

## 1 / 10

```

leekh@my-server:~
~ » wget -qO- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh | bash
=> nvm is already installed in /home/leekh/.nvm, trying to update using git
=> => Compressing and cleaning up git repository

=> Appending nvm source string to /home/leekh/.zshrc
=> Appending bash_completion source string to /home/leekh/.zshrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh" # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion" # This loads nvm bash_completion

```

### 3) 설치 결과 확인

설치 완료 여부를 확인하기 위해 버전을 확인한다.

```
$ nvm --version
```

```

leekh@my-server:~ -- 95x7
~ » source ~/.zshrc
~ » nvm --version
0.39.1
~ »

```

## 3. NVM을 사용한 Node 버전 관리

### 1) 최신 버전 설치하기

```
$ nvm install
```

### 2) 특정 버전 설치하기

```

$ nvm install 버전
$ nvm install 12.22.9

```

### 3) 설치되어 있는 Node 목록 확인하기

버전이 여러개 나오고 특정 버전 앞에 \* 표시가 있다면 현재 \* 표시 버전이 사용중인 상태임

```
$ nvm ls
```

#### 4) 특정 버전으로 변경하기

```
$ nvm use 버전
$ nvm use 12.22.7
```

#### 5) 설치한 특정 노드버전 삭제

```
$ nvm uninstall 버전
$ nvm uninstall 12.22.6
```

#### 5) node 설치 경로 확인

```
$ which node
```

## #02. Node.js 설치하기

### 1. Node.js 설치

NVM을 사용하여 Node를 설치한다. 기본적으로 가장 최신 버전을 설치하게 된다.

```
$ nvm install node
```

```
leekh@my-server:~ — 78×11

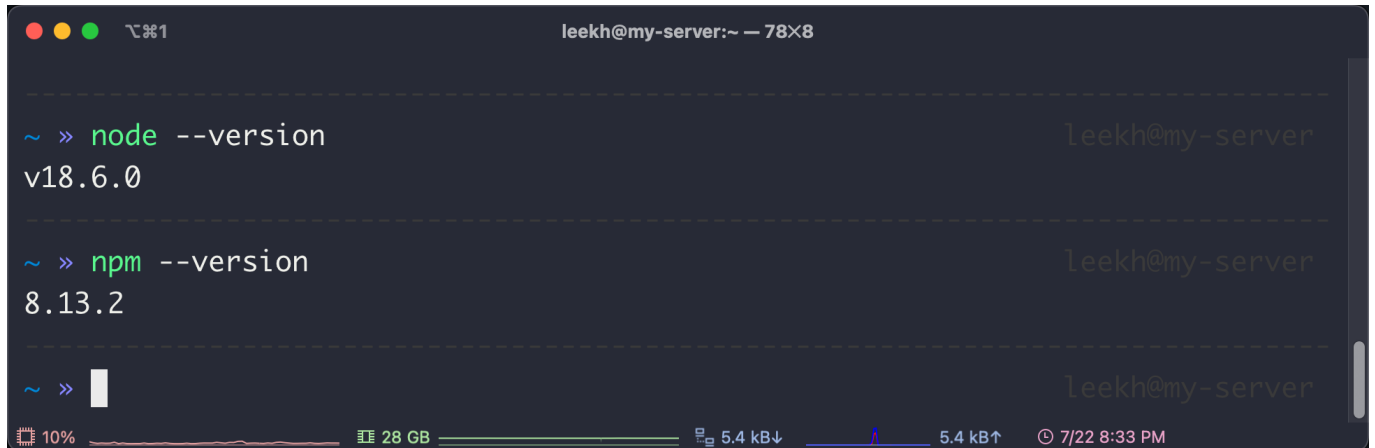
~ >> nvm install node
Downloading and installing node v18.6.0...
Downloading https://nodejs.org/dist/v18.6.0/node-v18.6.0-linux-arm64.tar.xz...
##### 100.0%
Computing checksum with sha256sum
Checksums matched!
Now using node v18.6.0 (npm v8.13.2)
Creating default alias: default -> node (-> v18.6.0)

~ >> 
```

## 2. 설치 결과 확인

node와 npm의 버전을 각각 확인한다.

```
$ node --version
$ npm --version
```



```
leekh@my-server:~ -- 78x8
~ >> node --version
v18.6.0
~ >> npm --version
8.13.2
~ >> 
```

## 3. Yarn 설치하기

Yarn은 Node.js 자바스크립트 런타임 환경을 위해 페이스북이 2016년 개발한 소프트웨어 패키지 시스템

npm 패키지 관리자의 대안으로서 대형 코드의 일관성, 보안, 성능 문제를 해결하고자 개발됨

```
$ npm install -g yarn
```



```
leekh@my-server:~ -- 78x8
~ >> npm install -g yarn
added 1 package, and audited 2 packages in 373ms
found 0 vulnerabilities
~ >> 
```

## #03. Express 기반 백엔드 가동하기

### 1) 소스코드 업로드

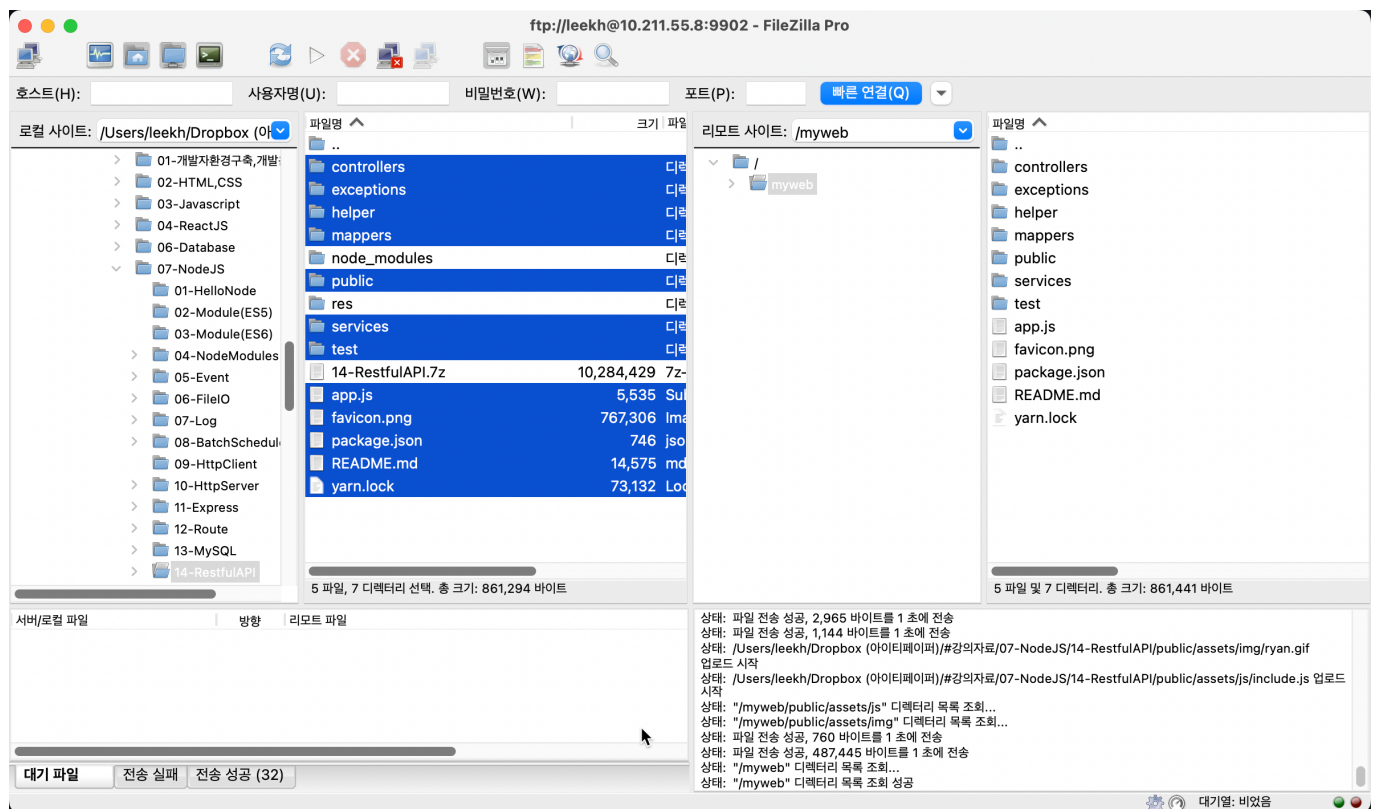
FTP등의 프로그램을 사용하여 완성된 Express 백엔드 프로젝트를 Linux의 적절한 위치에 업로드한다.

이 때 `node_modules` 디렉토리는 업로드하지 않는다.

`package.json`은 반드시 포함되어야 한다.

여기서는 `~/myweb` 경로에 업로드 하였다.

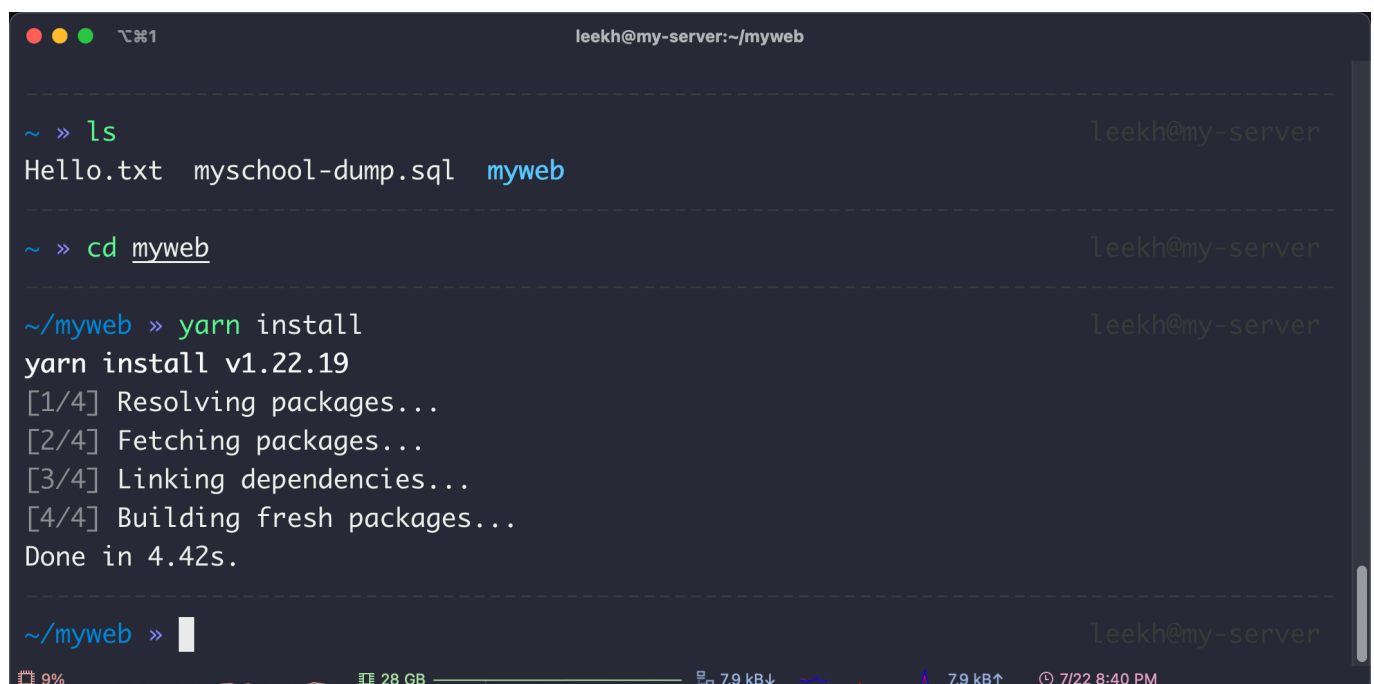
git등을 통해 저장소에서 clone받아도 같은 결과를 얻을 수 있다.



## 2) 패키지 설치하기

프로젝트 폴더 안에서 `package.json`에 명시된 패키지를 일괄 설치한다.

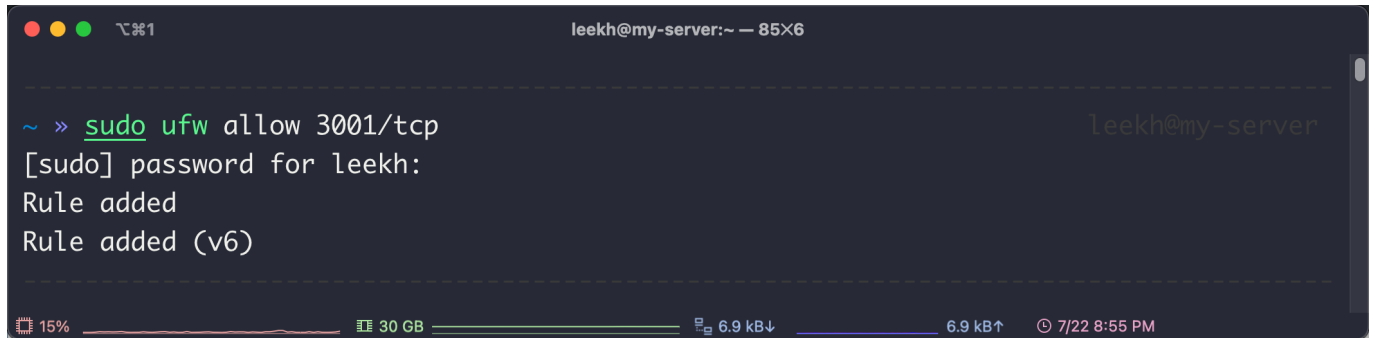
```
$ yarn install
```



### 3) 포트 개방하기

백엔드에서 사용중인 포트번호에 대한 접근을 허용한다.

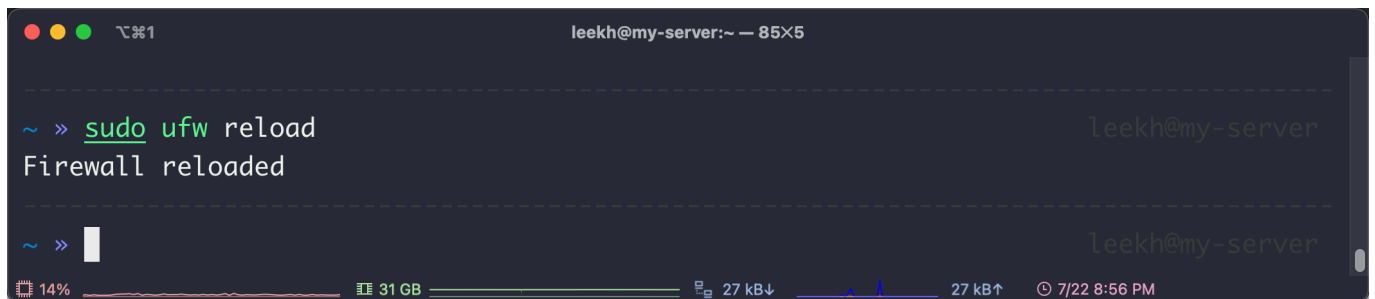
```
$ sudo ufw allow 포트번호/tcp
```



```
leekh@my-server:~ — 85X6  
~ >> sudo ufw allow 3001/tcp  
[sudo] password for leekh:  
Rule added  
Rule added (v6)
```

### 3) 방화벽 새로고침

```
$ sudo ufw reload
```



```
leekh@my-server:~ — 85X5  
~ >> sudo ufw reload  
Firewall reloaded  
~ >> 
```

### 4) 방화벽 상태 확인

```
$ sudo ufw status
```

```

leekh@my-server:~ — 85X17

~ » sudo ufw status
Status: active

To Action From
--
9902/tcp ALLOW Anywhere
10100:10200/tcp ALLOW Anywhere
9901/tcp ALLOW Anywhere
9903/tcp ALLOW Anywhere
3001/tcp ALLOW Anywhere
9902/tcp (v6) ALLOW Anywhere (v6)
10100:10200/tcp (v6) ALLOW Anywhere (v6)
9901/tcp (v6) ALLOW Anywhere (v6)
9903/tcp (v6) ALLOW Anywhere (v6)
3001/tcp (v6) ALLOW Anywhere (v6)

```

## 5) 백엔드 시작하기

시작 파일이 위치한 디렉토리에서 프로그램을 가동한다.

```
$ node app.js
```

```

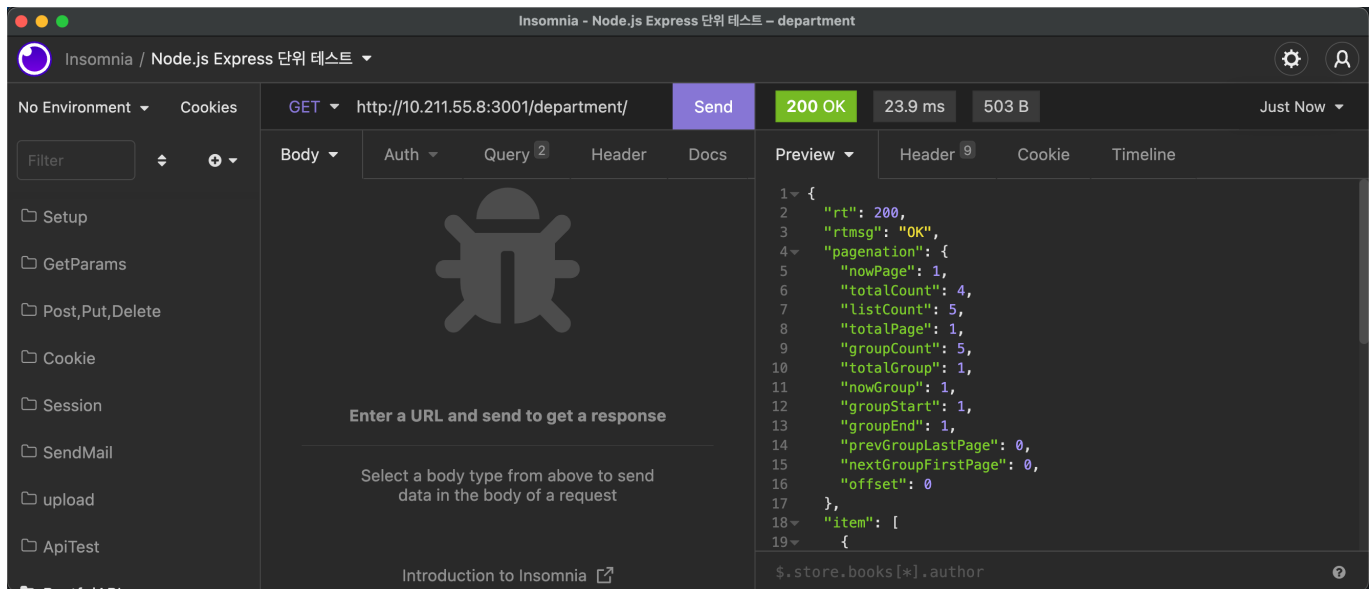
node app.js

~/myweb » node app.js
22/07/22 11:58:08 519 [debug]: -----
22/07/22 11:58:08 520 [debug]: |           Start Express Server           |
22/07/22 11:58:08 520 [debug]: -----
22/07/22 11:58:08 521 [debug]: server address => http://10.211.55.8:3001
22/07/22 11:58:08 521 [debug]: -----

```

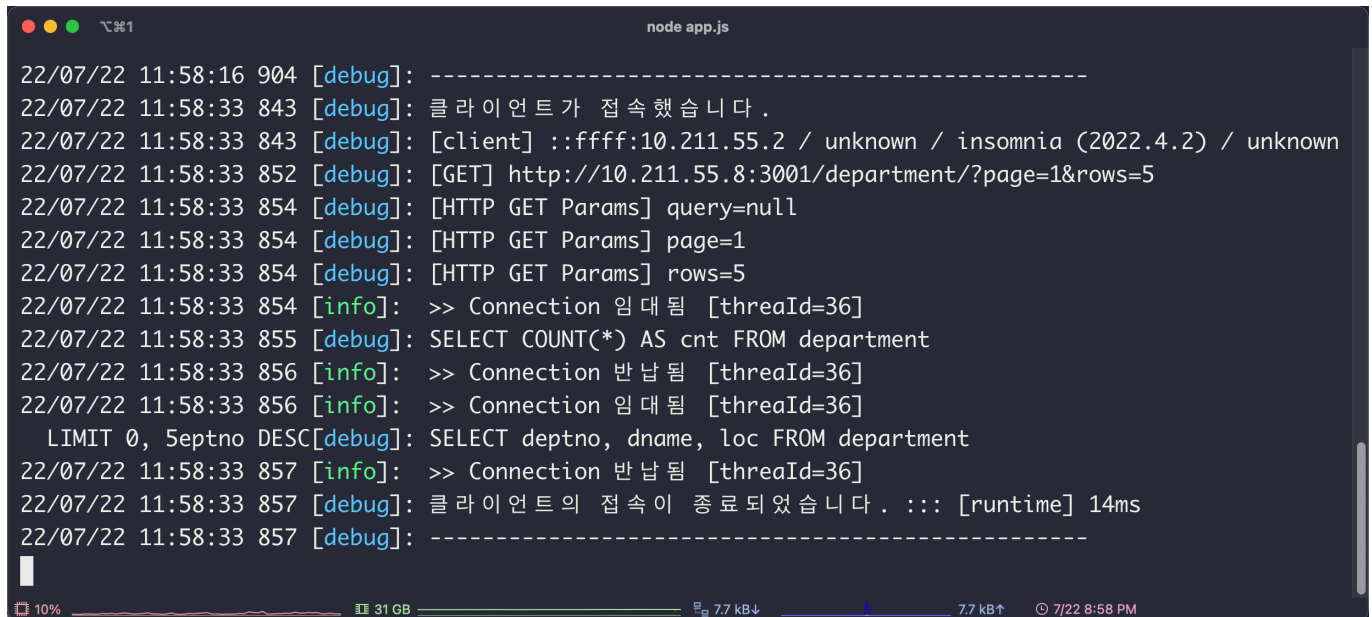
## 6) 백엔드 접속 테스트

Insomnia 등의 클라이언트 프로그램으로 정상 동작을 확인한다.



## 7) 로그 확인

터미널을 통해 출력되는 로그를 확인한다.



종료를 위해서는 **Ctrl+C**를 누른다. 이 상태에서는 리눅스에서 로그아웃하면 백엔드도 종료된다.

## #04. PM2

PM2는 Node.js 어플리케이션을 쉽게 관리할 수 있게 해주는 Process Manager

Node.js 어플리케이션을 cluster mode 로 실행시킨다거나, 메모리가 넘친다거나, 오류로 인해 프로세스가 종료되는 등의 상황에 직면했을 때 간단한 설정만으로도 이러한 처리를 손쉽게 해결할 수 있다.

### 1. PM2 설치하기

```
$ npm install -g pm2
```



```

leekh@my-server:~
~ >> npm install -g pm2
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic.
See https://v8.dev/blog/math-random for details.

added 183 packages, and audited 184 packages in 10s

12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

```

## 2. PM2를 사용한 Express 프로그램 가동하기

```
$ pm2 start 시작스크립트이름
```

```

leekh@my-server:~/myweb
~/myweb >> pm2 start app.js
[PM2] Applying action restartProcessId on app [app](ids: [ 0 ])
[PM2] [app](0) ✓
[PM2] Process successfully started

```

id	name	namespace	version	mode	pid	uptime	⌵	status	cpu	mem	user	watching
0	app	default	1.0.0	fork	22697	0s	0	online	0%	3.4mb	leekh	disabled

## 2. PM2를 통해 관리되는 Express 프로그램 목록 보기

```
$ pm2 list
```

```

leekh@my-server:~/myweb - 133x8
~/myweb >> pm2 list

```

id	name	namespace	version	mode	pid	uptime	⌵	status	cpu	mem	user	watching
0	app	default	1.0.0	fork	22697	36s	0	online	0%	71.3mb	leekh	disabled

## 2. PM2를 통해 관리되는 Express 프로그램중 특정 항목 중지

```
$ pm2 stop id번호
```

```

leekh@my-server:~/myweb — 133x10
~/myweb » pm2 stop 0
[PM2] Applying action stopProcessId on app [0](ids: [ '0' ])
[PM2] [app](0) ✓

```

id	name	namespace	version	mode	pid	uptime	⚡	status	cpu	mem	user	watching
0	app	default	1.0.0	fork	0	0	0	stopped	0%	0b	leekh	disabled

```

~/myweb »

```

16% 26 GB 3.6 kB↓ 3.6 kB↑ 7/25 11:14 AM

## 2. PM2를 통해 관리되는 Express 프로그램중 특정 항목 시작

한번 가동했던 프로그램은 PM2에 등록되기 때문에 현재 작업중인 디렉토리 위치에 상관 없이 id번호를 통해 시작/중지가 가능하다.

```
$ pm2 start id번호
```

```

leekh@my-server:~/myweb — 133x11
~/myweb » pm2 start 0
[PM2] Applying action restartProcessId on app [0](ids: [ '0' ])
[PM2] [app](0) ✓
[PM2] Process successfully started

```

id	name	namespace	version	mode	pid	uptime	⚡	status	cpu	mem	user	watching
0	app	default	1.0.0	fork	22903	0s	0	online	0%	3.4mb	leekh	disabled

```

~/myweb »

```

16% 26 GB 0.5 kB↓ 0.0 kB↑ 7/25 11:16 AM