



## 제이쿼리란?

- 모질라의 자바스크립트 개발자인 존 레식(John Resig)이 만든 자바스크립트 라이브러리
  - 라이브러리 : 다양한 함수들의 집합
- 필요할 때마다 준비된 함수들을 꺼내서 쓴다!

## 제이쿼리의 장점

- 브라우저 간 호환성 문제 해결 (자바스크립트의 DOM, Event 등)
- 쉽고 편리한 애니메이션 효과 구현

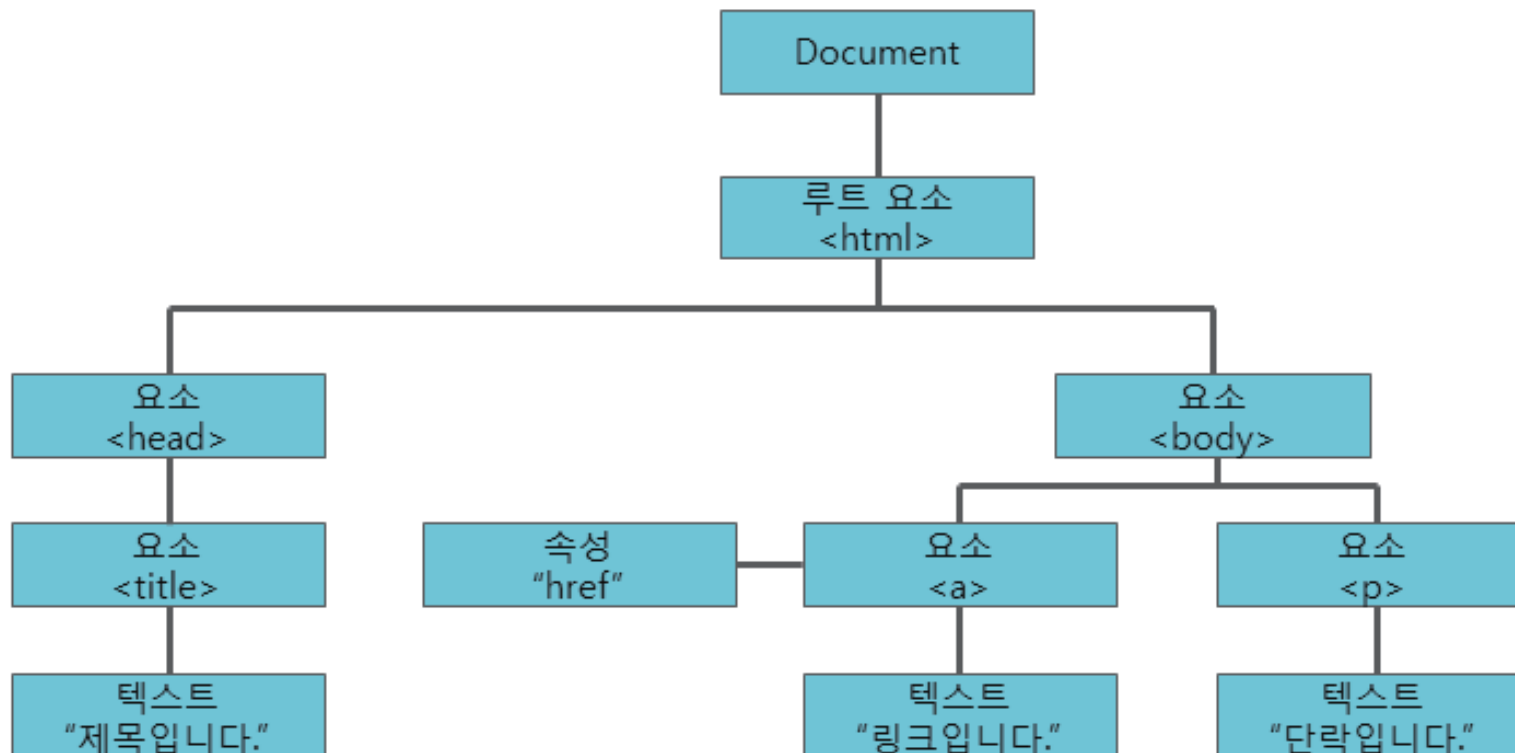
## 제이쿼리 라이브러리 연결

[www.jquery.com](http://www.jquery.com)

## DOM (Document Object Model)

웹 화면에 보이는 요소를 조작하기 위한 기능으로 가득 찬 라이브러리

- \* 노드는 HTML 웹페이지 구성 요소의 가장 작은 단위
- \* 태그뿐 아니라 주석까지도 모두 노드
- \* 주석과 텍스트 노드를 제외한 나머지 노드를 Element 노드라고 한다.



## 제이쿼리 기본 선택자

전체, 아이디/클래스, 요소, 그룹, 자손  
CSS 선택자와 동일

## 제이쿼리 인접관계 선택자

종류	사용법	설명
부모 요소 선택	<code>\$("요소선택").parent()</code>	선택한 요소의 부모 요소를 선택
상위 요소 선택	<code>\$("요소선택").parents()</code>	선택한 요소의 상위 요소를 모두 선택
자식 요소들 선택	<code>\$("요소선택").children()</code>	선택한 요소의 자식 요소들 모두 선택
형 요소 선택	<code>\$("요소선택").prev()</code>	선택한 요소의 바로 위 형 선택
형 요소 모두 선택	<code>\$("요소선택").prevAll()</code>	선택한 요소 기준 위 형들 모두 선택
동생 요소 선택	<code>\$("요소선택").next()</code>	선택한 요소 기준 아래 동생 선택
동생 요소 모두 선택	<code>\$("요소선택").nextAll()</code>	선택한 요소 기준 아래 동생들 모두 선택
형제들 모두 선택	<code>\$("요소선택").siblings()</code>	선택한 요소 기준 형제들 모두 선택

## 제이쿼리 탐색 선택자

기본 선택자로 선택한 요소 중 원하는 요소를 한 번 더 탐색  
CSS 선택자와 같거나 다른...

종류	사용법	설명	비고
<code>\$(".요소선택:first")</code> <code>\$(".요소선택").first()</code>	<code>\$(".li:first")</code> <code>\$(".li").first()</code>	전체 li 중에서 첫번째 li만 선택 <i>이것은 시작</i>	<code>\$(".li:last")</code> <code>\$(".li").last()</code>
<code>\$(".요소선택:odd")</code>	<code>\$(".li:odd")</code>	li 중에서 홀수번째 인덱스 요소만 선택	<code>\$(".li:even")</code>
<code>\$(".요소선택:nth-child(n)")</code>	<code>\$(".li:nth-child(3)")</code>	li 중에서 세번째 요소만 선택	<code>\$(".li:nth-child(2n)")</code>
<code>\$(".요소선택:eq(index)")</code> <code>\$(".요소선택").eq(index)</code>	<code>\$(".li:eq(2)")</code> <code>\$(".li").eq(2)</code>	li 중에서 두번째 인덱스 li만 선택	<code>\$(".li:nth-child(2n)")</code> <i>index()와 관계</i>
<code>\$(".요소선택").find(요소선택2)</code>	<code>\$(".div").find("strong")</code>	div 하위(자손) 요소인 strong 선택	<code>\$(".div").children("strong")</code>

*참이해! 6/1*

## 속성 조작 메서드 HTML 조작!

종류	사용법	설명	비고
<code>.html();</code> <code>.html("새 요소");</code>	<code>\$("#div").html();</code> <code>\$("#div").html("p");</code>	<div>div 요소의 하위 요소를 가져옵니다.</div> <div>div 요소의 하위 요소 모두 지우고 p요소 생성</div>	<code>.text();</code> <code>.text("새 텍스트");</code>
<code>.attr("속성명");</code> <code>.attr("속성명", "값");</code>	<code>\$("#img").attr("src");</code> <code>\$("#img").attr("src", "new.jpg");</code>	<div>img 요소의 src 속성 값을 가져옵니다.</div> <div>img 요소의 src속성에 new.jpg 값을 생성</div>	
<code>.removeAttr("속성명");</code>	<code>\$("#img").removeAttr("alt");</code>	img 요소의 alt속성을 제거	
<code>.addClass("class명");</code>	<code>\$("#div").addClass("box")</code>	div 요소에 box라는 클래스명 부여	<code>.removeClass("class명");</code>
<code>.toggleClass("class명");</code>	<code>\$("#div").toggleClass("box")</code>	div 요소에 box라는 클래스명이 없으면 부여, 있으면 제거	
<code>.hasClass("class명");</code>	<code>\$("#div").hasClass("active")</code>	div 요소에 active라는 클래스명이 있으면 true, 없으면 false	

이벤트 등록 메서드 이벤트 이벤트

종류	설명	비고
<code>.load()</code>	선택한 이미지 또는 프레임 요소에 연동된 소스의 로딩이 완료된 후 이벤트 발생	
<code>.ready()</code>	지정한 HTML문서 객체의 로딩이 완료된 후 이벤트 발생	
<code>.click()</code>	선택한 요소를 클릭했을 때	
<code>.mouseenter()</code>	선택한 <b>요소 범위</b> 에 마우스 포인터를 올렸을 때	<code>.mouseover()</code>
<code>.mouseleave()</code>	선택한 요소 범위에 마우스 포인터가 나갈 때	<code>.mouseout()</code>
<code>.hover()</code>	선택한 요소에 마우스 포인터를 올렸을 때, 내렸을 때 <b>각각의</b> 이벤트 발생	각각의 동작을 등록해준다.
<code>.scroll()</code>	스크롤바를 움직일 때	가로, 세로 스크롤 각각 구별
<code>.focus()</code>	선택한 요소에 포커스가 생성될 때	<code>.blur()</code>

## 효과 메서드

구분	종류
숨김	.hide();
	.fadeOut();
	.slideUp();
보임	.show();
	.fadeIn();
	.slideDown();
숨기거나 보이거나	.toggle();
	.fadeToggle();
	.slideToggle();

`$("#요소선택").효과메서드( 효과 소요시간, 가속도, 콜백함수 );`

```
$("#box").slideUp( 2000, "linear", function() {
    alert("창 다 올라 갔슈~");
} );
```