

[디지털 컨버전스] 스마트 콘텐츠와 웹 융합 응용SW 개발자 양성과정

강사 : 이상훈

학생 : 임초롱

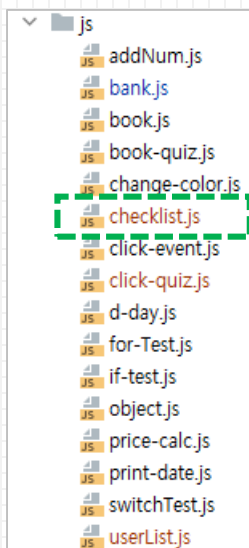
JavaScript - Array

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

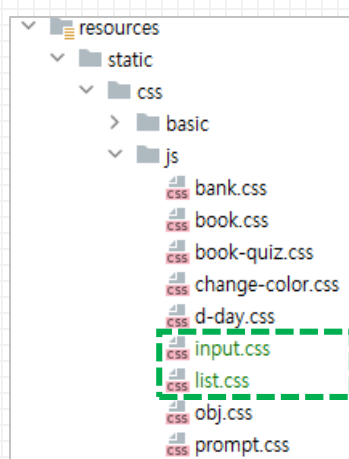
컨트롤러 클래스

```
7  @Slf4j
8  @Controller
9  public class TwelfthJavaScriptTestController {
10
11      @GetMapping("/checkListWithArrayTest")
12      public String getCheckListWithArrayTest() {
13          log.info("getCheckListWithArrayTest()");
14
15          return "/javascript/Twelfth/checkListWithArrayTest";
16      }
17  }
```

.js 위치



.css 위치



html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Title</title>
6      <link rel="stylesheet" href="/css/js/input.css">
7      <link rel="stylesheet" href="/css/js/list.css">
8  </head>
9  <body>
10     <div id="wrapper">
11         <h2>체크 리스트</h2>
12         <form>
13             <input type="text" id="item" value="값을 입력하세요." autofocus="true">
14             <button type="button" id="add" class="addBtn">추가</button>
15         </form>
16         <div id="itemList"></div>
17     </div>
18     <script src="/js/checklist.js"></script>
19 </body>
20 </html>
```

JavaScript - Array

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

CSS

```
<link rel="stylesheet" href="/css/js/input.css">
```

```
1  * {
2    box-sizing: border-box;
3  }
4
5  #wrapper {
6    width: 600px;
7    margin: 0 auto;
8  }
9
10 h2{
11   text-align: center;
12 }
13
14 form {
15   background-color: lightpink;
16   padding: 30px 40px;
17   color: white;
18   text-align: center;
19 }
20
21 input {
22   border: none;
23   width: 440px;
24   padding: 10px;
25   float: left;
26   font-size: 16px;
27 }
28
```

```
29 .addBtn {
30   padding: 10px;
31   width: 50px;
32   border: none;
33   background-color: #ffffff;
34   box-shadow: 2px 4px 6px #5f5079;
35   color: #555555;
36   text-align: center;
37   font-size: 14px;
38   cursor: pointer;
39   transition: 0.3s; /* 포커싱에 걸리는 시간 및 변경 타이밍 */
40 }
```

```
<link rel="stylesheet" href="/css/js/list.css">
```

```
1  ul {
2    margin: 0;
3    padding: 0;
4    list-style: none;
5  }
6
7  ul li {
8    cursor: pointer;
9    /* 상대적인 위치 */
10   position: relative;
11   padding: 12px 8px 12px 40px;
12   background: whitesmoke;
13   font-size: 18px;
14 }
15
16 ul li:nth-child(odd) {
17   background-color: seashell;
18 }
```

```
20 ul li:hover {
21   background-color: lavenderblush;
22 }
23
24 .close {
25   /* 절대 위치 */
26   position: absolute;
27   right: 0;
28   top: 0;
29   padding: 12px 16px;
30   border: none;
31   background: rgba(255, 255, 255, 0);
32 }
33
34 .colse:hover {
35   background-color: lemonchiffon;
36   color: sandybrown;
37 }
```

JavaScript - Array

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

js

```
1  var itemList = [];  
2  
3  // id가 add인 녀석을 클릭 했을때 addList()가 동작함  
4  var addBtn = document.querySelector("#add");  
5  addBtn.addEventListener("click", addList);  
6  
7  function addList () {  
8      // 입력 데이터 가져오기  
9      var item = document.querySelector("#item").value;  
10  
11     if (item != null) {  
12         // 입력 데이터 추가  
13         itemList.push(item);  
14         // id가 "item"인 요소 값을 지움  
15         document.querySelector("#item").value = "";  
16         // 입력창에 포커싱 적용  
17         document.querySelector("#item").focus();  
18     }  
19  
20     showList();  
21 }
```

```
23 function showList () {  
24     // <ul> 태그로 리스팅 준비  
25     var list = "<ul>";  
26  
27     // 요소마다 반복시켜 리스팅하고 x 표시 눌러서 사라지게  
28     for (var i = 0; i < itemList.length; i++) {  
29         list += "<li>" + itemList[i] + "<span class='close' id=" + i + "> X</span></li>";  
30     }  
31     // 리스팅을 끝내는 </ul> 태그 셋팅  
32     list += "</ul>";  
33  
34     document.querySelector("#itemList").innerHTML = list;  
35  
36     // querySelector는 날개로 가져오고  
37     // querySelectorAll은 전부 가져옴  
38     var remove = document.querySelectorAll(".close");  
39  
40     // X 표시 누르면 removeList()가 동작하게  
41     for (var i = 0; i < remove.length; i++) {  
42         remove[i].addEventListener("click", removeList);  
43     }  
44 }
```

```
46 function removeList () {  
47     // 모든 객체는 자기 자신인 this를 가지고 있어서  
48     // this(클릭한 삭제 버튼 x)의 id 값을 가져와 id로 설정  
49     var id = this.getAttribute("id");  
50  
51     // itemList 배열에서 인덱스 값이 id인 요소 1개를 삭제함  
52     // (2개면 숫자가 2가 들어가며 3개면 3임)  
53     itemList.splice(id, 1);  
54  
55     // 변경 정보를 다시 화면 출력함  
56     showList();  
57 }
```

JavaScript - Array

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

생성된 html

체크 리스트

추가

a	X
b	X
c	X
d	X
e	X
f	X
g	X
h	X



체크 리스트

추가

a	X
b	X
d	X
e	X
h	X

JavaScript - Array

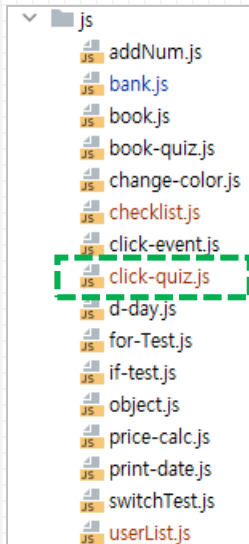
링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

컨트롤러 클래스

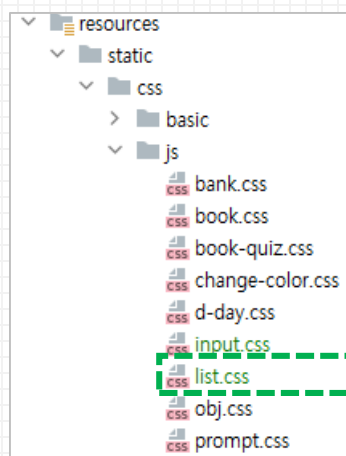
```
7  @Slf4j
8  @Controller
9  public class TwelfthJavaScriptTestController {

18  @GetMapping("/75Quiz")
19  public String get75Quiz() {
20      log.info("get75Quiz()");
21
22      return "/javascript/Twelfth/75Quiz";
23  }
```

.js 위치



.css 위치



html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Title</title>
6      <link rel="stylesheet" href="/css/js/list.css">
7  </head>
8  <body>
9      <div id="container">
10         <h2>사용자 정보 등록</h2>
11         <form>
12             <input type="text" id="name" value="이름">
13             <input type="text" id="age" value="나이">
14             <input type="text" id="major" value="전공">
15             <button type="button" id="add" class="addBtn">추가하기</button>
16         </form>
17         <div id="userList"></div>
18     </div>
19     <script src="js/userList.js"></script>
20 </body>
21 </html>
```

JavaScript - Array

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

js

```
1  var userList = [];  
2  
3  var addBtn = document.querySelector("#add");  
4  addBtn.addEventListener("click", addList);  
5  
6  function addList () {  
7  
8      var name = document.querySelector("#name").value;  
9      var age = document.querySelector("#age").value;  
10     var major = document.querySelector("#major").value;  
11  
12  
13     if (name != null && age != null && major != null) {  
14         var item = name + ", " + age + ", " + major;  
15         userList.push(item);  
16         document.querySelector("#name", "#age", "major").value = "";  
17         document.querySelector("#name", "#age", "major").focus();  
18     }  
19     showList();  
20 }
```

세 값 모두 입력되었을 때를 실행하고 싶어서 &&를 사용했습니다. 정상 작동하지만 세 값이 모두 입력되지 않아도 작동합니다. 어떤 방식으로 코드를 사용해야 할까요?

```
22 function showList () {  
23  
24     var list = "<ul>";  
25  
26     for (var i = 0; i < userList.length; i++) {  
27         list += "<li>" + userList[i] + "<span class='close' id=" + i + "> X</span></li>";  
28     }  
29     list += "</ul>";  
30  
31     document.querySelector("#userList").innerHTML = list;  
32  
33     var remove = document.querySelectorAll(".close");  
34  
35     for (var i = 0; i < remove.length; i++) {  
36         remove[i].addEventListener("click", removeList);  
37     }  
38 }  
39  
40 function removeList () {  
41     var id = this.getAttribute("id");  
42  
43     userList.splice(id, 1);  
44  
45     showList();  
46 }
```

생성된 html

사용자 정보 등록

<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="추가하기"/>
가가, 20, 국어			
나나, 21, 수학			
다다, 22, 영문			
라라, ,			



JavaScript - dom

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

컨트롤러 클래스

```
7  @Slf4j
8  @Controller
9  public class TwelfthJavaScriptTestController {

25  @GetMapping("/domTest")
26  public String getDomTest() {
27      log.info("getDomTest()");
28
29      return "/javascript/Twelfth/domTest";
30  }
```

html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Title</title>
6  </head>
7  <body>
8      <div id="detail">
9          <h2>프로그래밍 언어</h2>
10         <ul>
11             <li>Golang</li>
12             <li>Rust</li>
13             <li>C++</li>
14             <li>C#</li>
15             <li>C</li>
16             <li>Assembly</li>
17             <li>Verilog</li>
18             <li>Python</li>
19             <li>Fortran</li>
20         </ul>
21         <h3 style="visibility: hidden">세부사항</h3>
22         <p style="visibility: hidden">엄청 많다.</p>
23         <h3>너무 많아요.</h3>
24         <p>엄청나게 많네요.</p>
25     </div>
26 </body>
27 </html>
```


JavaScript - dom

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

생성된 html

프로그래밍 언어

- Golang
- Rust
- C++
- C#
- C
- Assembly
- Verilog
- Python
- Fortran

Hidden

너무 많아요.

엄청나게 많네요.

```
Elements Console Sources Network Performance Memory Application Security Lighthouse
top Filter Default levels No Issues
> document
< ▶ #document
> document.querySelector("#detail h3");
< <h3 style="visibility: hidden">세부사항</h3>
> document.querySelector("#detail h3").style;
< ▶ CSSStyleDeclaration {0: "visibility", additiveSymbols: "", alignContent: "", alignItems: "", alignSelf: "", alignmentBaseline: "", ...}
> document.querySelector("#detail h3").style.visibility;
< "hidden"
> |
```



JavaScript - dom

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

생성된 html

프로그래밍 언어

- Golang
- Rust
- C++
- C#
- C
- Assembly
- Verilog
- Python
- Fortran

세부사항

Visible

너무 많아요.

엄청나게 많네요.

```
Elements Console Sources Network Performance Memory Application Security Lighthouse
top Filter Default levels No Issues
> document
< #document
> document.querySelector("#detail h3");
< <h3 style="visibility: hidden">세부사항</h3>
> document.querySelector("#detail h3").style;
< CSSStyleDeclaration {0: "visibility", additiveSymbols: "", alignContent: "", alignItems: "", alignSelf: "", alignmentBaseline: "", ...}
> document.querySelector("#detail h3").style.visibility
< "hidden"
> document.querySelector("#detail h3").style.visibility = "visible"
< "visible"
>
```

JavaScript - dom

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

생성된 html

프로그래밍 언어

- Golang
- Rust
- C++
- C#
- C
- Assembly
- Verilog
- Python
- Fortran

세부사항

Hidden

엄청나게 많네요.

```
> document
< ▶ #document
> document.querySelector("#detail h3");
< <h3 style="visibility: hidden">세부사항</h3>
> document.querySelector("#detail h3").style;
< ▶ CSSStyleDeclaration {0: "visibility", additiveSymbols: "", align
> document.querySelector("#detail h3").style.visibility
< "hidden"
> document.querySelector("#detail h3").style.visibility = "visible"
< "visible"
> document.querySelectorAll("#detail h3")
< ▼ NodeList(2) [h3, h3] ⓘ
  ▶ 0: h3
  ▶ 1: h3
  length: 2
  __proto__: NodeList
> list = document.querySelectorAll("#detail h3")
< ▼ NodeList(2) [h3, h3] ⓘ
  ▶ 0: h3
  ▶ 1: h3
  length: 2
  __proto__: NodeList
> list[1].style.visibility = "hidden"
< "hidden"
```



JavaScript - querySelectorAll

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

컨트롤러 클래스

```
7  @Slf4j
8  @Controller
9  public class TwelfthJavaScriptTestController {

32  @GetMapping("/76Quiz")
33  public String get76Quiz() {
34      log.info("get76Quiz()");
35
36      return "/javascript/Twelfth/76Quiz";
37  }
38 }
```

여태까지 학습한 내용을 기반으로
querySelectorAll을 활용해서 버튼을 눌렀을 때
짝수 번째에 해당하는 p 태그만 visible로 변경해보자.

html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Title</title>
6  </head>
7  <body>
8      <div id="vueapp">
9          <p style="visibility: hidden">우왕 많다.</p>
10         <p style="visibility: hidden">오진다.</p>
11         <p style="visibility: hidden">책</p>
12         <p style="visibility: hidden">많다.</p>
13         <p style="visibility: hidden">살려주세요</p>
14         <p style="visibility: hidden">공부할 분량이 너무 많아요</p>
15     </div>
16     <button onclick="toggleButton()">보여주세요!</button>
17     <script src="/js/click-quiz.js"></script>
18 </body>
19 </html>
```

JavaScript - querySelectorAll

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

js

```
1  var state = 1;
2
3  function toggleButton() {
4      var list = document.querySelectorAll("#vueapp p");
5
6      for ( var i =0; i < list.length; i++) {
7          // state 값이 0이면 짝수 번째, 1이면 홀수 번째
8          // 루프 전체는 6개
9          // 1이면 1, 3, 5
10         // 0이면 경우엔 0, 2, 4
11
12         // 0이면 거짓, 0이 아니면 참
13         if(!((i + state) % 2)) {
14             list[i].style.visibility = "visible";
15         } else {
16             list[i].style.visibility = "hidden";
17         }
18     }
19
20     if (state == 1){
21         state = 0;
22     } else {
23         state = 1;
24     }
25 }
```

```
if(!((i + state) % 2)) {
    list[i].style.visibility = "visible";
} else {
    list[i].style.visibility = "hidden";
}
```

// 0이면 거짓, 0이 아니면 참
// !((0+1) %2) → 나머지 값이 1이다 → !(참) → 거짓
→ "hidden"
// !((1+1) %2) → 나머지 값이 0이다 → !(거짓) → 참
→ "visible"

... 상단 if문을 통해 visible, hidden이 결정되고,
하단 if문

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

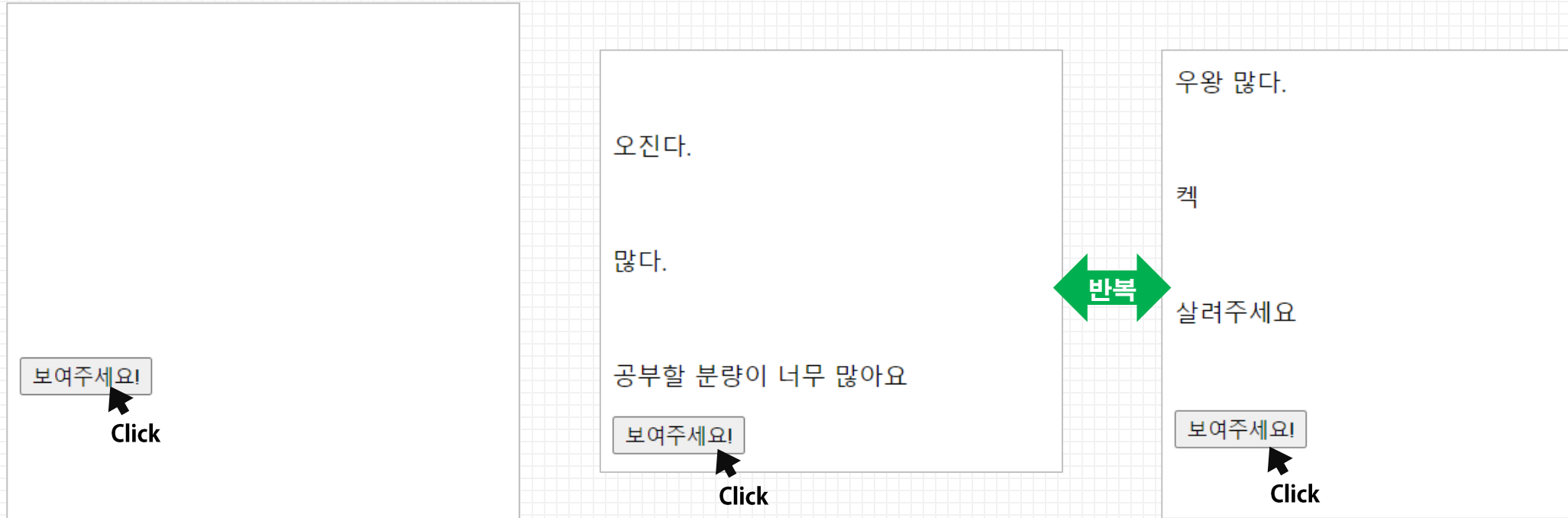
state 의 초기값이 1이므로 if문을 통해 state는 0이 되며,
for 문으로 state=0 인 상태로 상단 if문을 통해 앞이랑 다른 visible,
hidden이 결정된다.



JavaScript - querySelectorAll

링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

생성된 html





링크 <https://github.com/limcholong/LectureContents/tree/main/javascript/CholongLim/cholongtest/src/main>

컨트롤러 클래스

```
7  @Slf4j
8  @Controller
9  public class FirstVueController {
10
11      @GetMapping("/vueIndex")
12      public String getVueIndex () {
13          log.info("getVueIndex()");
14
15          return "/vue/index";
16      }
17  }
```

생성된 html

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

html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Title</title>
6      <script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
7  </head>
8  <body>
9      <div id="vueapp">
10         <h1>{{ message }}</h1>
11     </div>
12 </body>
13
14 <script>
15     var app = new Vue ({
16         el: '#vueapp',
17         data: {
18             message: '나는 겁나 길어도 이렇게 편하게 쓸 수 있어.' +
19                     '주구장창 늘여서 쓸 필요가 없단 말이지.' +
20                     '이 메시지가 또 필요하면 재활용하면 돼'
21         }
22     })
23
24 </script>
25 </html>
```