

강좌3 - NodeJS - 6년 전 등록 / 4년 전 수정

## HTML 전송하기

Node.js 교과서 책이 출간되었습니다. 이 포스팅보다는 책이나 동영상 강좌를 보시는 것을 추천합니다.



안녕하세요! 이번 시간에는 **Node.js**로 text 대신에 HTML을 전송하는 방법을 알아보겠습니다! 지난 시간에는 간단한 text를 전송했는데, text로 HTML 구조를 만들어서 전송하기는 힘들겠죠? 그래서 아예 통째로 HTML 파일을 전송하는 방법이 있습니다.

일단 지난 시간 만들었던 `server.js`와 같은 폴더 위치에 `main.html`을 만듭니다. (이름은 상관 없습니다) HTML 파일의 내용물은 **자바스크립트 강좌28 턴제 게임 만들기**의 HTML을 쓰겠습니다.

그리고 실제 서비스에서는 여러 페이지를 사용하기 때문에 저희도 여러 페이지 구현을 위한 `about.html`도 만들겠습니다. 게임에 관한 정보가 나와있는 페이지입니다.

`about.html`

```

1 <html>
2 <head>
3   <meta charset="utf-8" />
4   <title>About</title>
5 </head>
6 <body>
7   <div>제작자 : ZeroCho</div>
8 </body>
9 </html>
```

[목록](#)




server.js

```

1  const http = require('http');
2  const url = require('url');
3  const fs = require('fs');
4  http.createServer((request, response) => {
5      const path = url.parse(request.url, true).pathname; // url에서 path 추출
6      if (request.method === 'GET') { // GET 요청이면
7          if (path === '/about') { // 주소가 /about이면
8              response.writeHead(200, {'Content-Type': 'text/html'}); // header 설정
9              fs.readFile(__dirname + '/about.html', (err, data) => { // 파일 읽는 메소드
10                  if (err) {
11                      return console.error(err); // 에러 발생시 에러 기록하고 종료
12                  }
13                  response.end(data, 'utf-8'); // 브라우저로 전송
14              });
15          } else if (path === '/') { // 주소가 /이면
16              response.writeHead(200, {'Content-Type': 'text/html'});
17              fs.readFile(__dirname + '/main.html', (err, data) => {
18                  if (err) {
19                      return console.error(err);
20                  }
21                  response.end(data, 'utf-8');
22              });
23          } else { // 매칭되는 주소가 없으면
24              response.statusCode = 404; // 404 상태 코드
25              response.end('주소가 없습니다');
26          }
27      }
28  }).listen(8080);

```

**fs**와 **url** 모듈을 추가했습니다. fs 모듈은 파일을 읽고 쓰는 모듈입니다. 여기서는 **readFile** 메소드로 HTML 파일을 읽어서 읽은 데이터를 브라우저로 보내고 있는데요. url 모듈은 주소를 분석하는 모듈입니다. 이 모듈을 사용해서 url에 따라 다른 HTML을 전송하는 **라우팅**을 구현하고 있습니다.

**url.parse** 메소드로 요청한 주소 `request.url`을 분석할 수 있습니다. path 변수에 분석한 url을 담고, GET 요청(기본적으로 GET 요청입니다. 폼의 POST, GET 요청과 같은 겁니다)일 때 url에 따라 각각 다른 HTML 파일을 읽습니다.

그 전에 `response.writeHead` 메소드로 `response.setHeader` 메소드와 같이 header를 설정합니다. 파일을 다 읽은 후 `response.end` 메소드로 읽은 data를 전송하고 있습니다. `__dirname`은 현재 프로젝트의 경로를 의미합니다. `__dirname + '/main.html'`은 현재 프로젝트 경로의 main.html의 주소겠죠? 만약 요청 주소가 '/'도 아니고 '/about'도 아닌 경우에는 **404 NOT FOUND** 에러를 전송합니다. 해당하는 주소에 맞는 파일이 없을 때 발생하는 에러입니다. 몇 번 보신 적이 있을 겁니다.

**cmd**로 서버가 있는 곳으로 가서 **node server**하면 다음과 같이 브라우저 주소를 쳤을 때 경로에 맞는 HTML 페이지가 보여집니다.



목록



v7.3.0

로그인      회원가입

영웅 이름을 입력하세요!

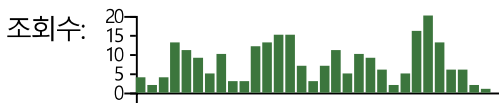
시작

← → ↺ localhost:8080/about

 열  Facebook  네이버 :: 나의 경쟁력

제작자: ZeroCho

이렇게 HTML을 전송해보았는데요. 코드가 너무 반복되고, 페이지 별로 HTML을 따로 만들어야하며, 이를 각각 주소와 일일이 연결해야하는 단점이 있습니다. 페이지가 수십 페이지 이상 늘어난다고 생각해보세요. 매번 GET 요청인지 확인하고, 주소를 파악하고, fs로 매칭되는 파일을 읽어서 브라우저로 전송해야 합니다. 이렇게 하서는 실제 서비스는 무리겠죠? 다음 시간에는 이 귀찮은 과정을 간단히 해결해 줄 **Express** 프레임워크에 대해서 알아보시다!

이전글: [서버 만들기](#)다음글: [Express 프레임워크](#)

투표로 게시글에 관해 피드백을 해주시면 게시글 수정 시 반영됩니다. 오류가 있다면 어떤 부분에 오류가 있는지도 알려주세요! 잘못된 정보가 퍼져나가지 않도록 도와주세요.

만족해요

투표

설명이 부족해요

투표

너무 어려워요

투표

오류가 있는 거 같아요

투표

결과 보기

Copyright 2016- ZeroCho. 무단 전재 및 재배포 금지. 출처 표기 시 인용 가능.


이메일

비밀번호(수정 및 삭제 시 사용)

댓글을 입력하세요. 근거없는 비난 대신 날카로운 피드백을 원합니다. 답글이 달리면 삭제할 수 없습니다.

등록

7개의 댓글이 있습니다.

+  익명 안녕하세요, 책을보며 공부하고있습니다.

↶ 2년 전

궁금한게 있어서 질문드립니다.

`http.createServer((req, res) => {``if(req.method === 'GET'){``if(req.url === '/'){``fs.readFile('./practice4.3/restFront.html', (err, data) => {`

목록



v7.3.0

로그인      회원가입

```
if(req.method === 'GET'){
  if(req.url === '/'){
    return fs.readFile('./practice4.3/restFront.html', (err, data) => {
```

sample 소스코드를보니깐 저렇게 return이 있는것도 있고 없는것도 있던데, return 유무에 따라 어떤 차이가 있는지 궁금합니다.



**ZeroCho Blog**  
 2,029 좋아요

페이지 좋아요

메시지 보내기

타임라인

메시지


**ZeroCho Blog**  
 약 한 달 전

[https://www.zerocho.com/cat.../.../post/61d064b6a7f9490004dd9358](https://www.zerocho.com/cat.../post/61d064b6a7f9490004dd9358)



ZEROCHO.COM  
 (etc) 2021년 ...  
 [2020년 블로그 ...

2개

댓글 달기

1개


**ZeroCho Blog**  
 약 7개월 전



목록

