

요소의 탐색과 생성

#01. 요소의 탐색

메서드 체인

- 하나의 jQuery요소에 대하여 메서드를 연속적으로 호출하는 기법.
- 특별한 경우가 아닌 이상 jQuery()의 함수들은 객체 자신을 리턴한다.

```
$("#foo").attr(key, value)
        .css(key, value)
        .addClass(cls)
        .on("click", () => {});
```

주변 요소 탐색하기

함수	설명
prev()	이전 요소를 리턴한다.
next()	다음 요소를 리턴한다.
parent()	상위 요소를 리턴한다.
children()	하위 요소(들)을 리턴한다.
eq(n)	n번째 요소를 리턴한다.

부모 요소 얻기

주어진 셀렉터를 갖는 가장 인접한 부모 요소를 리턴한다.

```
$("#foo").parents("셀렉터");
```

자식 요소 얻기

주어진 셀렉터를 갖는 가장 인접한 자식 요소를 리턴한다.

```
$("#foo").find("셀렉터");
```

#02. 요소의 동적 생성

<body>태그 안에 명시된 태그 요소를 객체화 하는 경우

태그 이름을 \$() 함수에 전달한다.

```
const obj = $("div");
```

<body>태그 안에 명시되지 않은 태그 요소를 객체화 하는 경우

태그 이름을 <>로 감싸서 \$() 함수에 전달한다.

```
const obj = $("<div>");
```

이렇게 생성된 요소들은 jQuery의 기능들을 적용할 수 있다.

```
const obj = $("div");  
obj.attr('id', 'hello').css(...).addClass('item');
```

기존에 존재하는 요소를 복사하는 경우

```
const obj = $("#foo").clone();  
obj.attr('id', 'helloworld');
```

모든 속성을 동일하게 복사하므로 id속성값 같은 경우(중복되어서는 안되므로)는 다른 값으로 변경해 주어야 한다.

동적으로 생성된 요소를 HTML 문서에 삽입하기

함수	설명
A.html(B)	A의 시작태그와 끝태그 사이의 내용을 B로 대체한다.
A.append(B)	A에 B를 추가한다. 기존의 내용을 유지하면서 맨 뒤에 추가한다.
B.appendTo(A)	B를 A에 추가한다. 기존의 내용을 유지하면서 맨 뒤에 추가한다.
A.prepend(B)	A에 B를 추가한다. 기존의 내용을 유지하면서 맨 앞에 추가한다.
B.prependTo(A)	B를 A에 추가한다. 기존의 내용을 유지하면서 맨 앞에 추가한다.
A.insertBefore(B)	A를 B의 직전에 삽입한다.
A.insertAfter(B)	A를 B의 직후에 삽입한다.
A.empty()	A의 모든 내용을 비운다.

동적으로 생성된 요소에 대한 이벤트 적용

기존의 이벤트 처리는 생성한 객체에 적용한다는 개념이지만 동적 요소에 대한 이벤트 처리는 앞으로 생성될 객체에 적용할 이벤트를 미리 준비해 둔다는 개념이다.

```
$(document).on('click', '#hello', function(e) {  
    e.preventDefault();  
    ...  
});
```

주의: 동적 생성 요소가 `<form>`인 경우에는 `e`객체를 사용해서는 안된다.

HTML요소의 위치 변경

`append()`, `prepend()`, `insertBefore()`, `insertAfter()` 함수등을 동적 요소가 아닌 기존에 존재하는 HTML요소끼리 사용하면 서로 위치를 바꿀 수 있다.

```
<div id="foo"></div>  
<div id="bar"></div>  
  
<script>  
// #foo의 위치가 #bar 다음으로 이동하게 된다.  
$("#foo").insertAfter("#bar");  
</script>
```