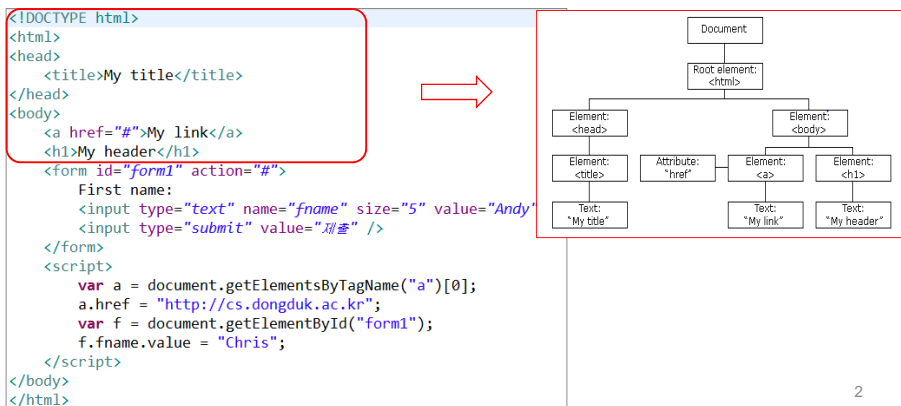


8. HTML DOM

HTML DOM 개요

- HTML 문서에 대한 표준 객체 모델 및 프로그래밍 인터페이스
 - HTML DOM Tree: 문서의 구성 요소들을 나타내는 객체들의 트리
 - 문서에 속한 엘리먼트와 속성들을 조회, 수정, 삭제, 추가 가능



2

Content

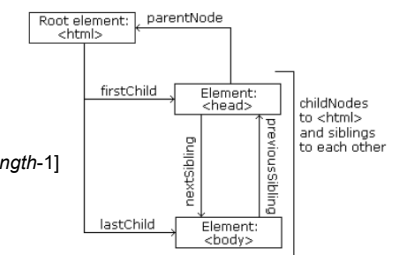
- HTML DOM 개요
- HTML DOM API
- HTML DOM 사용 예
- jQuery 개요
- jQuery Syntax
- Selectors
- Events & Effects
- DOM Manipulation
- jQuery 사용 예

1

HTML DOM 개요

Navigation of HTML DOM Tree

- Node에 정의된 속성(property) 이용
 - `parentNode`
 - `childNodes[index]`
 - `firstChild == childNodes[0]`
 - `lastChild == childNodes[childNodes.length-1]`
 - `nextSibling`
 - `previousSibling`



Node에 관한 정보를 나타내는 속성

- `nodeName` – 태그 이름(엘리먼트 노드), 속성 이름(속성 노드) 등
- `nodeValue` – 속성 값(속성 노드), 텍스트 문자열(텍스트 노드) 등
- `nodeType` – 엘리먼트 노드: 1, 속성 노드: 2, 텍스트 노드: 3, ...

```
<title id="demo">HTML DOM</title>
<script>
  document.getElementById("demo").firstChild.nodeValue; // "HTML DOM"
  document.getElementById("demo").childNodes[0].nodeValue; // "HTML DOM"
  document.getElementById("demo").innerHTML; // "HTML DOM"
</script>
```

HTML DOM에서 get 함수!

3

HTML DOM API

Document Object

- 특정 엘리먼트 찾기

Method	Description
document.getElementById(id)	ID로 특정 엘리먼트를 찾아 반환
document.getElementsByTagName(name)*	태그 이름으로 엘리먼트들을 찾아 반환
document.getElementsByClassName(name)*	클래스 이름으로 엘리먼트들을 찾아 반환

* 배열과 유사한 HTMLCollection 타입의 객체를 반환 (인덱스를 이용해서 원소 접근)

- 특정 엘리먼트 및 텍스트 노드 생성

Method	Description
document.createElement(name)	새로운 엘리먼트 노드 생성
document.createTextNode(text)	새로운 텍스트 노드 생성
document.write(text / object)	HTML 문서에 텍스트 스트림을 출력

4

HTML DOM API

Document Object

- HTML Object 및 Object Collection들을 찾음

- document.documentElement - 루트 엘리먼트(<html>) 노드 반환
- document.head / body / title - <head>, <body>, <title> 엘리먼트 노드 반환
- document.forms - 모든 <form> 엘리먼트 노드들을 반환
- document.images - 모든 엘리먼트 노드들을 반환
- document.anchors - name 속성을 갖는 모든 <a> 엘리먼트 노드들을 반환
- document.links - href 속성을 갖는 모든 <a> 또는 <area> 엘리먼트 노드들을 반환
- document.URL - 문서의 URL을 반환
- ...

6

HTML DOM API

Document Object

- 예: ID와 태그 이름으로 엘리먼트 찾기

```
<!DOCTYPE html>
<html>
<head>
  <title>My title</title>
</head>
<body>
  <a href="#">My link</a>
  <h1>My header</h1>
  <form id="form1" action="#">
    First name:
    <input type="text" name="fname" size="5" value="Andy" />
    <input type="submit" value="제출" />
  </form>
  <script>
    var a = document.getElementsByTagName("a")[0];
    a.href = "http://cs.dongduk.ac.kr";
    var f = document.getElementById("form1");
    f.fname.value = "Chris";
  </script>
</body>
</html>
```



5

HTML DOM API

Element Object

- 자식 엘리먼트 추가, 삭제, 교체

Method	Description
element.appendChild(element)	element를 마지막 자식으로 추가
element.insertBefore(element, refChild)	element를 refChild 자식 앞에 추가
element.removeChild(element)	자식 엘리먼트인 element를 삭제
element.replaceChild(new, old)	자식 엘리먼트인 old를 새로운 엘리먼트 new로 교체

7

HTML DOM API

□ Element Object

- Content를 텍스트 또는 엘리먼트들로 변경
 - `element.innerHTML = new-content` (문자열; 자식 엘리먼트 포함 가능)
- 속성 값 변경
 - `element.attribute-name = attribute-value`
- 예

```
<p id="p1">Hello World!</p>


<script>
document.getElementById("p1").innerHTML = "Hello <b>New<b> World!";
document.getElementById("myImage").src = "landscape.jpg";
</script>
```

속성 값이 변경됨
content가 변경됨

8

HTML DOM API

□ Element Object

- Event Handler 지정
 - 엘리먼트에 대한 이벤트를 처리하는 함수를 정의 및 할당
 - ✓ `document.getElementById(id).이벤트명 = function() { 이벤트 처리 코드 ... }`
 - ✓ `document.getElementById(id).이벤트명 = function-name; // 별도로 정의된 함수 지정`
 - 이벤트 종류 (예)
 - ✓ onclick – 엘리먼트를 click할 때 발생
 - ✓ onchange – form 구성요소의 내용이 변경될 때 발생
 - <input>, <select>, <textarea> 등
 - ✓ onfocus / onblur – 엘리먼트가 focus를 얻거나 잃을 때 발생
 - ✓ onmouseover / onmouseout – 엘리먼트에 마우스가 올라가거나 벗어날 때 발생
 - ✓ onsubmit – form 입력을 전송(submit) 할 때 발생

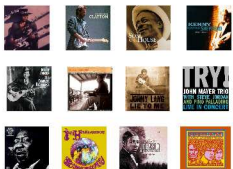
```
<button id="myBtn">Try it</button>
<p id="demo"></p>
<script>
function displayDate() {
    document.getElementById("demo").innerHTML = Date();
}
document.getElementById("myBtn").onclick = displayDate;
</script>
```

9

HTML DOM 사용 예

□ 예: Top-5 CD List

Click on a CD cover to add it to the Top 5 list. If you want to start over, click the "Start Over" button to clear the Top 5 list.



My Top 5 CDs

Start Over

Click on a CD cover to add it to the Top 5 list. If you want to start over, click the "Start Over" button to clear the Top 5 list.



My Top 5 CDs



Start Over

HTML DOM 사용 예

```
<html>
<head><title>Top 5 CD Recommendations</title>
<link rel="stylesheet" type="text/css" href="top5.css" />
<script type="text/javascript" src="top5.js"> </script>
</head>
<body onLoad="addOnClickHandlers();">
<div id="instructions">
    Click on a CD cover to add it to the Top 5 list. If you want to
    start over, click the "Start Over" button to clear the Top 5 list.
</div>
<div id="cds">
    
    
    
    
    ...
</div>
<div id="top5-listings">
    <h2>My Top 5 CDs</h2>
    <div id="top5"></div>
</div>
<form><input type="button" value="Start Over" onClick="startOver();" />
</form>
</body></html>
```

top5.html

선택 시 아래로 이동

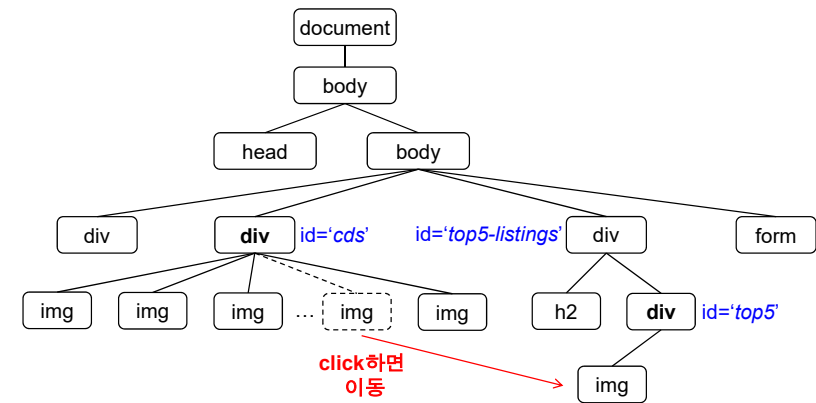
HTML DOM 사용 예

- CD 리스트에서 두 번째와 세 번째 이미지 선택 시 해당되는 `` 엘리먼트들이 `<div id="top5">`의 자식으로 이동해야 함

```
<div id="instructions">
  Click on a CD cover to add it to the Top 5 list. If you want to
  start over, click the "Start Over" button to clear the Top 5 list.
</div>
<div id="cds">
  
  
  ...
</div>
<div id="top5-listings">
  <h2>My Top 5 CDs</h2>
  <div id="top5">
    
    
  </div>
</div>
...
```

HTML DOM 사용 예

- DOM tree



13

HTML DOM 사용 예

```
function addOnClickHandlers() {
  var cdsDiv = document.getElementById("cds");
  var cdImages = cdsDiv.getElementsByTagName("img");
  for (var i=0; i<cdImages.length; i++) // CD 리스트의 각 <img> 마다
    cdImages[i].onclick = addToTop5; // onclick event handler 추가
}
function addToTop5() {
  // 각 <img> 클릭 시 호출됨
  var imgElement = this; // 이 함수가 매핑된 <img> 엘리먼트
  var top5Element = document.getElementById("top5");
  top5Element.appendChild(imgElement); // <img>가 <top5>의 자식으로 변경됨
  imgElement.onclick = null; // onclick event handler 제거
}
function startOver() {
  // Start Over 버튼 클릭 시 호출됨
  var top5Element = document.getElementById("top5");
  var cdsElement = document.getElementById("cds");
  while (top5Element.hasChildNodes()) {
    var firstChild = top5Element.firstChild; // <top5> 밑의 각 <img>를
    cdsElement.appendChild(firstChild); // <cds>의 자식으로 변경
  }
  addOnClickHandlers();
}
```

top5.js

removeChild가
생략됨

HTML DOM 사용 예

- 선택된 CD에 순위(rank) 값 추가

```
function addToTop5() {
  var imgElement = this; // 선택된 <img> 엘리먼트
  var top5Element = document.getElementById("top5"); // <top5> 엘리먼트

  var numCDs = 0;
  for (var i=0; i<top5Element.childNodes.length; i++) {
    if (top5Element.childNodes[i].nodeName.toLowerCase() == "img")
      numCDs = numCDs + 1; // <top5> 밑의 <img>들의 개수를 count
  }

  top5Element.appendChild(imgElement); // 선택된 <img>가 <top5>의 자식이 됨
  imgElement.onclick = null;

  var newSpanElement = document.createElement("span"); // <span> 생성
  newSpanElement.className = "rank"; // class 속성 추가
  var newTextElement = document.createTextNode(numCDs + 1);
  newSpanElement.appendChild(newTextElement); // <span>의 텍스트 값 생성
  top5Element.insertBefore(newSpanElement, imgElement); // <img> 앞에 추가
}
```

HTML DOM 사용 예

- 선택된 CD에 순위(rank) 값 추가

```
<div id="top5">
  <span class="rank">1</span>
  
  <span class="rank">2</span>
  
  ...
</div>
```



추가된 엘리먼트들



My Top 5 CDs

HTML DOM 사용 예

- CD를 최대 5개만 선택 가능하도록 제한

```
function addToTop5() {
  var imgElement = this;
  var top5Element = document.getElementById("top5");
  var numCDs = 0;

  for (var i=0; i<top5Element.childNodes.length; i++) {
    if (top5Element.childNodes[i].nodeName.toLowerCase() == "img")
      numCDs = numCDs + 1;          // <top5> 밑의 <img>들의 개수를 count
  }

  if (numCDs >= 5) {
    alert("You already have 5 CDs. Click \"Start Over\" to try again.");
    return;
  }

  top5Element.appendChild(imgElement);
  imgElement.onclick = null;
  ...
}
```

HTML DOM 사용 예

- Top 5 CD들을 모두 이동 시 각 CD의 순위 값들은 삭제

```
function startOver() {
  var top5Element = document.getElementById("top5");
  var cdsElement = document.getElementById("cds");
  while (top5Element.hasChildNodes()) {
    var firstChild = top5Element.firstChild;    // <top5>의 각 자식 노드
    if (firstChild.nodeName.toLowerCase() == "img") {
      cdsElement.appendChild(firstChild);        // <img>는 이동
    } else {
      top5Element.removeChild(firstChild);      // <span>은 삭제
    }
  }
  addOnClickHandlers();
}
```

jQuery 개요

- A lightweight, "write less, do more", JavaScript library
- HTML DOM 조작과 이벤트 처리, Animation, Ajax 등을 효율적으로 구현 가능
- JavaScript의 브라우저 호환성 문제를 해결
- 지원 기능(API)
 - HTML DOM manipulation
 - CSS manipulation
 - HTML event methods
 - Effects & animations
 - Ajax
 - 기타 Utilities

사용 예:

```
$(document).ready( function() {
  $("p").click( function() {
    $(this).hide();
  });
});
```

jQuery 개요

- HTML 문서에서 jQuery Library 이용 방법
 - jQuery library는 하나의 JavaScript file로 구성됨
 - jQuery.com에서 직접 file을 다운로드 받거나, Google, Microsoft 등에서 CDN을 통해 제공하는 file의 URL을 이용
 - HTML 문서에서 <script src="..."> 로 지정

```
<head>
  <script src="jquery-3.6.0.min.js"></script>
  또는
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js">
  </script>
</head>
```

20

jQuery Syntax

- **\$(selector).action()**
 - HTML 문서에 포함된 element들을 선택(select)하고, 그것들에 대해 어떤 동작(action)을 수행함
 - \$: jQuery API 사용
 - (selector) : HTML element들을 검색(선택)
 - action() : element들에 대해 action을 실행
 - 예:
 - \$("p").hide() : 모든 <p> element들을 찾아 숨김(hide)
 - \$("#test").hide() : id="test"인 특정 element를 찾아 숨김(hide)

21

jQuery Syntax

- Document Ready Event *랜더링 직후에 실행*
 - **\$(document).ready(function() {** // 또는 **\$(function() {**
... // jQuery methods go here
// 예:
// \$("p").click(function() { // <p>가 click될 때
// \$(this).hide(); // <p>를 숨김
// });
});
 - HTML 문서가 브라우저에 loading이 완료되어 이용가능한 상태가 된 후에 주어진 function의 code를 실행함
 - jQuery code가 미리 실행될 경우 오류 발생 가능
 - 일반적으로 <head> section 내에 작성

ready 함수

22

Selectors

- HTML element들을 검색 및 선택하는 방법을 표현
 - HTML element를 이름, ID, class, 종류, 속성 등을 통해 검색(선택)함
 - CSS에서 사용되는 selector를 기반으로 확장해서 정의됨
- 주의: jQuery의 모든 selector들은 **\$(...)** 와 함께 사용해야 함

23

Selectors

Element Selector

- `$("p")` : HTML page 내의 모든 `<p>` element들을 선택
- `document.getElementsByTagName("p")`와 동일 (HTML DOM API)
- 예

for문 필요 (X) ←
모든 버튼이 찾아감

```
$(document).ready( function() {  
    $("button").click( function() {  
        $("p").hide();  
    });  
});
```

#id Selector

- `$("#test")` : id="test"인 특정 element를 선택
- `document.getElementById("test")`와 동일 (HTML DOM)

.class Selector

- `$(".test")` : 특정 CSS class가 적용된(즉, class="test") element들을 선택
- `document.getElementsByClassName("test")`와 동일 (HTML DOM)

24

Selectors

기타

- `$("")` : 모든 element들을 선택
- `$(this)` : 현재의 element를 선택
- `$("p.intro")` : class="intro"를 포함하는 모든 `<p>` element들을 선택
- `$("p:first")` : 첫 번째 `<p>` element를 선택
- `$("p:first-child")` : 첫 번째 자식인 모든 `<p>` element들을 선택
- `$("div > p")` : `<div>` element의 자식인 모든 `<p>` element들을 선택
- `$("div p")` : `<div>` element의 자손인 모든 `<p>` element들을 선택
- `$("h1, div, p")` : 모든 `<h1>`, `<div>`, `<p>` element들을 선택
- `$("[href])` : href 속성을 갖고 있는 모든 element들을 선택
- `$("[href=default.htm])` : href 속성을 갖고 그 값이 "default.htm"인 모든 element들을 선택
- `$("input")` : 모든 `<input>` element들을 선택
- `$(":text")` : type="text"인 모든 `<input>` element들을 선택
- `$(":checkbox")` : type="checkbox"인 모든 `<input>` element들을 선택
- `$(":text")` : type="text"인 모든 `<input>` element들을 선택
- `$("tr:even")` : 짝수 번째 `<tr>` element들을 모두 선택
- `$("tr:odd")` : 홀수 번째 `<tr>` element들을 모두 선택
- ...

25

Events & Effects

대부분의 HTML event에 대해 대응되는 jQuery function이 정의됨

- `$(document).ready()`
- `click()`, `dblclick()`, `mouseenter()`, `mouseleave()`, `mousedown()`, `mouseup()`, `hover()` : 마우스와 관련된 events
 - 예:

```
$( "p" ).click( function() {  
    // click event에 대한 handler function 지정  
    $(this).hide();  
    // action goes here...  
});
```
- `focus()`, `blur()` : HTML form field에 대해 event handler function 정의
 - 예:

```
$( "input" ).focus( function() {  
    $(this).css("background-color", "#cccccc");  
});
```

26

Events & Effects

Effects

- `hide()`, `show()`, `toggle()` : HTML element를 숨기거나 보임
 - `$("#hide").click(function() {
 $("#p1").hide(); // element.style.display = "none";
 $("#p2").show(); // element.style.display = "";
 $("#p3").toggle();
});`
- `fadeIn()`, `fadeOut()`, `fadeToggle()`, `fadeTo()` : element를 fade-in / fade-out 처리
 - CSS로도 가능
`$("button").click(function() {
 $("#div1").fadeIn();
 $("#div2").fadeOut("slow"); // speed (duration)
 $("#div3").fadeToggle(3000); // 3000 milliseconds
 $("#div4").fadeTo("slow", 0.4); // 투명도 0.4까지 (0~1)
});`
- `slideDown()`, `slideUp()`, `slideToggle()` : sliding effect 정의
- `animate()` : 다양한 animation 효과 정의

27

DOM Manipulation

- HTML DOM tree의 element 및 attribute들을 변경하고 조작하기 위한 다양한 method들을 제공
 - Get/Set content: `text()`, `html()`, `val()`
 - Get/Set attributes: `attr()`
 - Add new content: `append()`, `appendTo()`, `prepend()`, `prependTo()`, `insertBefore()`, `before()`, `insertAfter()`, `after()`
 - Remove elements/content: `remove()`, `empty()`
 - Get/Set CSS classes: `addClass()`, `removeClass()`, `toggleClass()`, `css()`

28

Get/Set Content

- `text()` : 선택된 element의 text content를 읽거나 변경 → 문자열만
 - `element.textContent` 와 동일 (HTML DOM API)
- `html()` : 선택된 element의 content를 읽거나 변경 → 자식 엘리먼트 포함
 - HTML markup 포함 가능
 - `element.innerHTML` 와 동일 (HTML DOM API)
- `val()` : form field의 값을 읽거나 변경 → value값
 - `element.value` 와 동일 (HTML DOM API)
- 예:

```
$("#btn1").click(function() {  
    alert("Text: " + $("#para1").text()); // 읽기  
    alert("HTML: " + $("#div2").html());  
    alert("Value: " + $("#input3").val());  
  
    $("#para1").text("Hello world!"); // 변경  
    $("#div2").html("<b>Hello world!</b>");  
    $("#input3").val("Dolly Duck");  
});
```

29

Get/Set Attributes

- `attr()` : attribute의 값을 읽거나 변경
 - `element.attribute-name` 과 동일 (HTML DOM)

□ 예:

```
$("#btn1").click(function() {  
    alert( $("#a1").attr("href") ); // 읽기  
    $("#a1").attr("href", "http://cs.dongduk.ac.kr/"); // 변경  
});
```

30

Add New Elements/Content

- `append()` : element/content를 선택된 element의 마지막 자식으로 추가
 - `parent.appendChild(newChild)`와 동일 (HTML DOM API)
 - 예
 - var txt1 = "<p>Text.</p>";
 \$("#body").append(txt1); // append the new element <p>
 - var txt1 = document.createElement("p"); // create <p> by HTML DOM
 txt1.innerHTML = "Text."; // 또는 txt1.text("Text.") → <p>Text.</p>
 \$("#body").append(txt1); // append the new element <p>
 - `A.append(B) == B.appendTo(A)`
- `prepend()` : element/content를 선택된 element의 첫번째 자식으로 추가
 - `A.prepend(B) == B.prependTo(A)`

31

Add New Elements/Content

- **before()** : element/content를 선택된 element의 이전 형제로 추가

- 예

```
var txt1 = "<b>jQuery!</b>";  
$("#img1").before(txt1); // Insert new element before <img>  
  
var txt1 = document.createElement("b"); // create <b> by HTML DOM  
txt1.text("jQuery!"); // <b>jQuery!</b>  
$("#img1").before(txt1); // Insert new element before <img>  
  
var img1 = document.getElementById("img1");  
img1.parentNode.insertBefore(txt1, img1); // HTML DOM API
```

- A.before(B) == B.insertBefore(A)

형제 관계 (부모 밑에 됨!)

- **after()** : element/content를 선택된 element의 다음 형제로 추가

- A.after(B) == B.insertAfter(A)

32

Remove Elements/Content

- **remove()** : 선택된 element와 그것의 content(자식 노드들)을 삭제
- **detach()** : remove()와 동일하나, 삭제된 element를 유지하고 나중에 다시 삽입 가능
- **empty()** : 선택된 element의 content(자식 노드들)을 삭제 → 자원은 남기고 자식만 삭제
 - element 자신과 속성들은 유지하되 empty element로 만들

□ 예:

```
$("#button").click( function(){  
    → $("#p").remove();  
    var divs = $("#div").detach();  
    $("#span").empty();  
});
```

선택된 element를 선택해서
대상을 선택한 후
삭제해야 함!

33

Traversing DOM Tree

- **parent(filter)** : 선택된 element의 부모 element를 반환
 - element.parentNode와 동일 (HTML DOM API)
- **parents(filter)** : 선택된 element의 모든 조상 element들을 반환
- **children(filter)** : 선택된 element의 모든 자식 element들을 반환
 - element.childNodes와 동일 (HTML DOM API)
- **find(filter)** : 선택된 element의 모든 자손들을 선택
 - filter가 element의 이름일 경우,
element.getElementsByTagName(filter)와 동일 (HTML DOM API)
- **siblings(filter)** : 선택된 element의 모든 형제 element들을 반환
- **next(filter), nextAll(filter)** : 선택된 element의 다음 형제 element(들)을 반환
 - next()는 element.nextSibling와 동일 (HTML DOM API)
- **prev(filter), prevAll(filter)** : 선택된 element의 이전 형제 element(들)을 반환
 - prev()는 element.previousSibling과 동일 (HTML DOM API)

34

Filtering Elements

- **first() / last()** : 선택된 element들 중 첫 번째/마지막 element를 반환
 - 예: \$("#div p").first().css(...); // <div> 내의 첫 번째 <p>를 선택하여 css 스타일 적용
- **eq(index)** : 선택된 element들 중 주어진 index(순서)에 해당하는 것을 반환 (index는 0부터 시작)
 - 예: \$("p").eq(1).css(...); // <p>들 중 두 번째 것을 선택
- **filter(criteria)** : 주어진 조건을 만족하는 것들만 선택하여 반환
 - 예: \$("p").filter(".intro"); // <p>들 중 class가 "intro"인 것들만 선택
- **not(criteria)** : 주어진 조건을 만족하지 않는 것들만 선택하여 반환
 - 예: \$("p").not(".intro"); // <p>들 중 class가 "intro"가 아닌 것들만 선택
- **each(function)** : 선택된 element들 각각에 대해 같은 작업을 반복 수행

```
- 예: $("#div1").find("a").each( function() {  
    ... $(this).text() ... $(this).attr(href) ... // this는 <a>를 가리킴  
});
```

for문처럼 동작

이항연산자를 의미!

35

jQuery 사용 예

□ Top-5 CD List

Click on a CD cover to add it to the Top 5 list. If you want to start over, click the "Start Over" button to clear the Top 5 list.



My Top 5 CDs

Start Over



Click on a CD cover to add it to the Top 5 list. If you want to start over, click the "Start Over" button to clear the Top 5 list.



My Top 5 CDs



Start Over

36

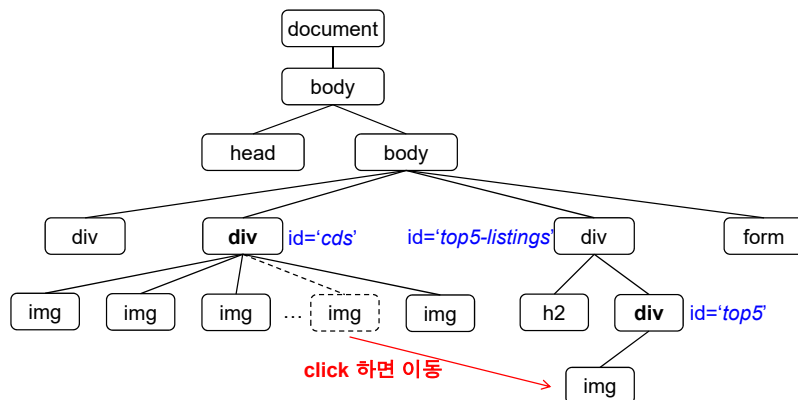
jQuery 사용 예

```
<html>
<head><title>Top 5 CD Recommendations</title>
<link rel="stylesheet" type="text/css" href="top5.css" />
<script src=
  "https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js">
</script>
<script type="text/javascript" src="top5.js"> </script>
</head>
<body>                                <!-- onLoad="..." 생략 -->
  <div id="instructions">
    Click on a CD cover to add it to the Top 5 list. ...
  </div>
  <div id="cds">
    
    
    ...
  </div>
  <div id="top5-listings">
    <h2>My Top 5 CDs</h2>
    <div id="top5"></div>
  </div>
  <form><input type="button" value="Start Over" /> <!-- onClick 생략 -->
</form>
</body></html>
```

top5.html

jQuery 사용 예

□ DOM tree



38

jQuery 사용 예

□ top5.js

```
/*
function addOnClickHandlers() {
  var cdsDiv = document.getElementById("cds");
  var cdImages = cdsDiv.getElementsByTagName("img");
  for (var i=0; i<cdImages.length; i++) {
    cdImages[i].onclick = addToTop5;
  }
}
*/
$(document).ready(function(){
  $("#cds").find("img").click(addToTop5);
  // 또는 $("#cds img").click(addToTop5);

  $(".button").click(startOver);
  // button 타입 <input>에 onclick handler로 startOver 함수 지정
});
```

39

jQuery 사용 예

```
function addToTop5() {
    var imgElement = this;
    /*
    var top5Element = document.getElementById("top5");
    var numCDs = 0;
    for (var i=0; i<top5Element.childNodes.length; i++) {
        if (top5Element.childNodes[i].nodeName.toLowerCase() == "img")
            numCDs = numCDs + 1;
    }
    */
    var numCDs = $("#top5").children("img").length;

    if (numCDs >= 5) {
        alert("You already have 5 CDs. Click \"Start Over\" to try again.");
        return;
    }
}
(계속)
```

40

jQuery 사용 예

```
/*
top5Element.appendChild(imgElement);
imgElement.onclick = null;

var newSpanElement = document.createElement("span");
newSpanElement.className = "rank";
var newText = document.createTextNode(numCDs + 1);
newSpanElement.appendChild(newText);
top5Element.insertBefore(newSpanElement, imgElement);
*/
$("#top5").append(imgElement);
$(imgElement).off("click"); // click event handler를 제거

var newSpanElement = "<span></span>";
$(newSpanElement).addClass("rank")
                    .text(numCDs + 1)
                    .insertBefore(imgElement);
// 또는 $(imgElement).before(newSpanElement);
}

```

→ span이 imgElement의 형제요소로 앞에 추가됨

41

jQuery 사용 예

```
function startOver() {
    /*
    var top5Element = document.getElementById("top5");
    var cdsElement = document.getElementById("cds");
    while (top5Element.hasChildNodes()) {
        var firstChild = top5Element.firstChild;
        if (firstChild.nodeName.toLowerCase() == "img")
            cdsElement.appendChild(firstChild);
        else
            top5Element.removeChild(firstChild);
    }
    addOnClickHandlers();
    */
    $("#top5").children("img").each( function() {
        $(this).appendTo("#cds"); // this: <img>
        $(this).on("click", addToTop5); // event handler 추가
    });
    $("#top5").children().remove();
}
```

42

References

Books

- Ryan Benedetti/한성용 역, Head First jQuery, 한빛미디어, 2012.
- 정인용, 자바스크립트+제이쿼리 입문, 이지스퍼블리싱, 2014.

Manual & Tutorials

- jQuery homepage: <https://jquery.com/>
- jQuery API documentation: <https://api.jquery.com/>
- jQuery tutorial: <https://www.w3schools.com/jquery/default.asp>

43