

# event.html

2019-14505 엄세희

## 문제 풀이

### 문제1. 클릭하여 색깔 바꾸기

클릭하면 색상이 바뀌는 이벤트 리스너를 추가합니다. 한 번도 클릭하지 않은 경우 또는 배경색이 "white" 인 경우, 배경색을 "lightblue"로 바꾸고, 배경색이 "lightblue"인 경우 "white"로 바꿔줍니다.

```
<html>
  <head>
    <title>클릭하여 색깔 바꾸기</title>
    <style>
      #click-me {
        padding: 10px;
        border: solid 1px black;
        width: 100%;
      }
    </style>
  </head>
  <body>
    <h1>클릭하여 색깔 바꾸기</h1>
    <div id="click-me">
      클릭하면 배경색이 바뀝니다.
    </div>
    <script>
      var obj = document.getElementById("click-me");
      obj.addEventListener("click", function() {
        if (this.style.backgroundColor == "lightblue")
        {this.style.backgroundColor = "white";}
        else {this.style.backgroundColor = "lightblue";}
      });
    </script>
  </body>
</html>
```

### 문제2. 클릭 횟수 세기

count버튼은 클릭하면 누른 횟수만큼 countDisplay 버튼에 기록되도록 이벤트 리스너를 추가합니다. reset버튼은 누르면 countDisplay에 숫자 "0"이 뜨도록 해줍니다.

```
<html>
  <head>
    <meta charset="utf-8">
    <title>클릭 횟수 세기</title>
    <style>
      #countDisplay {
        padding: 4px;
        border: 2px solid darkgreen;
      }
    </style>
  </head>
  <body>
    <div id="count">0</div>
    <div id="countDisplay">0</div>
    <div id="reset">Reset</div>
  </body>
</html>
```

```

    }
    </style>
</head>
<body>
    <h1>클릭 횟수 세기</h1>
    <button id="countButton">COUNT</button>
    <span id="countDisplay">0</span>
    <button id="resetButton">RESET</button>
    <script>
        var clicknum = 0;
        var countButton = document.getElementById("countButton");
        var resetButton = document.getElementById("resetButton");
        var countDisplay = document.getElementById("countDisplay");
        countButton.addEventListener("click",function() {
            clicknum += 1;
            countDisplay.innerHTML = clicknum;
        });
        resetButton.addEventListener("click", function() {
            clicknum = 0;
            countDisplay.innerHTML = clicknum;
        });
    </script>
</body>
</html>

```

### 문제3. 전화 번호 입력기

전화번호를 입력하도록, "key"라는 클래스를 부여받은 버튼들에 이벤트 리스너를 추가해줍니다. 문제2에서의 리셋버튼처럼, 전화 버튼을 누르면 telephone-number란에 입력되었던 전화번호들이 지워지고, 기존의 디폴트 디스플레이 값이었던 "전화번호를 누르세요"라는 문구가 뜨도록 합니다. 이와 동시에, 통화 버튼을 누른 이후 다시 전화번호를 누르면 해당 번호부터 표시되도록, 전화번호 입력을 받는 변수를 ""로 리셋해줍니다.

```

<html>
<head>
    <meta charset="utf-8">
    <title>전화 번호 입력기</title>
    <style>
        #telephone-number {
            padding : 4px;
            border : 1px solid white
        }
    </style>
</head>
<body>
    <h1>전화 번호 입력기</h1>
    <div id="telephone">
    <div id="telephone-number">전화번호를 누르세요</div>
    <table id="keypad">
        <tr>
            <td><button value="1" class="key">1</button></td>
            <td><button value="2" class="key">2</button></td>
            <td><button value="3" class="key">3</button></td>
        </tr>
    </table>

```

```

<tr>
  <td><button value="4" class="key">4</button></td>
  <td><button value="5" class="key">5</button></td>
  <td><button value="6" class="key">6</button></td>
</tr>
<tr>
  <td><button value="7" class="key">7</button></td>
  <td><button value="8" class="key">8</button></td>
  <td><button value="9" class="key">9</button></td>
</tr>
<tr>
  <td><button value="*" class="key">*</button></td>
  <td><button value="0" class="key">0</button></td>
  <td><button value="#" class="key">#</button></td>
</tr>
<tr>
  <td></td>
  <td><button id="call">통화</button></td>
  <td></td>
</tr>
</table>
</div>
<script>
  var keyButtons = document.getElementsByClassName("key");
  var callButton = document.getElementById("call");
  var input = "";
  var telephoneNumber = document.getElementById("telephone-number");
  for (i = 0; i <= 11; i++) {
    keyButtons[i].addEventListener("click", function(){
      input += this.value;
      telephoneNumber.innerHTML = input;
    });
    callButton.addEventListener("click", function(){
      input = "";
      telephoneNumber.innerHTML = "전화번호를 누르세요";
    })
  }
</script>
</body>
</html>

```

## 문제4. 계산기

계산기의 css는 문제에 있던 것으로 대체합니다. 계산기의 다른 버튼들은 누르는 대로 값을 입력받아 디스플레이에 수식의 형태로 나타나도록 하고, "="값을 가진 버튼의 경우 누르면 직전까지 디스플레이에 나타냈던 수식을 eval함수에 넣은 값을 디스플레이에 표시하도록 합니다. 이 때도 문제3과 동일하게, "="버튼을 누른 뒤 다시 수식을 입력하기 시작하면 그 값부터 표시하도록, 수식을 나타내는 변수의 값을 ""로 초기화해줍니다.

```

<html>
  <head>
    <meta charset="utf-8">
    <title>계산기</title>
    <style>

```



```

        </tr>
        <tr>
        <td><button value="4" class="key">4</button></td>
        <td><button value="5" class="key">5</button></td>
        <td><button value="6" class="key">6</button></td>
        <td><button value="-" class="key">-</button></td>
        </tr>
        <tr>
        <td><button value="1" class="key">1</button></td>
        <td><button value="2" class="key">2</button></td>
        <td><button value="3" class="key">3</button></td>
        <td><button value="*" class="key">*</button></td>
        </tr>
        <tr>
        <td><button value="0" class="key">0</button></td>
        <td><button value="." class="key">.</button></td>
        <td><button value="/" class="key">/</button></td>
        <td><button value="=" id="eval-button">=</button></td>
        </tr>
    </table>
</div>
<script>
    var keyButtons = document.getElementsByClassName("key");
    var display = document.getElementById("display");
    var mathinput = "";
    var evalButton = document.getElementById("eval-button");
    for (i=0; i<=14; i++){
        keyButtons[i].addEventListener("click", function(){
            mathinput += this.value;
            display.innerHTML = mathinput;
        })
    }
    evalButton.addEventListener("click", function(){
        displayval = eval(mathinput);
        mathinput = "";
        display.innerHTML = displayval;
    })
</script>
</body>
</html>

```

## 문제7. 만년 달력 만들기

마찬가지로 달력을 만드는 부분, 그리고 css는 그대로 가져왔습니다. 숫자를 입력하면 이에 맞춰 디스플레이 되도록 이벤트 리스너만 추가했습니다. html을 처음 열었을 때 나타나는 디폴트 달력, 그리고 반짝이는 날짜는 오늘로 설정했습니다. textarea에 새로운 숫자를 입력하면 기존에 있던 text는 지워지도록 하고, 새로운 값에 맞춰서 문자열을 추가하도록 합니다.

```

<html>
  <head>
    <meta charset="utf-8">
    <style>
      table {
        border-collapse : collapse;
        font-family: Verdana, Geneva, Tahoma, sans-serif;

```

```

    }
    td {
        text-align: right;
        padding: 5px;
        color : gray;
    }
    .thismonth {
        color : black;
    }
    .w0 {
        color : white;
        background-color: red;
    }
    .w6 {
        color: white;
        background-color: blue;
    }
    .theday {
        animation: blinker 2s step-start infinite;
    }
    @keyframes blinker {
        50% {
            opacity: 0.1;
        }
    }
}
</style>
<script>
// initialize calendar
function init() {
    inputYear = document.getElementById("year")
    inputMonth = document.getElementById("month")
    inputDay = document.getElementById("date")
    calendarContainer = document.getElementById("calendar-container")
    var today = new Date();
    var thisyear = today.getFullYear();
    var thismonth = today.getMonth();
    var thisday = today.getDate();
    inputYear.value = thisyear; // 디폴트 디스플레이에 맞춰 설정해준다
    inputMonth.value = thismonth + 1;
    inputDay.value = thisday;
    createCalendar(thisyear, thismonth, thisday);
    inputYear.addEventListener("change", newCalendar);
    inputMonth.addEventListener("change", newCalendar);
    inputDay.addEventListener("change", newCalendar);
}
// implement calendar
function newCalendar() {
    var year = inputYear.value;
    var month = inputMonth.value - 1; // createCalendar 함수 참고
    var day = inputDay.value;
    calendarContainer.innerHTML = ""; // 기존에 있던 캘린더를 지워야함
    createCalendar(year, month, day)
}
function createCalendar(year, month, day) {
    var weeknames = ['㉠', '㉡', '㉢', '㉣', '㉤', '㉥', '㉦'];
    var theday = new Date(year, month, day);
    theday.setFullYear(year);
    var d = theday.getDate();

```

```

        var yyyy = theday.getFullYear();
        var m = theday.getMonth() + 1;
        var w = theday.getDay();
        var h1 = document.createElement("h1");
        calendarContainer.appendChild(h1);
        h1.innerHTML = yyyy + "년 " + m + "월";
        var date = new Date(yyyy, m-1, 1);
        date.setDate(date.getDate() - date.getDay())
        var table = document.createElement("table");
        calendarContainer.appendChild(table);
        var thead = document.createElement("thead");
        table.appendChild(thead);
        for(var j=0; j<7; j++) {
            var th = document.createElement(`th`);
            th.innerHTML = weeknames[j];
            thead.appendChild(th);
        }
        for(var i=0; i<5; i++) {
            var tr = document.createElement("tr");
            for(var j=0; j<7; j++) {
                cls = 'w' + date.getDay();
                if (date.toDateString() == theday.toDateString()) {
                    cls += ' ' + 'theday';
                }
                if (date.getMonth() == theday.getMonth()) {
                    cls += ' ' + 'thismonth';
                }
                var td = document.createElement("td");
                td.setAttribute("class", cls);
                td.innerHTML = date.getDate();
                date.setDate(date.getDate() + 1);
                tr.appendChild(td);
            }
            table.appendChild(tr);
        }
    }
}
</script>
</head>
<body onload="init()">
    <input id="year" type="number">
    <input id="month" type="number" min="1" max="12">
    <input id="date" type="number" min="1" max="31">
    <div id="calendar-container">
    </div>
</body>
</html>

```

## 문제8. 문장 순서 맞추기 문제 만들기

문장 여러 개로 이루어진 문자열을 입력하면, 문장 단위로 쪼개서 랜덤하게 순서를 바꿔주는 문제입니다. 순서에 맞추어 버튼을 누르면, 해당 버튼의 문자열이 입력되고, 눌린 버튼의 색상이 바뀝니다. 리셋 버튼을 누르면 이전까지 클릭했던 버튼의 색상이 다시 흰색으로 바뀌고, 입력되었던 내용이 지워집니다.

```

<html>
<head>

```

```

<meta charset="utf-8">
<title>Sentence Order Test Generator</title>
<style>
#text-input {
width: 640px;
height: 100px;
}
#translation {
width: 640px;
}
#cloze-container {
width : 640px;
}
#sentenceButton {
width:640px;
text-align: left;
}
#answerArea {
width:640px;
height:100px;
background-color:royalblue;
color: white;
}
</style>
</head>
<body>
<h1>Sentence Order Test Generator</h1>
<h2>중국어 원문</h2>
<textarea id="text-input">

```

对身体好的水果可以原样吃，但是使用食谱的话，可以包含在每日饮食中。这三种菜谱，在无聊的时候做得很容易，所以只能待在家里的最近做这正合适。用香蕉可以做酪梨香蕉汁。 做这个很容易，连小孩子都能做。 牛奶， 酪梨，和香蕉一起磨就完了。 加坚果会更香，加蜂蜜会更甜</textarea>

```

<h2></h2>
<div>
<button id="gen-test-button">생성</button>
</div>
<div id="test-container">
</div>
<script>
var textInput = document.getElementById("text-input");
var genTestButton = document.getElementById("gen-test-button");
var testContainer = document.getElementById("test-container");
var textInput = textInput.value;
var sentences = textInput.split("。");

function shuffle(a) {
var j, x, i;
for (i = a.length - 1; i > 0; i--) {
j = Math.floor(Math.random() * (i + 1));
x = a[i];
a[i] = a[j];
a[j] = x;
}
return a;
}

genTestButton.addEventListener("click", function() {
sentences = shuffle(sentences)

```



```

        var testText = "";
        for (i=0; i<sentences.length; i++){
            testText += "<button class = \"sentence-button\">" + sentences[i]
+ "</button><br>";
        }
        testText += "<textarea id=\"answerArea\"></textarea>";
        testText += "<button id = \"reset-button\">RESET</button>"
        testText += "<button id = \"submit-button\">제출</button>";
        testContainer.innerHTML = testText;

        var answerArea = document.getElementById("answerArea");
        answerArea.value = "정답: ";
        var sentenceButtons = document.getElementsByClassName("sentence-
button");

        for (v=0; v<sentenceButtons.length; v++) {
            sentenceButtons[v].addEventListener("click", function(){
                answerArea.value += this.innerHTML + " ";
                this.style.backgroundColor = "grey";
            })
        }

        var resetButton = document.getElementById("reset-button");
        resetButton.addEventListener("click", function(){
            answerArea.value = "정답: ";
            for (n=0; n<sentenceButtons.length; n++){
                sentenceButtons[n].style.backgroundColor = "white";
            }
        })
        var submitButton = document.getElementById("submit-button");
        submitButton.addEventListener("click", function(){
            for (n=0; n<sentenceButtons.length; n++){
                sentenceButtons[n].style.backgroundColor = "white";
            }
            // submit answerArea.value
            answerArea.value = "정답:";
        })
    })
</script>
</body>
</html>

```

## 질문

- 이미 클릭된 버튼은 이후 다시 클릭하더라도 이벤트리스너가 작용될 수 없도록 만드는 방법이 있나  
요? 검색해보았는데 addEventListener와 removeEventListener를 함께 사용하라고 합니다. 8번  
문제에서 만든 코드에서 addEventListener("click", function(){함수작용부분}) <- 이 안의 함수작용  
부분에 removeEventListener를 집어넣어 보았는데 아무런 효과가 없었습니다. addEventListener  
안에 remove를 넣어야, 클릭하기도 전에 이벤트리스너가 작용하지 않는 것을 방지할 수 있을 것 같  
은데, 제가 잘못 생각하고 있는 것일까요? 그리고 removeEventListener의 인자로 어떤 것을 넣어  
야 할지도 모르겠습니다.
- 7번 문제 풀이를 보면 script안에 이런 코드가 있는데, 이 부분이 왜 필요한지, 어떤 역할을 하는지  
궁금합니다.

```
var yearInput = undefined;  
var monthInput = undefined;  
var dateInput = undefined;  
var calendarContainer = undefined;
```