

문서 객체 모델(DOM)



소프트웨어학부
진혜진



목차

1. 문서 객체 모델
2. DOM 요소에 접근하기
3. 웹 요소의 속성 가져와서 수정하기
4. DOM에서 이벤트 처리하기
5. 웹 요소 내용과 스타일 가져와서 수정하기
6. DOM에 요소 추가하기
7. 추가한 노드 순서 바꾸거나 삭제하기

3. 웹 요소의 속성 가져와서 수정하기

3

- 속성 노드에 접근하면 HTML 요소의 속성 값을 원하는 값으로 수정할 수 있음
- `getAttribute()` : 속성에 접근할 때
- `setAttribute()` : 접근한 속성의 값을 바꿀 때

```
<body>
  <div id="container">
    <h1 id="heading"></h1>
    <div id="c-img">
      
    </div>
  </div>
</body>
```



Elements Console Sources Network

top Filter

```
> document.querySelector("img").getAttribute("src")  
< "images/boy.png"  
  
>
```



Elements Console Sources Network Performance Memory

top Filter Default levels

```
> document.querySelector("img").getAttribute("src")  
< "images/boy.png"  
  
> document.querySelector("img").setAttribute("src","images/girl.png")  
< undefined  
  
> |
```

```
<div id="c-img">  
    
</div>
```

3. 웹 요소의 속성 가져와서 수정하기

5



```
Elements Console Sources Network Performance Memory >>
top Filter Default levels ▼
> document.querySelector("#c-img > img").getAttribute("src")
< "images/boy.png"
> document.querySelector("#c-img > img").setAttribute("src","images/girl.png")
< undefined
> |
```

```
<div id="c-img">
  
</div>
```

3. 웹 요소의 속성 가져와서 수정하기

6

□ 선택한 이미지 표시하기

▣ 태그 속성을 사용해 상품 이미지 변경하기

NETWORK



4차 산업혁명과 함께하는 네트워크
네트워크 개론
Easy Learning Network
김영사

상품명 : 네트워크개론
판매가 : 24,000원
배송비 : 2,500원
(20,000원 이상 구매 시 무료)
적립금 : 2400원(10%)

장바구니 담기



NETWORK



네트워크 개론
쉽게 배우는 네트워크의 기본 원리
김영사

상품명 : 네트워크개론
판매가 : 24,000원
배송비 : 2,500원
(20,000원 이상 구매 시 무료)
적립금 : 2400원(10%)

장바구니 담기



3. 웹 요소의 속성 가져와서 수정하기

```
<!DOCTYPE html>
<head>
  <link rel="stylesheet" href="product.css">
</head>
<body>
  <div id="container">
    <h1 id="heading">NETWORK</h1>
    <div id="prod-pic">
      
      <div id="smallbook">
        
        
      </div>
    </div>
    <div id="desc">
      <ul>
        <li>상품명 : 네트워크개론</li>
        <li class="price">판매가 : 24,000원</li>
        <li>배송비 : 2,500원<br>(20,000원 이상 구매 시 무료)</li>
        <li>적립금 : 2400원(10%)</li>
        <button>장바구니 담기</button>
      </ul>
    </div>
  </div>
  <script src="showBig-result.js"></script>
</body>
</html>
```

3. 웹 요소의 속성 가져와서 수정하기

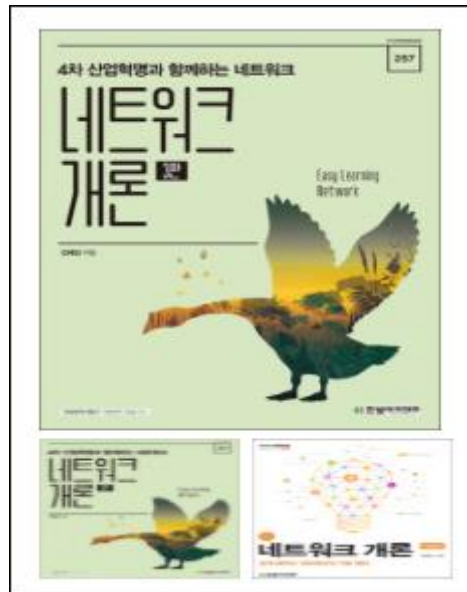
8

- 이미지 요소 접근 방법
 - ▣ 큰 이미지 : `id="book"`
 - ▣ 작은 이미지 : `class="small"`
 - ▣ 큰 이미지는 `id` 값, 작은 이미지는 `class` 값을 사용해 접근하면 됨
- 접근한 이미지의 `src` 속성에 접근해서 그 값을 바꾸는 방법
 - ▣ 큰 이미지 요소의 `src` 속성 값을 작은 이미지의 `src` 속성 값으로 바꾸면 됨

3. 웹 요소의 속성 가져와서 수정하기

9

- 작은 이미지 src 속성을 가져와 큰 이미지의 src 속성에 할당
 - ▣ 1) .small 요소 중 하나를 누르면 해당 요소의 src 속성 값을 알아냄
 - ▣ 2) #book 요소의 src 속성에 1)에서 알아낸 값을 할당함



3. 웹 요소의 속성 가져와서 수정하기

10

□ js파일

```
for(i=0; i<smallPics.length; i++) {  
    smallPics[i].onclick=changePic;  
}
```

```
var bigPic = document.querySelector("#book"); // 큰 이미지 가져옴  
var smallPics = document.querySelectorAll(".small"); // 작은 이미지들을 노드 리스트로 가져옴  
  
for(var i = 0; i < smallPics.length; i++) {  
    smallPics[i].addEventListener("click", changePic); // 노드를 클릭하면 changePic 함수 실행  
}  
  
function changePic() {  
    var newPic = this.src; // click 이벤트가 발생한 대상의 src 속성 값 가져옴  
    bigPic.setAttribute("src", newPic); // newPic 값을 큰 이미지의 src 속성에 할당.  
}
```

3. 웹 요소의 속성 가져와서 수정하기

11

- **setAttribute()함수 대신 속성 사용하기**
 - ▣ 웹 요소의 속성 값을 수정할 때 setAttribute()함수를 사용하지 않고 속성을 사용해도 됨

```
var bigPic = document.querySelector("#book");
var smallPics = document.querySelectorAll(".small");

for(i=0; i<smallPics.length; i++) {
    smallPics[i].onclick=function(e){
        bigPic.src = this.src;
    };
}
```

3. 웹 요소의 속성 가져와서 수정하기

12

□ CSS파일

```
#container {  
    width:600px;  
    margin:0 auto;  
}  
#prod-pic, #desc {  
    float:left;  
}  
#prod-pic {  
    margin:20px 20px auto 10px;  
    padding:0;  
}  
.small {  
    width:98px;  
    height:80px;  
}  
#desc {  
    width:300px;  
    padding-top:20px;  
    margin-bottom:50px;  
}  
.price {  
    color:#4343ff;  
    font-weight:bold;  
}
```

```
#desc button {  
    margin-top:20px;  
    margin-bottom:20px;  
    width:100%;  
    padding:10px;  
}  
#desc ul {  
    list-style:none;  
}  
#desc li{  
    font-size:0.9em;  
    line-height:1.8;  
}
```

4. DOM에서 이벤트 처리하기

13

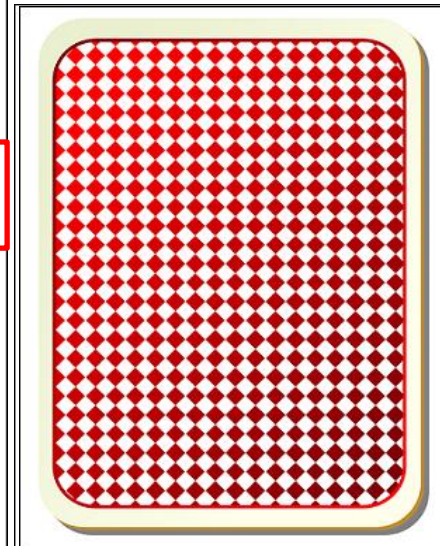
- 이벤트 처리기[이벤트 핸들러]
 - ▣ 이벤트와 이벤트 처리 함수를 연결해 주는 것
 - ▣ 이벤트 이름 앞에 on을 붙여서 사용
 - ▣ ex) click 이벤트가 발생했을 때 이벤트 처리기는 onclick 다음에 있는 실행 명령이나 함수를 연결함

4. DOM에서 이벤트 처리하기

14

- HTML 태그 안에서 이벤트 처리기 연결하기
 - ▣ 이벤트가 발생하는 HTML 태그 안에 직접 이벤트 처리기를 추가하는 방법

```
<!DOCTYPE html>
<head>
  <link rel="stylesheet" href="domevent.css">
</head>
<body>
  <div id="container">
    
  </div>
  <script>
    var pic = document.querySelector('#pic');
    function changePic() {
      pic.src = "card.png";
    }
  </script>
</body>
</html>
```



4. DOM에서 이벤트 처리하기

15

- DOM 요소에 이벤트 처리기 연결하기
 - ▣ 이미지 요소를 가져와 pic 변수에 저장한 후 click 이벤트가 발생했을 때 changePic()함수 실행
 - ▣ 이벤트 처리기를 자바스크립트 소스에서 실행

```
<body>
  <div id="container">
    
  </div>
  <script>
    var pic = document.querySelector('#pic');
    pic.onclick = changePic;

    function changePic() {
      pic.src = "card.png";
    }
  </script>
</body>
```

4. DOM에서 이벤트 처리하기

16

- DOM 요소에 이벤트를 처리하는 방법
 - ▣ HTML 태그와 뒤섞이지 않고 자바스크립트 소스를 사용한다는 점에서는 좋지만
 - ▣ 하나의 요소에 하나의 이벤트 처리기만 사용할 수 있음
- **addEventListener()**
 - ▣ 한 요소에 여러 이벤트가 발생했을 때 이를 동시에 처리할 때 사용하면 됨
 - ▣ 이벤트가 발생한 요소에 이벤트를 처리하는 연결을 주는 함수
 - ▣ 웹 문서에서 이미지나 텍스트 등 특정 요소뿐만 아니라 Document 객체나 Window 객체 어디에서든 사용할 수 있음

4. DOM에서 이벤트 처리하기

17

- 문서에 있는 이미지 요소를 pic 변수에 저장
- pic 변수에 지정된 요소에 마우스 커서를 올리면 changePic()함수 실행

```
var pic = document.querySelector('#pic');
pic.addEventListener("mouseover", changePic);
pic.addEventListener("mouseout", originPic);

function changePic() {
    pic.src = "tree1.jpg";
}
function originPic() {
    pic.src = "tree2.jpg";
}
```

mouseover
마우스 포인터가 요소위로 옮겨질 때 이벤트가 발생함

mouseout
마우스 포인터가 요소를 벗어날 때 이벤트가 발생함

```
<body>
  <div id="container">
    
  </div>

  <script src="domevent-result.js"></script>
</body>
```

4. DOM에서 이벤트 처리하기

18

- ❑ **pic.addEventListener("mouseover", changePic);**
- ❑ **이벤트 유형**
 - ❑ 처리할 이벤트 유형을 지정함
 - ❑ 이벤트 유형을 지정할 때는 on을 붙이지 않고 click, mouseover처럼 이름만 사용함
- ❑ **함수**
 - ❑ 이벤트가 발생했을 때 실행할 명령을 나열하거나 따로 함수를 만들었다면 함수 이름을 지정함



4. DOM에서 이벤트 처리하기

19

- **addEventListener()** 함수를 사용해서 웹 문서의 어딘가를 클릭하든지 웹 클라이언트 라는 알림 창이 나타나게 하려면?

```
document.addEventListener('click',function(){  
    alert("웹 클라이언트")  
});
```

이 페이지 내용:

웹 클라이언트

확인

5. 웹 요소의 스타일 가져와서 수정하기

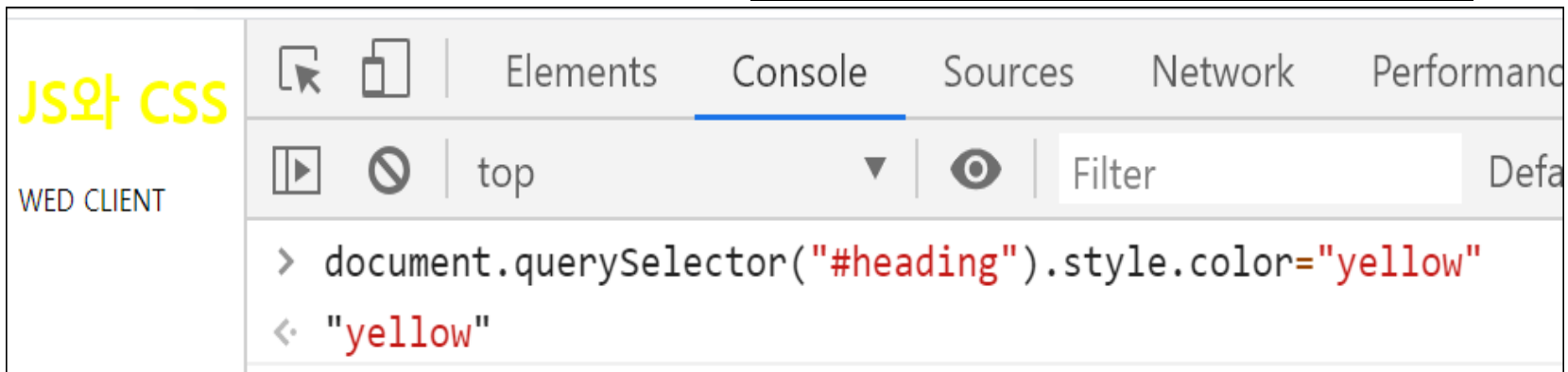
20

□ DOM으로 CSS 속성에 접근하고 수정하기

▣ 텍스트 색상 바꾸기

- 웹 요소의 스타일 속성에 접근할 때는 요소 다음에 .style 예약어 입력 후 CSS 속성을 사용함

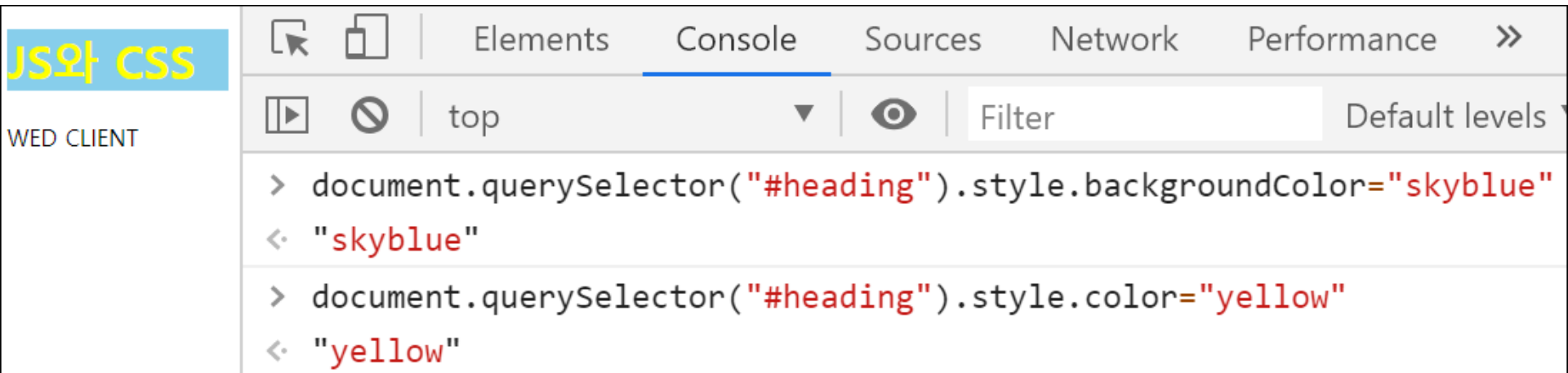
```
<body>  
  <h1 id="heading">JS와 CSS</h1>  
  <p id="desc">WED CLIENT</p>  
</body>
```



5. 웹 요소의 스타일 가져와서 수정하기

21

- background-color처럼 가운데 하이픈이 포함된 속성
 - ▣ backgroundColor처럼 두 단어를 합치고 두 번째 단어의 첫글자를 대문자로 쓰는 낙타 표기법으로 입력함



5. 웹 요소의 스타일 가져와서 수정하기

22

- **display : "none"**
 - ▣ 화면에서 감출 때 사용
- **display : "block"**
 - ▣ 화면에 표시할 때 사용



```
<div id="desc">
  <ul>
    <li>상품명 : 네트워크개론</li>
    <li class="price">판매가 : 24,000원</li>
    <li>배송비 : 2,500원<br>(20,000원 이상 구매 시 무료)</li>
    <li>적립금 : 2400원(10%)</li>
    <button>장바구니 담기</button>
  </ul>
  <a href="#" id="view">상세 설명 보기</a>
</div>
</div>

<hr>
<div id="detail">
  <h2>목차</h2>
  <ul>
    <li>네트워크 이해</li>
    <li>네트워크 구성</li>
    <li>네트워크 통신</li>
    <li>OSI 참조모델</li>
    <li>네트워크 계층</li>
    <li>전송 계층</li>
  </ul>
</div>
</div>
<script src="showBig-result.js"></script>
```

5. 웹 요소의 스타일 가져와서 수정하기

```
var isOpen = false;
var view = document.querySelector("#view"); // #view 요소를 가져옴

view.addEventListener("click", function() {
    if (isOpen == false) { // 상세 정보가 감춰져 있다면
        document.querySelector("#detail").style.display = "block";
        // 상세 정보를 화면에 표시
        view.innerHTML = "상세 설명 닫기"; // 링크 내용 수정
        isOpen = true; // 표시 상태로 지정
    }
    else { // 상세 정보가 표시되어 있다면
        document.querySelector("#detail").style.display = "none";
        // 상세 정보를 화면에서 감춤
        view.innerHTML = "상세 설명 보기";
        // 링크 내용 수정
        isOpen = false; // 감춰진 상태로 지정
    }
});
```