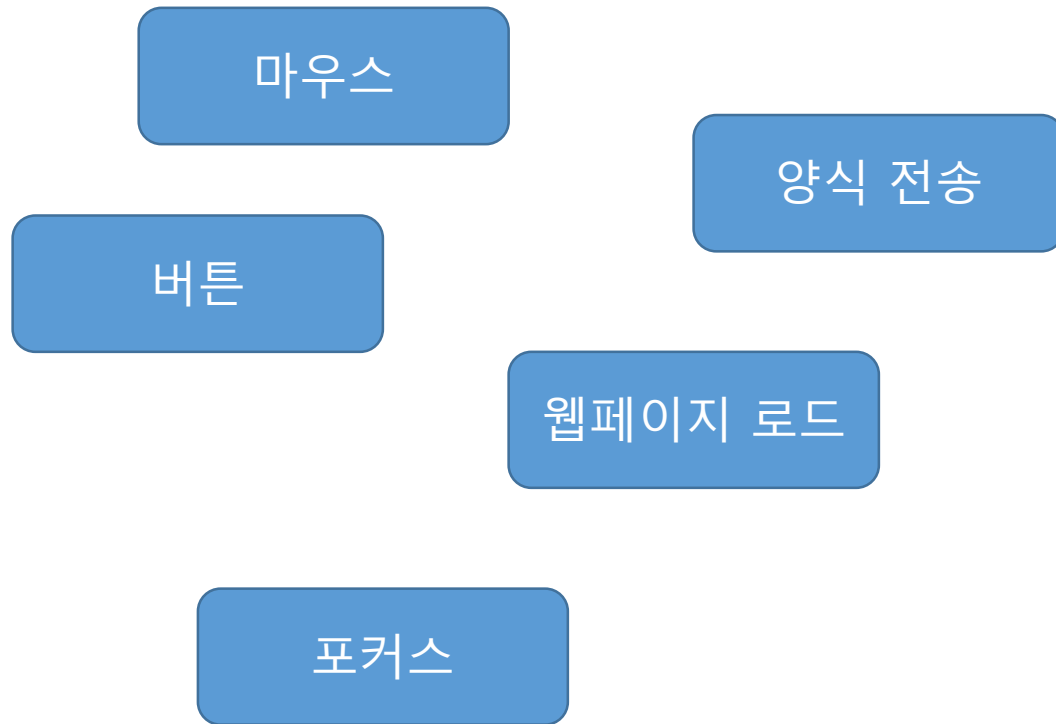


15강_javascript 이벤트 이론

- 15-1 이벤트란?
- 15-2 이벤트 모델
- 15-3 인라인 모델
- 15-4 기본 모델
- 15-5 MS핸들러 모델
- 15-6 표준핸들러 모델
- 15-7 이벤트 객체

15-1 : 이벤트란?



이벤트 형식

이벤트 속성 이벤트 핸들러

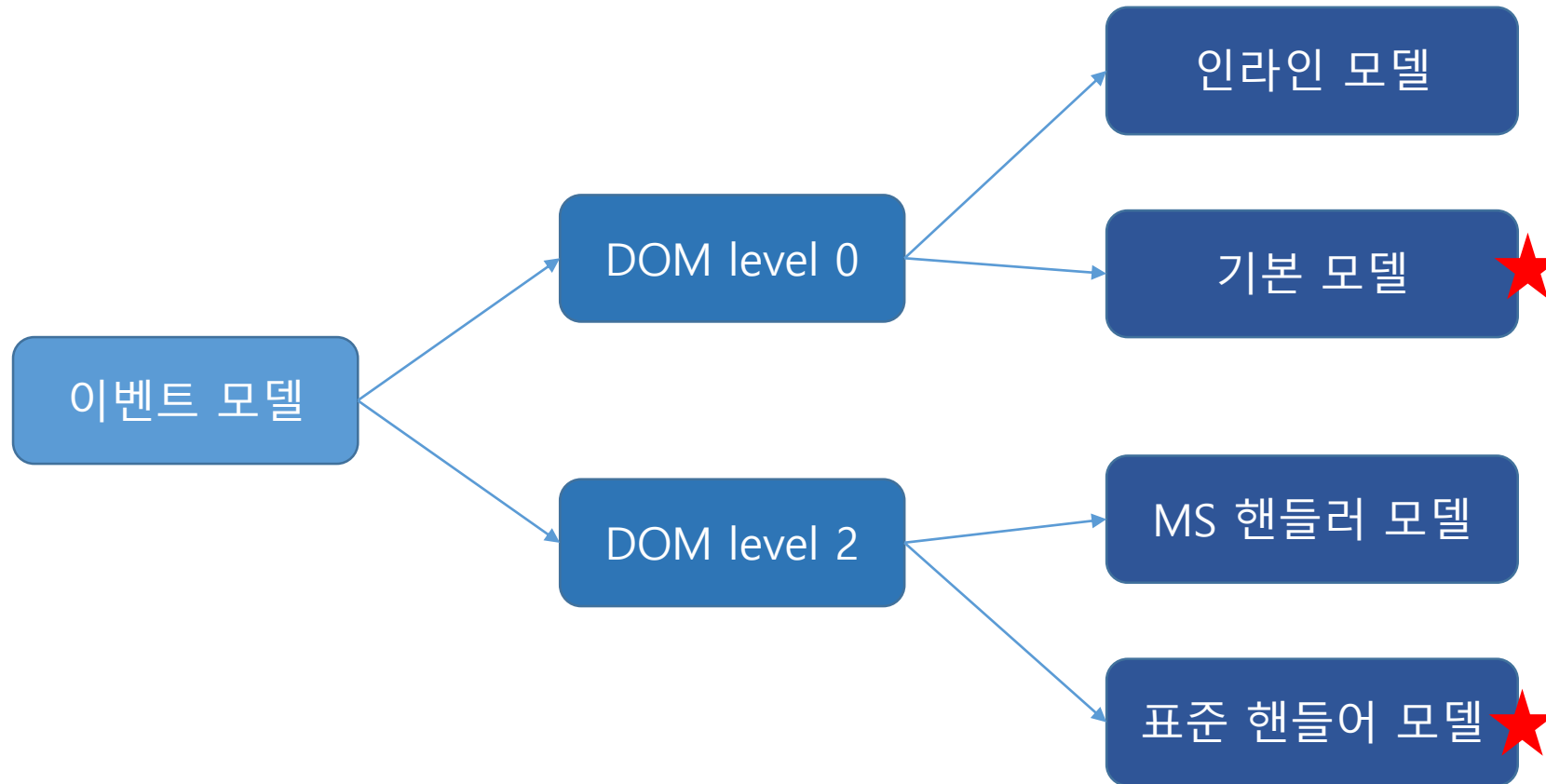
```
xxx.onclick = function(){
```

실행문...

```
};
```

이벤트 이름 이벤트 연결(모델)

15-2 : 이벤트 모델



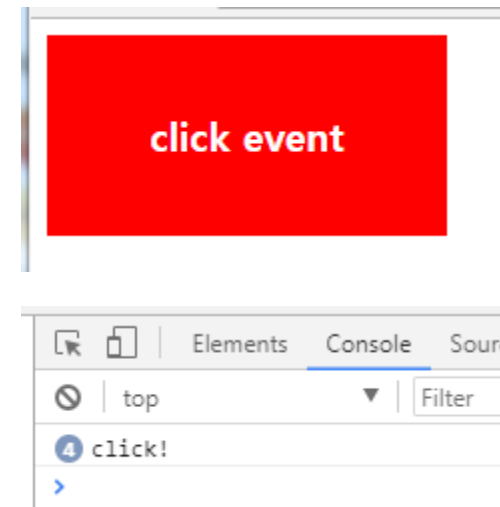
15-3 : 인라인 모델

Ex : 15_01.html

```
<script>
function headerClick(){
  console.log("click!");

  /*
  //이벤트 제거
  var ce = document.getElementById("cEvent");
  console.log(ce);
  ce.onclick = null;
  */
};
</script>

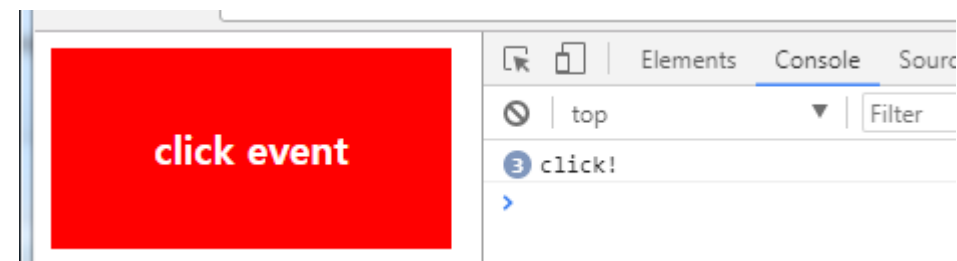
<style>
#cEvent {
  width: 200px; height: 100px;
  line-height: 100px;
  text-align: center;
  font-size: 1.2em;
  background-color: #f00;
  color: #fff;
  font-weight: bolder;
}
</style>
<div id="cEvent" onclick="headerClick();">click event</div>
</div>
```



15-4: 기본 모델

Ex: 15_02.html

```
window.onload = function() {  
    var ce = document.getElementById("cEvent");  
    //방법 I  
    ce.onclick = function() {  
        console.log("click!");  
        //이벤트 제거  
        //ce.onclick = null;  
    }  
    /*  
    //방법 II  
    ce.onclick = clickEventHandler;  
    function clickEventHandler() {  
        console.log("clickEventHandler!");  
        //이벤트 제거  
        //ce.onclick = null;  
    }  
    */  
};
```



```
<div id="cEvent">click event</div>
```

15-5: MS 핸들러 모델

Ex: 15_03.html

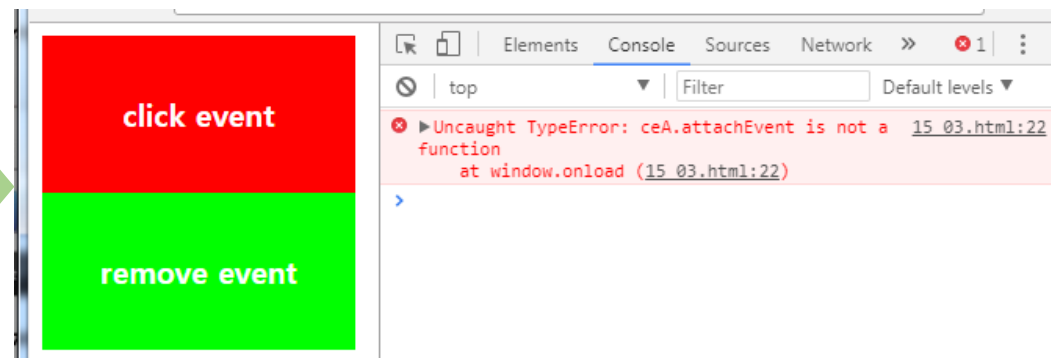
```
/*
  IE8이전 버전에서 동작
*/
window.onload = function() {
  function addHandler() {
    console.log("click!!");
  };

  function removeHandler() {
    console.log("event remove!!");
    ceA.detachEvent("onclick", addHandler, false);
  };

  var ceA = document.getElementById("cEventAdd");
  ceA.attachEvent("onclick", addHandler, false);

  var ceR = document.getElementById("cEventRem");
  ceR.attachEvent("onclick", removeHandler, false);
};
```

```
<div id="cEventAdd">click event</div>
<div id="cEventRem">remove event</div>
```

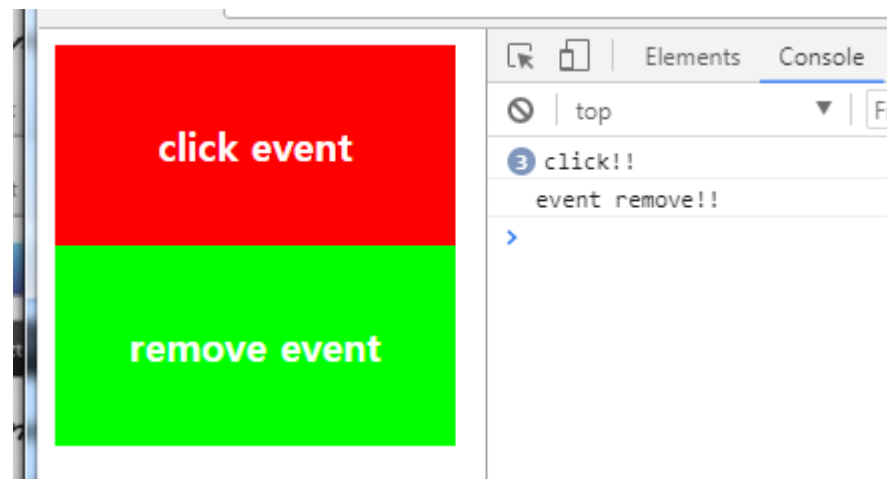


15-6 : 표준 핸들러 모델

Ex : 15_04.html

```
window.onload = function() {  
    function addHandler() {  
        console.log("click!!");  
    };  
  
    function removeHandler() {  
        console.log("event remove!!");  
        ceA.removeEventListener("click", addHandler, false);  
    };  
  
    var ceA = document.getElementById("cEventAdd");  
    ceA.addEventListener("click", addHandler, false);  
  
    var ceR = document.getElementById("cEventRem");  
    ceR.addEventListener("click", removeHandler, false);  
};
```

```
<div id="cEventAdd">click event</div>  
<div id="cEventRem">remove event</div>
```



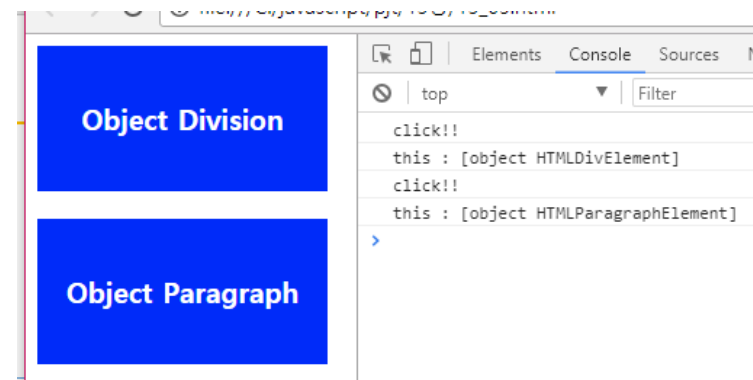
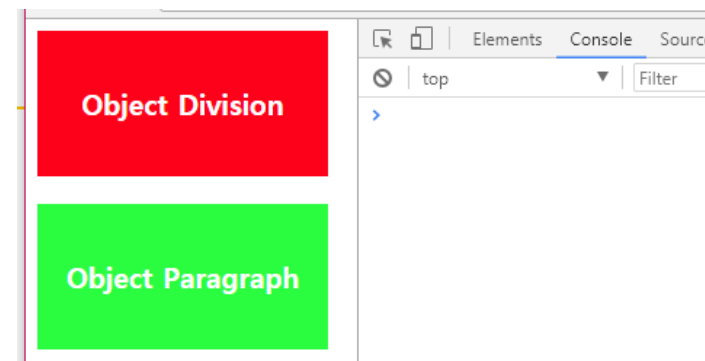
15-7 : 이벤트 객체

Ex : 15_05.html

this → 이벤트가 발생한 객체

```
window.onload = function() {  
  function addHandler() {  
    console.log("click!!");  
    console.log("this : " + this);  
  
    this.style.backgroundColor = "#0000ff";  
  };  
  
  var objD = document.getElementById("objDiv");  
  objD.addEventListener("click", addHandler, false);  
  
  var objP = document.getElementById("objPar");  
  objP.addEventListener("click", addHandler, false);  
};
```

```
<div id="objDiv">Object Division</div>  
<p id="objPar">Object Paragraph</p>
```



15-8 : 이벤트 객체

Ex : 15_06.html

event → 이벤트 객체

```
function addHandler(e) {  
    console.log("click!!");  
    console.log("this : " + this);  
  
    //IE8 이전버전을 고려  
    var event = e || window.event;  
    for( var key in event ) {  
        console.log(key + " : " + event[key]);  
    }  
};
```



Object Division

Object Paragraph

Property	Value
click!!	
this	[object HTMLDivElement]
isTrusted	true
screenX	2217
screenY	239
clientX	129
clientY	55
ctrlKey	false
shiftKey	false
altKey	false
metaKey	false
button	0
buttons	0
relatedTarget	null
pageX	129
pageY	55
x	129
y	55
offsetX	121
offsetY	47
movementX	0
movementY	0
fromElement	null
toElement	[object HTMLDivElement]
layerX	129