
jQuery 응용(2)

9주차_03

한 동 대 학 교
김군오 교수

학습 목표:

Jquery todo 리스트 고급 활용하기

학습내용:

- Todo 리스트 페이지 만들기
 - jQuery를 이용한 동적 화면 적용

Todo list 기능

1. 할 일을 기록할 수 있다
2. 저장된 일을 목록으로 보여준다
3. 할 일을 삭제 할 수 있다
4. 완료 상태로 만들 수 있다

Todo

Todo list

- ☐ 웹/앱 강의 듣기
- ☐ 인터넷 쇼핑 입금하기
- ☐ 투두 리스트 완성하기

Todo

```
1 <html>
2   <head>
3     <script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
4     <link rel="stylesheet" href="./todo.css" />
5   </head>
6   <body>
7     <div class="container">
8       <div class="title">
9         <h1>Todo list</h1>
10      </div>
11      <div class="contents">
12        <div class="input-div">
13          <input type="text" id="inputBox" />
14          <button id="addButton">추가하기</button>
15        </div>
16        <ul></ul>
17      </div>
18    </div>
19    <script src="./todo.js"></script>
20  </body>
21 </html>
```

Todo list

 추가하기

JSON

- Javascript Object Notation 의 약자
- 자바스크립트 객체를 자바스크립트 외부로 전송하기 위해 문자열 형태로 표현
- 데이터가 간결하고 사람이 이해하기 쉬움

JSON

자바스크립트 Object	JSON (Javascript Object Notion)
<pre>var object = { name: '이름', age: 20, major: 'GLS', friends: ['철수', '영희'] };</pre>	<pre>{ "name": "이름", "age": 20, "major": "GLS", "friends": ["철수","영희"] }</pre>

JSON

- JSON.parse 를 이용해 JSON 을 자바스크립트 객체로 변환
- JSON.stringify 를 이용해 자바스크립트 객체를 JSON 으로 변환

JSON

```
1  var obj = {  
2      a: 'JSON',  
3      b: 'NOSJ',  
4      arr: [1, 2, 'test']  
5  }  
6  console.log(obj);  
7  
8  var json = JSON.stringify(obj);  
9  console.log(json);  
10  
11 var newObj = JSON.parse(json);  
12 console.log(newObj);
```

JSON

```
> var obj = {  
  a: 'JSON',  
  b: 'NOSJ',  
  arr: [1, 2, 'test']  
}  
console.log(obj);  
  
var json = JSON.stringify(obj);  
console.log(json);  
  
var newObj = JSON.parse(json);  
console.log(newObj);
```

▶ {a: "JSON", b: "NOSJ", arr: Array(3)}

{"a":"JSON","b":"NOSJ","arr":[1,2,"test"]}

▶ {a: "JSON", b: "NOSJ", arr: Array(3)}

Todo

```
$("#addButton").click(function() {  
    var text = $("#inputBox").val();  
    todos[text] = false;  
    $("#inputBox").val("");  
    console.log(todos);  
});
```

Todo

Todo list

추가하기

ElementsConsoleSourcesNetwork

top

☐ Hide network☐ Preserve log☐ Selected context only☐ Group similar

Live reload enabled.

▶ {todo list 만들기: false}

▼ {todo list 만들기: false, jQuery 공부하기: false}

jQuery 공부하기: false

todo list 만들기: false

▶ __proto__: Object

>

Data 어트리뷰트

- HTML 엘리먼트에 데이터를 저장
- `$.data('key', 'value')` 로 값 저장
- `$.data('key')` 로 값 접근

Data 어트리뷰트

```
7      <ul></ul>
8
9      <script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
10     <script>
11         var students = [
12             { name: '김군오', id: '219xxx'},
13             { name: '홍길동', id: '210xxx'}
14         ];
15         var ul = $('ul');
16         for (let student of students) {
17             var li = $('<li>' + student.name + '</li>');
18             li.data('id', student.id);
19             li.click(function(event) {
20                 var el = $(event.target);
21                 alert(el.data('id'));
22             });
23
24             ul.append(li);
25         }
26     </script>
```

Data 어트리뷰트

- 김군오
- 홍길동

127.0.0.1:5500 내용:

219xxx

확인

Todo

```
function inputTemplate(text) {  
  var inputTag = $('<input type="checkbox" id="checkBox">');  
  inputTag.data("value", text);  
  return inputTag;  
}
```


Todo

```
function buttonTemplate(text) {  
  var buttonTag = $('<button id="deleteButton">X</button>');  
  buttonTag.data("value", text);  
  return buttonTag;  
}
```

Todo

```
function liTemplate(text) {  
  var li = $("<li></li>");  
  
  li.attr("value", text);  
  li.append(inputTemplate(text));  
  li.append(text);  
  li.append(buttonTemplate(text));  
  
  return li;  
}
```

Todo

```
$("#addButton").click(function() {  
  var text = $("#inputBox").val();  
  todos[text] = false;  
  $("#inputBox").val("");  
  console.log(todos);  
  
  $(".contents ul").append(liTemplate(text));  
});
```

Todo

Todo list

☐ 자바스크립트 공부하기

Todo

Todo list

☐ 자바스크립트 공부하기

```
Elements Console Sources Network
<html>
  <head>...</head>
  <body>
    <div class="container">
      <div class="title">...</div>
      <div class="contents">
        <div class="input-div">...</div>
        <ul>
          <li value="자바스크립트 공부하기" == $0
            <input type="checkbox" id="checkBox">
              "자바스크립트 공부하기"
            <button id="deleteButton">X</button>
          </li>
        </ul>
      </div>
    </div>
    <script src="./todo.js"></script>
    <!-- Code injected by live-server -->
    <script type="text/javascript">...</script>
  </body>
</html>
```

Todo

```
function liTemplate(text) {  
  var li = $("<li></li>");  
  
  li.attr("value", text);  
  li.append(inputTemplate(text));  
  li.append(text);  
  li.append(buttonTemplate(text));  
  
  li.click(function(event) {  
    var el = $(event.target);  
    console.log(event);  
  });  
  
  return li;  
}
```

Todo

```
todo.js:2/
▼ k.Event {originalEvent: MouseEvent, type: "click", isDefaultPrevented: f, target: li, cur
  rentTarget: li, ...}
  altKey: (...)
  bubbles: (...)
  button: (...)
  buttons: (...)
  cancelable: (...)
  changedTouches: (...)
  char: (...)
  charCode: (...)
  clientX: (...)
  clientY: (...)
  code: (...)
  ctrlKey: (...)
  ▶ currentTarget: li
    data: null
  ▶ delegateTarget: li
    detail: (...)
    eventPhase: (...)
  ▶ handleObj: {type: "click", origType: "click", data: null, handler: f, guid: 2, ...}
  ▶ isDefaultPrevented: f Se()
    jQuery34104206726864978285: true
    key: (...)
    keyCode: (...)
    metaKey: (...)
    offsetX: (...)
    offsetY: (...)
  ▶ originalEvent: MouseEvent {isTrusted: true, screenX: 289, screenY: 279, clientX: 248, ...}
    pageX: (...)
    pageY: (...)
    pointerId: (...)
    pointerType: (...)
    relatedTarget: null
    screenX: (...)
    screenY: (...)
    shiftKey: (...)
  ▼ target: li
    accessKey: ""
    assignedSlot: null
    ▶ attributeStyleMap: StylePropertyMap {size: 0}
    ▶ attributes: NamedNodeMap {0: value, value: value, length: 1}
    autocapitalize: ""
    baseURI: "http://127.0.0.1:8080/todo.html"
```

Todo

```
function liTemplate(text) {
  var li = $("<li></li>");

  li.attr("value", text);
  li.append(inputTemplate(text));
  li.append(text);
  li.append(buttonTemplate(text));

  li.click(function(event) {
    var el = $(event.target);
    console.log(el.data("value"));
  });

  return li;
}
```


Todo

```
function liTemplate(text) {
  var li = $("<li></li>");

  li.attr("value", text);
  li.append(inputTemplate(text));
  li.append(text);
  li.append(buttonTemplate(text));

  li.click(function(event) {
    var el = $(event.target);
    console.log(el.data("value"));

    if (el.is("button")) {
      delete todos[text];
      $('li[value='${text}']').remove();
      console.log(todos);
    }
  });

  return li;
}
```

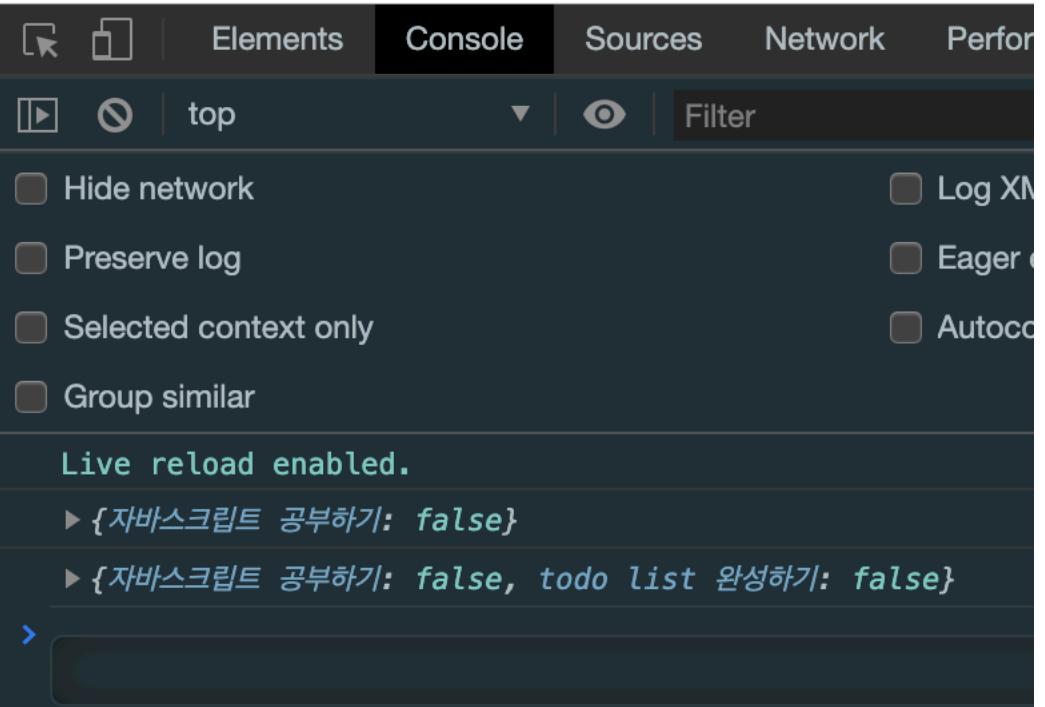
Todo

Todo list

추가하기

☐ 자바스크립트 공부하기 x

☐ todo list 완성하기 x



Todo

Todo list

☐ 자바스크립트 공부하기

ElementsConsoleSourcesNetworkPerf

top

☐ Hide network☐ Preserve log☐ Selected context only☐ Group similar

☐ Log X☐ Eager☐ Autoc

Live reload enabled.

▶ {자바스크립트 공부하기: false}

▶ {자바스크립트 공부하기: false, todo list 완성하기: false}

todo list 완성하기

▶ {자바스크립트 공부하기: false}

Todo

```
li.click(function(event) {  
  var el = $(event.target);  
  console.log(el.data("value"));  
  
  if (el.is("button")) {  
    delete todos[text];  
    $('li[value='${text}']").remove();  
  } else if (el.is("input[type='checkbox']")) {  
    var isChecked = el.is(":checked");  
    if (isChecked) {  
      $('li[value='${text}']").addClass("checked");  
      todos[text] = true;  
    } else {  
      $('li[value='${text}']").removeClass("checked");  
      todos[text] = false;  
    }  
  }  
});
```

Todo

```
.checked {  
  text-decoration: line-through;  
}
```

Todo

Todo list

추가하기

☐ 자바스크립트 공부하기

X

Elements

Console

Sources

Network

```
<html>
  <head>...</head>
  <body>
    <div class="container">
      <div class="title">...</div>
      <div class="contents">
        <div class="input-div">...</div>
        <ul>
          <li value="자바스크립트 공부하기"> == $0
            <input type="checkbox" id="checkBox">
              "자바스크립트 공부하기"
            <button id="deleteButton">X</button>
          </li>
        </ul>
      </div>
    </div>
  </body>
</html>
```

Todo

Todo list

추가하기

☒ 자바스크립트 공부하기

Elements

Console

Sources

Network

Perf

```
<html>
  <head>...</head>
  <body>
    ... <div class="container"> == $0
      <div class="title">...</div>
      <div class="contents">
        <div class="input-div">...</div>
        <ul>
          <li value="자바스크립트 공부하기" class="checked">
            <input type="checkbox" id="checkBox">
              "자바스크립트 공부하기"
            <button id="deleteButton">X</button>
          </li>
        </ul>
      </div>
    </div>
    <script src="./todo.js"></script>
    <!-- Code injected by live-server -->
    <script type="text/javascript">...</script>
  </body>
</html>
```

Todo

Todo list

추가하기

☐ 자바스크립트 공부하기

X

Elements

Console

Sources

Network

```
<html>
  <head>...</head>
  <body>
    ... <div class="container"> == $0
      <div class="title">...</div>
      <div class="contents">
        <div class="input-div">...</div>
        <ul>
          <li value="자바스크립트 공부하기" class>
            <input type="checkbox" id="checkBox">
              "자바스크립트 공부하기"
            <button id="deleteButton">X</button>
          </li>
        </ul>
      </div>
    </div>
    <script src="./todo.js"></script>
    <!-- Code injected by live-server -->
    <script type="text/javascript">...</script>
  </body>
</html>
```


요약

- Todo 리스트 페이지 만들기
 - jQuery를 이용한 동적 화면 적용

차시 예고

- 10-1 : Jquery AJAX 소개 및 활용하기
 - 웹 서버와 브라우저 통신 방법
 - http 프로토콜
 - Jquery AJAX 기술 소개
 - Jquery AJAX 활용

강의를 마치겠습니다
수고하셨습니다

9주차_03 jQuery 응용(2)