NodeJS란?

Javascript outside browser

Chrome V8 JavaScript engine로 빌드 된 자바스크립트 런타임

브라우저 밖에서 돌아가는 JavaScript. 95년 JS가 처음 만들어졌을 때, JS는 브라우저에서만 작동. 브라우저와 좀 더 상호작용하기 위해 만들어진 언어였지만 현재 가장 많이 사용하는 언어가 되었다. 브라우저만 있다면 누구든 사용할 수 있었던 언어. 하지만 유일한 언어이기도 했다.

Ryan Dahl이 브라우저와 분리된 환경에서도 JS를 이용하기 위해 만듦

NodeJS를 이용해 React Native와 electron 사용이 가능해졌고, electron으로 만들어진 데스크톱 앱에는 VS code, slack, Figma 등이 있다. React Native 이용 앱은 Facebook, Instagram, Discord 등.

NPM이란?

JavaScript 언어를 위한 패키지 매니저로, node.js설치 시 함께 설치되며, node.js와 상호작용을 하여 이용. 꼭! 함께 써야함

여러 개발자들이 패키지를 만들어 공유 할 수 있고, 다운로드 받아 다른 개발자들이 사용 가능

SET UP

1. 폴더 생성 후 package.json 생성.

package.json이란? 프로그래머가 파일에 정보를 저장하기 위해 만든 방식으로, 가장 처음으로만들어야 하는 파일.node.js에서 이름은 꼭 package.json이어야만 한다. 대소문자 구분하니 꼭 소문자로 작성!

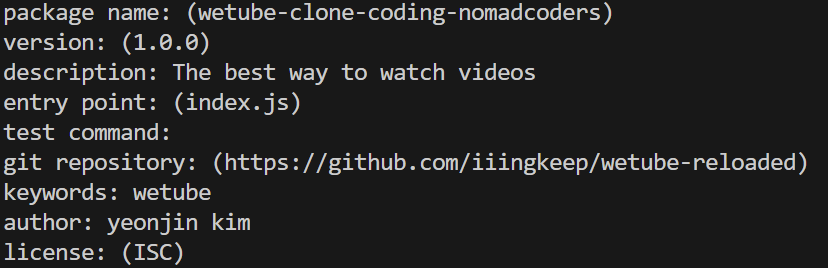
이 package.json은 직접 작성하면 실수가 있을 수 있으니, 나중에 도움을 받아 설치

2. git 등록

-폴더에서 git init으로 repository 생성

-깃허브에 repository를 생성 후, git remote add origin 깃헙url 등록

-npm init 입력 시 package.json의 내용에 담을 항목들이 나온다. 정보를 입력해주면 된다.



모두 입력 한 뒤, package.json 파일이 자동 생성되며 안에 입력한 항목이 담겨있다.

{

  "name": "wetube-clone-coding-nomadcoders",

  "version": "1.0.0",

  "description": "The best way to watch videos",

  "main": "index.js",

  "scripts": {

    "test": "echo \"Error: no test specified\" && exit 1"

  },

  "repository": {

    "type": "git",

    "url": "(https://github.com/iiingkeep/wetube-reloaded)"

  },

  "keywords": [

    "wetube"

  ],

  "author": "yeonjin kim",

  "license": "ISC"

}

Main은 main file이 되는 것을 말하며, index.js라고 써져 있으므로 index.js파일을 생성한다. Main이 필수는 아니니 지워줘도 된다.

시작은 이렇게, package.json파일 + index.js 두개의 파일로!

3. node.js 실행

첫 번째: 터미널에 node index.js 처럼 node 파일명 을 입력하면 실행

두 번째: package.json이 실행하도록 설정

"scripts": {

    "win": "node index.js"

  },

위와 같이 입력한다면 npm run win 입력시 node index.js를 입력하는 것과 같아 node.js 실행

이 명령어는 package.json이 있는 디렉터리에서만 작동!

4. npm i

프롬프트에 npm i 설치할 패키지이름 을 입력시, package-lock.json파일과 node\_modules 폴더가 생성 된다. Node\_modules 폴더에는 npm으로 설치한 모든 패키지 저장. 하나의 package를 설치 시, 다른 여러가지 패키지들이 함께 필요하며 각각의 package.json 파일을 가지고 있다.

또한, package.json의 dependencies부분에 설치한 패키지가 자동으로 등록된다.

 "dependencies": {

    "express": "^4.19.2"

  }

이 dependencies는 필요한 패키지들을 담아뒀다가 node\_modules폴더 또는 package-lock.json이 삭제됐더라도 npm i 만 입력하면 자동으로 설치되도록 한다.

Package.json이 열려있을 때 npm i를 하다가 버전이 충돌/변경 될 수 있으니 꼭 package.json파일은 닫고 npm i를 하도록 한다

5. package-lock.json

패키지들을 안전하게 관리. 즉, 다른 사람이 내 파일을 받아 npm i를 한다면 패키지의 버전을 모두 동일하게 맞춰주어 다른 컴퓨터에서도 잘 작동하게 한다