 수 있다.

```
var value = $("input").val();
```

- `$(선택자).val("설정할 값");`
 - Input 과 같은 곳에 value 값을 설정할 수 있다.

```
$("input").val("설정완료!");
```


Style 변경하기

- `$(선택자).css(“속성” , “속성값”);`

```
$("#span").css("font-size", "28px");
$("#span").css("color", "blue");
```

```
var span = document.querySelector("span");
span.style = "font-size: 10px; color: red";
```

속성 변경하기

- `$(선택자).attr(“속성”, “속성값”);`

```
$("#li2").attr("style", "color: red");
```

Text 변경하기

- `$(선택자).text(“글자”);`

```
$("span").text("jQuery로 바꿨습니다.");
```

```
var span = document.querySelector("span");
span.innerText = "js로 바꿨습니다."
```

HTML 변경하기

- \$(선택자).html(html);

```
$("#p").html("<h1>jQuery<h1>");
```

```
var p = document.querySelector("p");  
p.innerHTML = "<h3>javascript</h3>"
```

요소 추가

- 선택된 요소의 자식 요소 중 마지막으로 추가
- \$(선택자).append(요소);

```
$("#p").append(h1);
```

```
p.appendChild(h1);
```

요소 추가

- `$(선택자).prepend(요소);`
 - 선택된 요소의 자식 요소 중 첫번째로 추가
- `$(선택자).before(요소);`
 - 선택된 요소의 형제 요소로 이전 형제로 추가
- `$(선택자).after(요소);`
 - 선택된 요소의 형제 요소로 다음 형제로 추가

요소 삭제

- `$(선택자).remove();`
 - 선택한 요소를 DOM 트리에서 삭제
 - 인자로 선택자를 넘겨주면, 선택한 요소 중 조건을 만족하는 요소만 삭제
- `$(선택자).empty();`
 - 선택한 요소의 자식 요소를 모두 삭제
 - 선택된 요소는 남아 있고 그 안이 다 비워진다

요소 탐색

- \$(선택자).parent().함수()
 - 바로 위에 존재하는 하나의 부모 요소 선택
- \$(선택자).parents().함수()
 - 바로 위 부모뿐만 아니라 부모의 부모, 모든 조상을 선택
- \$(선택자).next().함수()
 - 선택된 요소 바로 다음에 위치한 형제 요소 선택
- \$(선택자).prev().함수()
 - 선택된 요소 바로 이전에 위치한 형제 요소 선택
- \$(선택자).children().함수()
 - 선택된 요소의 자식 요소 모두 선택

클래스 조작하기

- `$(선택자).addClass("클래스명")`
 - 선택된 요소에 클래스 추가
- `$(선택자).removeClass("클래스명")`
 - 선택된 요소에 클래스 삭제
- `$(선택자).hasClass("클래스명")`
 - 선택된 요소에 클래스가 있는지 여부 확인
 - Return 값은 true 와 false 로 나온다.
- `$(선택자).toggleClass("클래스명")`
 - 선택된 요소에 클래스가 있으면 추가, 없으면 삭제

jQuery 이벤트 리스너

이벤트 리스너란?

- 이벤트가 발생했을 때 그 처리를 담당하는 함수
- Event listener = Event handler
- 이벤트
 - 웹 브라우저가 알려주는 HTML 요소에 대한 사건
 - Ex. 클릭, 스크롤, 마우스오버 등

로딩 이벤트

- `.ready()`
- `$()`

마우스 이벤트

- `click()`
- `mouseover()/mouseout()`
- `hover()`
- `scroll()`

키보드 이벤트

- `keydown()`
- `keyup()`

on 메서드

- `on(“이벤트 타입”, 핸들러)`

Js 이벤트 리스너

addEventListener

- 요소.addEventListener(이벤트 타입, 핸들러)
- 선택 요소에 지정한 이벤트가 발생하면, 약속 된 명령어를 실행시키는 메소

```
let boxEl = document.querySelector(".box");  
  
console.log(boxEl);  
  
boxEl.addEventListener("click", function() {  
    alert("click!");  
})
```

```
document.querySelector(".box").addEventListener("click", function() {  
    alert("click");  
})
```

addEventListener 이벤트의 종류

- Click : 클릭
- Mouse 계열
 - Mouseover : 요소에 커서를 올렸을 때
 - Mouseout : 마우스가 요소를 벗어날 때
 - Mousedown : 마우스 버튼을 누르고 있는 상태
 - Mouseup : 마우스 버튼을 떼는 순간
- Focus : 포커스가 갔을 때
- Blur : 포커스가 벗어나는 순간

addEventListener 이벤트의 종류

- Key 계열
 - Keypress : 키를 누르는 순간 + 누르고 있는 동안 계속 발생
 - Keydown : 키를 누르는 순간에만 발생
 - Keyup : 키를 눌렀다가 떼는 순간
- Load : 웹페이지에 필요한 모든 파일(html, css, js 등)의 다운로드가 완료 되었을 때
- Resize : 브라우저 창의 크기가 변경 될 때
- Scroll : 스크롤이 발생할 때
- Unload : 링크를 타고 이동하거나, 브라우저를 닫을 때
- Change : 폼 필드의 상태가 변경 되었을 때