

(디지털커버전스)스마트 콘텐츠와 웹 융합응용SW개발자 양성과정

강사 - Innova Lee(이상훈)

gcccompil3r@gmail.com

학생 - Joongyeon Kim(김종연)

jjjr69@naver.com

2021년 6월 30일 지문노트

[김종연]

```

var itemList = [];

// id가 add인 녀석을 클릭 했을때 addList()가 동작함
var addBtn = document.querySelector("#add");
addBtn.addEventListener("click", addList);

function addList () {
    // 입력 데이터 가져오기
    var item = document.querySelector("#item").value;

    if (item != null) {
        // 입력 데이터 추가
        itemList.push(item);
        // id가 "item"인 요소 값을 지움
        document.querySelector("#item").value = "";
        // 입력창에 포커싱 적용
        document.querySelector("#item").focus();
    }

    showList();
}

function showList () {
    // <ul> 태그로 리스팅 준비
    var list = "<ul>";

    // 요소마다 반복시켜 리스팅하고 x 표시 눌러서 사라지게
    for (var i = 0; i < itemList.length; i++) {
        list += "<li>" + itemList[i] + "<span class='close' id=" + i + ">X</span></li>";
    }
    // 리스팅을 끝내는 </ul> 태그 셋팅
    list += "</ul>";

    document.querySelector('#itemList').innerHTML = list;

    // querySelector는 날개로 가져오고
    // querySelectorAll은 전부 가져옴
    var remove = document.querySelectorAll(".close");

    // x 표시 누르면 removeList()가 동작하게
    for (var i = 0; i < remove.length; i++) {
        remove[i].addEventListener("click", removeList);
    }
}

function removeList () {
    // 모든 객체는 자기 자신의 this를 가지고 있어서
    // this(클릭한 삭제 버튼 x)의 id 값을 가져와 id로 설정
    var id = this.getAttribute("id");

    // itemList 배열에서 인덱스 값이 id인 요소 1개를 삭제함
    // (2개면 숫자가 2가 들어가며 3개면 3임)
    itemList.splice(id, 1);

    // 변경 정보를 다시 화면 출력함
    showList();
}

```

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet" href="/css/js/input.css">
    <link rel="stylesheet" href="/css/js/list.css">
</head>
<body>
    <div id="wrapper">
        <h2>체크 리스트</h2>
        <form>
            <input type="text" id="item" autofocus="true">
            <button type="button" id="add" class="addBtn">추가</button>
        </form>
        <div id="itemList"></div>
    </div>
    <script src="/js/checklist.js"></script>
</body>
</html>

```

체크 리스트

ddd
X

Js로 ArrayList를 사용하기

```

var state = 1;

function toggleButton () {
  var list = document.querySelectorAll('#vueapp p');

  for (var i=0; i< list.length; i++){
    //state 값이 0이면 짝수 번째, 1이면 홀수 번째
    //루프 전체는 6개
    //어떤 경우엔 1, 3, 5
    //어떤 경우엔 0, 2, 4

    // 0이면 거짓 0이 아니면 참
    if (!((i+state) % 2)) {
      list[i].style.visibility="visible";
    } else{
      list[i].style.visibility="hidden";
    }
  }

  if(state == 1){
    state = 0;
  } else {
    state = 1;
  }
}

```

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
  <div id="vueapp">
    <p style="visibility: hidden">우왕 많다.</p>
    <p style="visibility: hidden">오진다.</p>
    <p style="visibility: hidden">책</p>
    <p style="visibility: hidden">많다.</p>
    <p style="visibility: hidden">살려주세요</p>
    <p style="visibility: hidden">공부할 분량이 너무 많아요</p>
  </div>
  <button onclick="toggleButton()">보여주세요!</button>
  <script src="js/domSelfTest.js"></script>
</body>
</html>

```

우왕 많다.

오진다.

책

많다.

살려주세요

공부할 분량이 너무 많아요

보여주세요!

보여주세요!

Dom은 사용하여 버튼을 누를 때 마다 홀수열과 짝수열이 번갈아 나오는 코드

if (!((i+state) % 2)) 부분에서 처음에 $!(0+1 \% 2 = 1)$ 이 되어서 거짓이지만 !로 인해서 반전되어 참이 된다.
 그렇게 첫 반복에서 빠져나온후에는
 아래 if(state == 1) 부분에서 값이 0으로 초기화되고 다시 반복할 때는 $!(0+0 \% 2 = 0)$ 이 되어서 값이 거짓이
 되어 엘스문에 들어가게 된다
 그렇게 짝수 번째 홀수 번째가 번갈아 가며 상태가 바뀌게 되는 것이다
 (요약 : state가 반복할때마다 0과1이 바뀌어서 출력이 달라짐)

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
  <script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
</head>
<body>
  <div id="vueapp">
    <h1>{{ message }}</h1>
    <p>{{ repeatText }}</p>
    <p>{{ repeatText }}</p>
    <p>{{ repeatText }}</p>
    <p>{{ repeatText }}</p>
    <p>{{ repeatText }}</p>
    <p>{{ repeatText }}</p>
    <p>{{ repeatText }}</p>
  </div>
</body>
<script>
  var app = new Vue ({
    el: '#vueapp',
    data: {
      message: '나는 겁나 길어도 이렇게 편하게 쓸 수 있어.' +
        '주구장창 늘여서 쓸 필요가 없단 말이지.' +
        '이 메시지가 또 필요하면 재활용하면 돼',
      repeatText: '재활용짱'
    }
  })
</script>
</html>

```

나는 겁나 길어도 이렇게 편하게 쓸 수 있어.주구장창 늘여서 쓸 필요가 없단 말이지.이 메시지가 또 필요하면 재활용하면 돼

재활용짱
재활용짱
재활용짱
재활용짱
재활용짱
재활용짱

Vue를 사용하면 편한 점(보잡하고 긴 코드들을
문자를 입력하여 호출할 수 있다)

