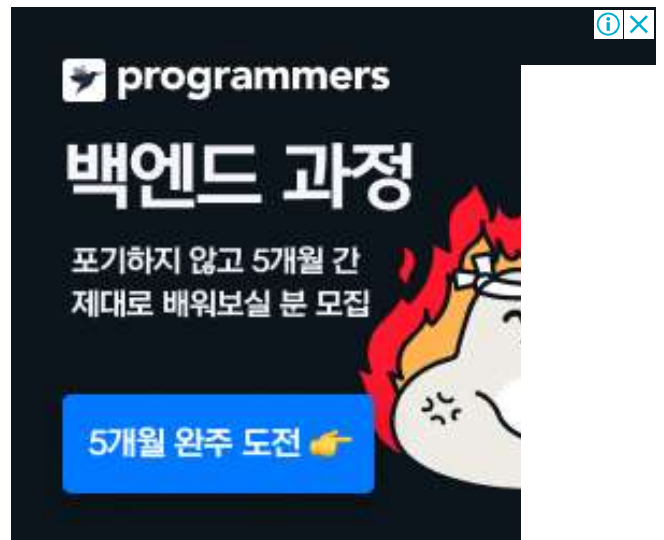
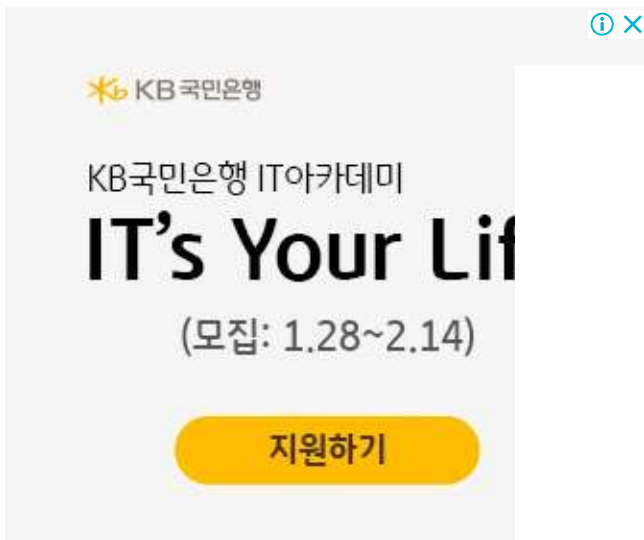


강좌1 - NodeJS - 6년 전 등록 / 4년 전 수정

Node.js와 npm

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



안녕하세요. 자바스크립트, ES2015에 이은 세 번째 강의 주제로 **Node.js**를 선정했습니다! (현재 Node.js 교과서(길벗 출판사)를 출간했습니다. 블로그 강좌와 내용이 다르며, 블로그의 강좌보다 훨씬 더 퀄리티가 높습니다)

Node.js

Node.js는 자바스크립트를 브라우저 외의 다른 환경에서도 사용할 수 있게 해주는 **런타임**입니다. **http서버**가 내장되어 있기 때문에 보통은 **서버**로 많이 사용합니다. 하지만 응용 프로그램을 실행하는 데 쓸 수도 있습니다. 웹팩, 바벨과 같은 도구들과 일렉트론 같이 자바스크립트로 데스크탑 프로그램을 만드는 프레임워크가 모두 Node.js를 기반으로 돌아갑니다. 이 강좌는 웹서버로서의 Node.js를 주로 다룰 겁니다.

흔히들 생각하기로는 서버가 엄청 복잡하다고 생각하지만, **Node.js**에서는 10줄도 안 되는 코드로 서버를 구동할 수 있습니다. 게다가 자바스크립트 문법을 사용하기 때문에 다른 서버 언어를 배우지 않아도 돼서 부담이 훨씬 덜하죠.

어차피 **클라이언트**(웹 브라우저) 상에서는 자바스크립트를 쓸 수 밖에 없습니다. 그런데 서버 언어는 JSP나 PHP, ASP, Python, Ruby 등의 다른 언어를 사용했기 때문에 개발자 입장에서는 새로운 언어를 배워야한다는 부담감이 컸습니다. 거기에 Apache 같은 웹서버와 Tomcat같은 웹 어플리케이션 서버를 따로 관리해야 해서 배워야할 것이 너무 많았습니



으니 금상첨화죠. (LESS라고 하는 자바스크립트로 된 CSS도 있습니다.)

Node.js를 하려면 일단 설치부터 해야합니다. 자바스크립트는 기존 브라우저에서도 지원을 하는 거지만, Node.js는 서버이기 때문에 설치가 필요합니다.

여기를 눌러 Node.js 공식 홈페이지로 갑시다. 현재 8버전 대 **LTS**와 10버전 대 **Current**가 있네요. 실험적인 10버전 대를 받는 게 공부하는 입장에서는 좋을 것 같네요. (버전이 혹시 바뀌면 댓글로 알려주세요!) 8버전은 2020년 4월에 지원이 끊깁니다. 하지만 사정이 있어 2019년 12월에 끊기는 것으로 앞당겨졌습니다. 10버전은 2021년 4월에 지원이 끊깁니다. 나중에 나올 12버전은 2022년에 끊기겠죠? 참고로 6버전은 2019년 4월에 끊깁니다. 즉, 6버전도 아직 사용가능하다는 뜻이죠.

Node.js는 이렇게 여러 가지 버전을 동시에 지원합니다. 버전이 너무 빠르게 바뀌면 기업들이 따라가기 힘들니까요. 옛날 버전은 대략 3년 정도 지원 기간을 둡니다. 이렇게 여러 버전이 동시에 지원되면서 순차적으로 종료해나가는 시스템을 LTS라고 합니다.

근데 홀수 버전은 어디갔을까요? 사실 6개월마다 새로운 버전이 나오긴 합니다. 10버전 출시 후 6개월 후에 11버전이 나올 겁니다. 하지만 홀수 버전은 실험적인 버전이라고 생각하시면 됩니다. LTS를 하지 않는데요. 8 이후에 LTS에 돌입해 기업들을 지원하는 버전은 9가 아니라 10입니다.

다운로드 받은 설치 파일을 실행한 뒤 계속 Next를 눌러 설치를 하고, 설치 완료 후에는 **명령프롬프트**를 띄워봅니다. 명령프롬프트는 windows 메뉴 검색에 **cmd**를 치면 나옵니다. 까만 화면에 커서가 깜빡거리고 있을텐데요. **node -v**를 쳐서 버전이 나오는지 확인해보세요(스크린샷에서는 8버전으로 되어있는데 10버전이 나오면 됩니다).

```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\zerocho\WebstormProjects\zerocho>node -v
v8.1.2
```

npm

Node.js와 땔래야 땔 수 없는 관계로 **npm**이라는 게 있습니다. node package manager의 줄임말인데요. Node.js에서는 자주 쓰이고 재사용되는 자바스크립트 코드들을 **패키지**로 만들어서 사용할 수 있습니다. 그러한 패키지를 모아놓은 저장소가 npm입니다. 만약 자바스크립트로 무언가를 만들고 싶은데 귀찮거나, 너무 어려워서 만들지 못하겠다면 **npm 홈페이지**에 가서 쇼핑하듯 미리 만들어져 있는 패키지를 고르면 됩니다. 저도 reactJS를 위한 패키지를 몇 개 만들었습니다. 제 프로필 메뉴에 가면 볼 수 있습니다.

Node.js를 설치하면 자동으로 npm이 설치됩니다. npm도 **npm -v**로 버전을 체크해봅시다. 버전은 사람마다 다를 수 있습니다. 버전이 좀 낮다 싶으면 최신 버전으로 업데이트도 가능합니다! 웬만해서는 최신 버전이 좋습니다. **npm**



이제 다른 사람의 패키지들을 **npm install [패키지 이름]**을 명령으로 **cmd**에서 설치할 수 있습니다. 그 후 `const 변수 이름 = require(패키지이름);` 하면 해당 변수 이름으로 패키지를 사용할 수 있습니다. 다음 시간부터 이렇게 require해서 쓰는 패키지들을 보게 될 겁니다.

이렇게 **npm**에서 **패키지**를 사용할 것이라면 패키지들을 체계적으로 관리할 필요가 있습니다. 실제 서비스에서는 패키지를 수십, 수백 개 가량 다운받게 되기 때문에 각각의 패키지에 대한 버전을 기록해야 합니다. 버전이 중요한 이유는 버전이 업그레이드 되면서 이전 버전과 호환이 안 될 수도 있기 때문입니다. 그래서 패키지명과 함께 패키지 버전도 함께 기록할 필요가 있습니다. 그 역할을 하는 게 **package.json** 파일입니다. npm은 **package.json**을 보고 '이 프로젝트는 이런 패키지를 필요로 하는구나'하고 알아차립니다.

package.json을 만드는 방법은 두 가지가 있는데, 하나는 그냥 직접 만드는 것이고 다른 하나는 **cmd**에 **npm init** 명령어를 치는 겁니다. 지금 한 번 입력해볼까요? **npm init**은 자신이 만들 프로젝트 폴더 안으로 **cmd**를 통해서 이동한 후 입력해야 합니다. 폴더 안으로 이동하는 방법은 **cd [폴더경로]** 하시면 됩니다. 잘 모르겠으면 명령프롬프트 명령어를 검색해보세요. 예를 들어 C:\program files\nodejs로 가고 싶다면 **cd C:\program files\nodejs**를 치면 됩니다.

```
c:\Users\ZeroCho>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg> --save` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
name: (ZeroCho)
Sorry, name can no longer contain capital letters.
name: (ZeroCho) zerocho
version: (1.0.0)
description:
```

npm init을 하면 영어로 설명이 나온 후(아래서 프로그래밍을 하려면 영어를 잘해야 합니다...) 프로젝트 이름, 버전, 설명, 라이선스 등등을 물어봅니다. 다 입력하면 지금까지 입력한 정보가 들어있는 **package.json**을 만들어줍니다. 아직은 쓸 일이 없으니 갖고만 있도록 하죠. 만들어진 **package.json**은 다음과 같은 형식입니다.

```
1 {
2   "name": "zerocho",
3   "version": "1.0.0",
4   "description": "npm description",
5   "main": "server.js",
6   "scripts": {
7     "test": "echo \"Error: no test specified\" && exit 1",
8     "start": "node server.js"
9   },
10  "keywords": [
```

v7.3.0

로그인      회원가입

```

16   "author": "ZeroCho",
17   "license": "ISC"
18 }

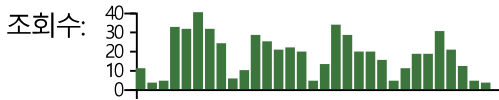
```

아직은 자세히 알 필요가 없습니다. 나중에 `package.json`에 대해 설명하는 **강좌**가 있으니 그것을 참고하세요.

본격적으로 **서버**를 만드는 것은 다음시간부터입니다!

이전글: 없음

다음글: **서버 만들기**



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

만족해요	24(80.00%)
설명이 부족해요	3(10.00%)
너무 어려워요	3(10.00%)
오류가 있는 거 같아요	0(0.00%)
총 투표 수	30

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

6개의 댓글이 있습니다.

-  **익명** 빼애액 18년 6월 기준 10버전 써서 따라가도 되나요  4년 전
-  **ZeroCho** 빼애액 노드 10버전, npm 6버전으로 하시면 됩니다.  4년 전
-  **floorViper** 공부하고갑니다.  4년 전
-  **익명** 웹서버로서의 -> 웹서버로서의, 사소하지만, 교정에 도움이 되길 바랍니다.  4년 전
-  **ZeroCho** 감사합니다. 책에서는 로서로 썼는데 블로그에서는 신경을 못 썼네요  4년 전
-  **Legacy** NodeJS 강좌를 찾다가 이 사이트를 알게 되었습니다. 정말 좋은 강좌가 많이 있네요. 열심히 배워 보겠습니다!  4년 전
-  **ZeroCho** 반갑습니다! 감사합니다~  4년 전
-  **Taegyun Sooran Kim** package.json에 16번째 줄 author에 string이 안 닫혀있어요~  5년 전
-  **ZeroCho** 감사하네요  5년 전

목록



>

다양한 서비스를 직접 만드는, 실무에 가장 가까운 강좌

페이지 좋아요

메시지 보내기

타임라인

메시지



ZeroCho Blog

약 한 달 전

<https://www.zerocho.com/cat.../post/61d064b6a7f9490004dd9358>



ZEROCHO.COM

(etc) 2021년 ...

[2020년 블로그 ...

2개

댓글 달기

1개



ZeroCho Blog

약 7개월 전



목록







>

<https://www.zerocho.com/category/NodeJS/post/57387cb8715202c8679b3af1>

5/5