

2022 퍼블리셔 부트캠프 013

- Node.js / Express, localhost:3000

!Tips & Links

※ 강의 <https://www.udemy.com/course/the-complete-web-development-bootcamp/>

※ Node.js <https://nodejs.org/api/index.html> 📖 native modules list:

※ NPM <https://www.npmjs.com/> 📖 search packages

※ State of JS <https://stateofjs.com/>

<https://2020.stateofjs.com/en-US/technologies/> 글로벌 개발자 서베이, 통계 및 추세 확인

※ 버전힌트: //jshint esversion:6

※ Back-end tools: Jave, Ruby on Rails, PHP, ASP.Net, Node.js, Python....

Node.js (JavaScript): js가 browser뿐만 아니라 하드웨어, 네트워크 등에 접근할 수 있게 함.
서버, 사용자컴퓨터에서 동작할 수 있음
server-side(process) => client-side(print) 보안, 처리속도
able to write code that can interact directly with the computer independently,
can access local files, listen to network traffic, manipulate databases

※ node REPL

read-eval-print loop
(program in a browser)

REPL	node쳐서 active... >
	command 치다가 TAB 2번 📖 show possible commands
	나가기: .exit 또는 Ctrl + C 2번

※ node.js modules

(예제: superheroes)

Internal module	module의 method 사용하려면 require(); 해야함 const ~= require('module'); ~~.copyFileSync('file1', 'file2') + node internal_file.js 로 실행
External module	npm // packages: reusable code

NPM

npm init : package파일 initialize, package.json

atom . : 프로젝트 폴더 열기 start atom ~.js : 파일 열기

Express

npm install express Express는 node.js의 프레임워크, --save: 디펜던시 추가, 필요x

로컬 서버

```
const express=require('express');  
const app=express(); << 관습  
app.listen(3000, function(){ ~= })
```

app.listen(3000) == localhost:3000

콜백함수로 서버연결 확인(log)

Ctrl + C : 연결 끊기

```
app.get('/', function(request, response){  
  console.log(request) << log기록  
  response.send('<h1>Hello, world!</h1>')  
})
```

※ '/': 홈, 연결됨

※ req=request, res=response

※ 서버에 get없을 때 브라우저 에러: Cannot GET /

/* -----<lang: en>----- */

> angle bracket // callback함수 >> higher order함수에 의해 trigger되면 작동하는 함수

sever started port 3000 => localhost:3000 // e, evt: event / req: request / res=response