

---

# HTML에서 웹앱까지

## 12주차\_01

한 동 대 학 교  
김군오 교수

# 학습 목표:

## nodejs Todo list 프로그램 만들기

---

### 학습내용:

- NodeJS Todo list 프로그램
  - Todo list API 만들기
  - Todo list 클라이언트 코드 수정

# NodeJS To-do list

---

- todo list 웹 페이지를 Node JS 서버를 통해 호스팅
- todo list 저장 및 불러오기 API 구현

# Todo list API

---

Nodejs의 index.js 파일

```
1  var express = require("express");
2  var app = express();
3
4  var todos = {};
5
6  app.use(express.json());
7  app.use(express.static("public"));
8
9  app.listen(3000, function() {
10 |   console.log("App listening on port 3000!");
11 | });
12
```

# Todo list API

---

```
12
13 app.post("/todos", function(req, res) {
14     todos = req.body.todos;
15     console.log(req.body);
16     res.status(200).send({ message: "success" });
17 });
18
```

# Todo list API

---

```
19 app.get("/todos", function(req, res) {  
20   res.status(200).send(todos);  
21 });  
22
```

# Todo list API

---

```
3  
4  var todos = { "자바스크립트 공부하기": true };  
5
```

# Todo list API

---

```
3  
4 var todos = { "자바스크립트 공부하기": true };  
5
```



localhost:3000/todos

▼ {  
 ...  
}

"자바스크립트 공부하기": true



# Todo list 클라이언트

---

```
54 function saveTodos() {  
55     $.ajax({  
56         url: "http://localhost:3000/todos",  
57         method: "POST",  
58         data: JSON.stringify({ todos: todos }),  
59         dataType: "json",  
60         contentType: "application/json"  
61     }).done(function() {  
62         console.log("POST done");  
63     });  
64 }
```

# Todo list 클라이언트

---

```
42 function inputTemplate(text, checked) {  
43     var inputTag = $('<input type="checkbox" id="checkBox">');  
44     inputTag.data("value", text);  
45     inputTag.attr("checked", checked);  
46     return inputTag;  
47 }
```

# Todo list 클라이언트

---

```
12 function liTemplate(text, checked) {  
13     var li = $("<li></li>");  
14  
15     li.attr("value", text);  
16     li.append(inputTemplate(text, checked));  
17     li.append(text);  
18     li.append(buttonTemplate(text));  
19 }
```

# Todo list 클라이언트

---

```
3  $("#addButton").click(function() {  
4      var text = $("#inputBox").val();  
5      todos[text] = false;  
6      $("#inputBox").val("");  
7      console.log(todos);  
8  
9      $(".contents ul").append(liTemplate(text, false));  
10 });
```

# Todo list 클라이언트

---

```
3  $("#addButton").click(function() {  
4      var text = $("#inputBox").val();  
5      todos[text] = false;  
6      $("#inputBox").val("");  
7      console.log(todos);  
8  
9      $(".contents ul").append(liTemplate(text, false));  
10     saveTodos();  
11 });
```

# Todo list 클라이언트

---

```
13 function liTemplate(text, checked) {
14     var li = $("- </li>");
15
16     li.attr("value", text);
17     li.append(inputTemplate(text, checked));
18     li.append(text);
19     li.append(buttonTemplate(text));
20
21     li.click(function(event) {
22         var el = $(event.target);
23         console.log(el.data("value"));
24
25         if (el.is("button")) {
26             delete todos[text];
27             $('li[value='${text}']").remove();
28         } else if (el.is("input[type='checkbox']")) {
29             var isChecked = el.is(":checked");
30             if (isChecked) {
31                 $('li[value='${text}']").addClass("checked");
32                 todos[text] = true;
33             } else {
34                 $('li[value='${text}']").removeClass("checked");
35                 todos[text] = false;
36             }
37         }
38
39         saveTodos()
40     });

```

# Todo list 클라이언트

---

```
1  var todos = {};  
2  
3  $.ajax("http://localhost:3000/todos").done(function(result) {  
4      console.log(result);  
5  
6      todos = result;  
7      for (const todo of Object.keys(todos)) {  
8          $(".contents ul").append(liTemplate(todo, todos[todo]));  
9      }  
10 });
```

# todo list

## Todo list

추가하기

☐ 자바스크립트 공부하기 x

☐ todo list 완성하기 x

Elements

Console

Sources

Network

Performance

>>

⋮

×

▶

⌵

top

▼

👁

Filter

Default levels ▼

⚙

☐ Hide network

☐ Preserve log

☐ Selected context only

☐ Group similar

☐ Log XMLHttpRequests

☐ Eager evaluation

☐ Autocomplete from history

▶ Object [todo.js:4](#)

▶ {자바스크립트 공부하기: false} [todo.js:16](#)

POST done [todo.js:75](#)

▶ {자바스크립트 공부하기: false, todo list 완성하기: false} [todo.js:16](#)

POST done [todo.js:75](#)

>



# todo list

---

문제 출력 디버그 콘솔 터미널

```
14:37:27 C ...Dropbox/kmooc/11강 o v8.11.2
$ node index.js
App listening on port 3000!
{ todos: { '자바스크립트 공부하기': false } }
{ todos: { '자바스크립트 공부하기': false, 'todo list 완성하기': false } }
█
```

# todo list

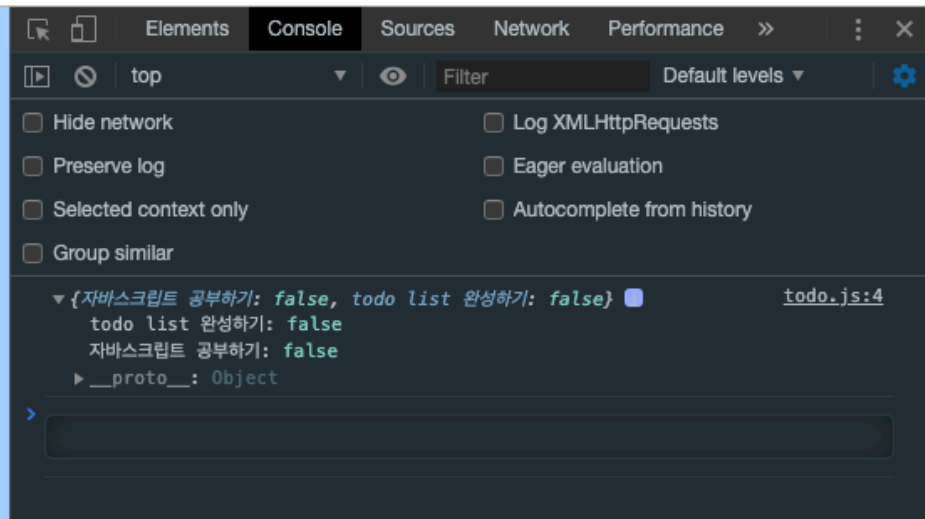
---

## Todo list

추가하기

☐ 자바스크립트 공부하기 x

☐ todo list 완성하기 x



# 요약

---

- NodeJS Todo list 프로그램
  - Todo list API 만들기
  - Todo list 클라이언트 코드 수정

# 차시 예고

---

- 12-2 : 동시통역 채팅 프로그램 만들기
  - 동시통역 채팅 프로그램

---

**강의를 마치겠습니다**  
**수고하셨습니다**

11주차\_03 채팅, Todo list 만들기