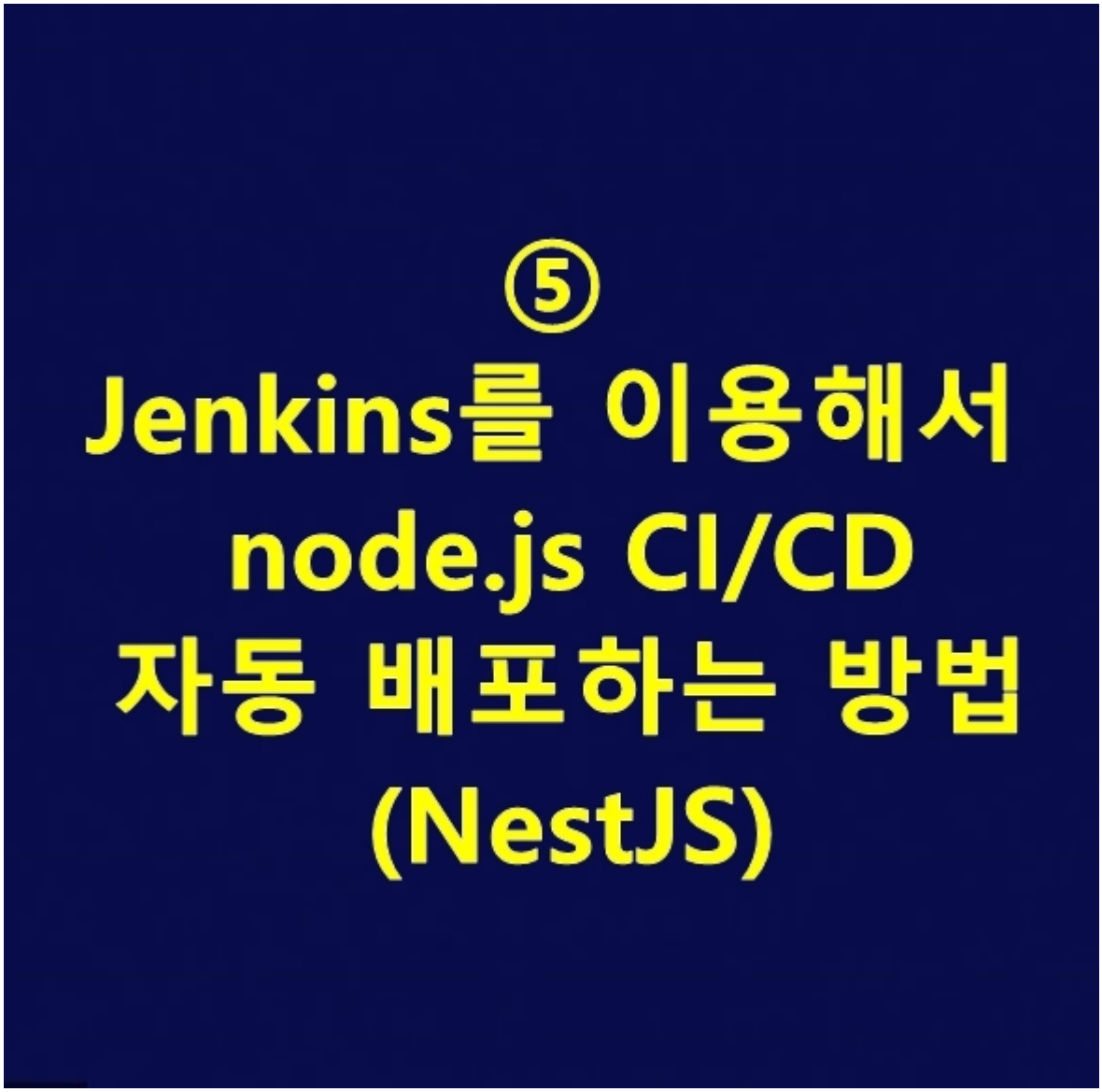
Dev. Back-End/CI & CD

[JenKins] ⑤ 젠킨스를 이용해서 node.js CI/CD 자동 배포 하는 방법 (NestJS)

TROLL's | 2021. 6. 4. 17:17



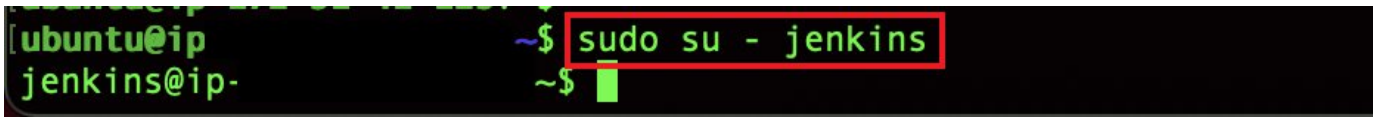
JenKins를 이용해서 node.js 배포하는 방법 (NestJS) 4편에 이어서

5편을 포스팅하겠습니다.

▼ Jenkins를 통해 자동배포 (NodeJS) 시리즈

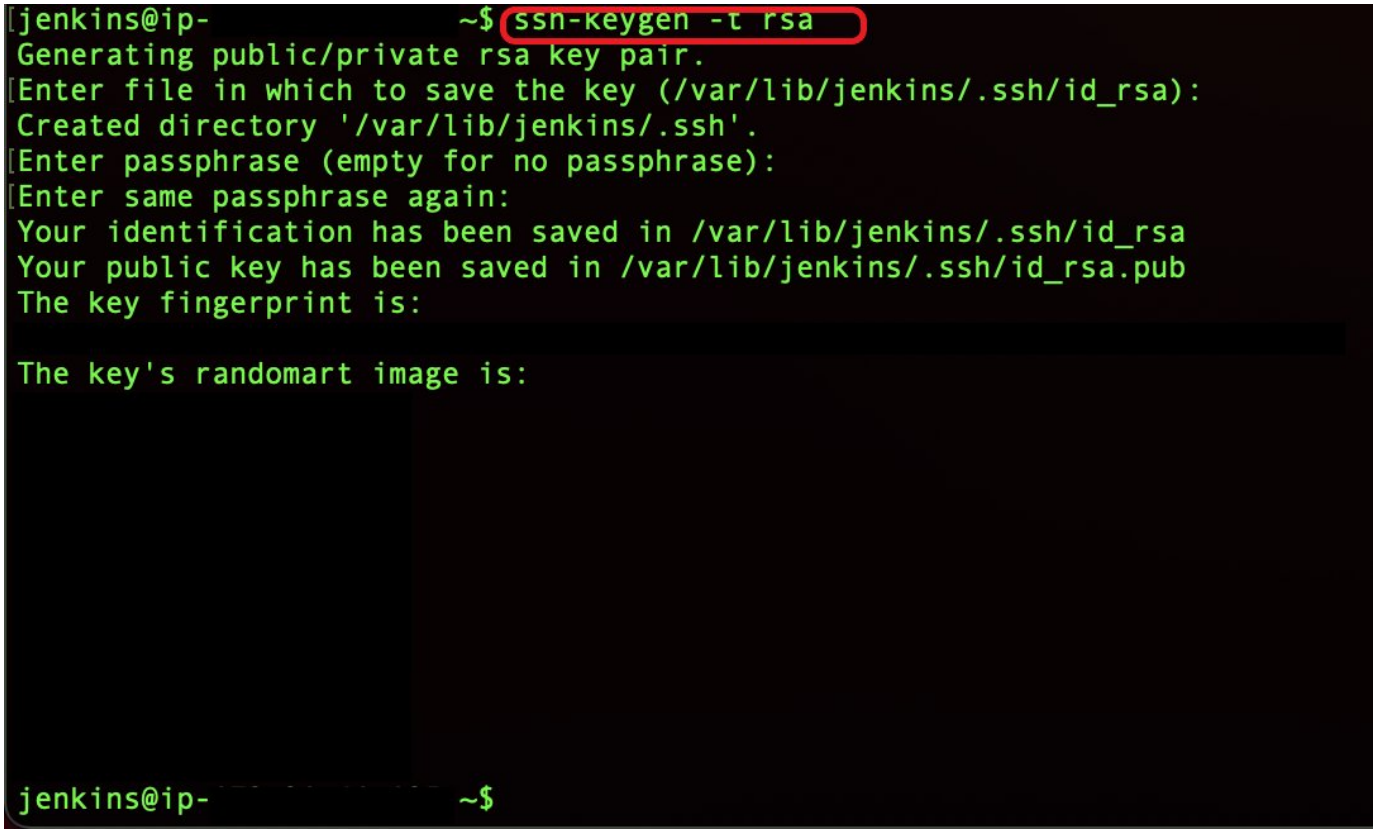
▶ ① 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기
▶ ② 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기
▶ ③ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기
▶ ④ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기
▶ ⑤ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기

● EC2 SSH Authentication



$ sudo su -jenkins

1. Jenkins 서버로 SSH 접속해서 젠킨스 사용자로 유저를 변경합니다.



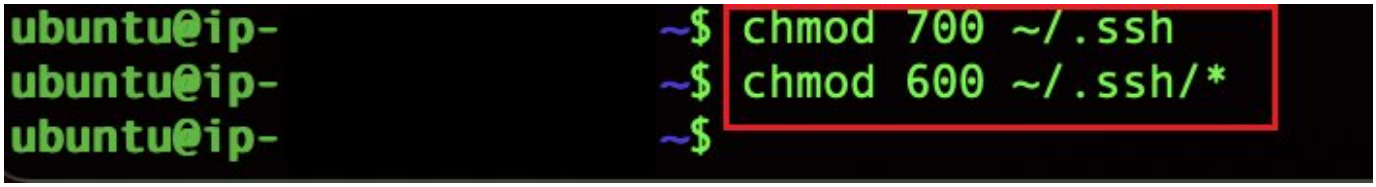
$ ssh-keygen -t rsa

2. SSH키를 발행하고 계속 엔터를 누르고 진행합니다.

$ cat ~/.ssh/id_rsa.pub

3. 파란부분을 복사해줍니다.

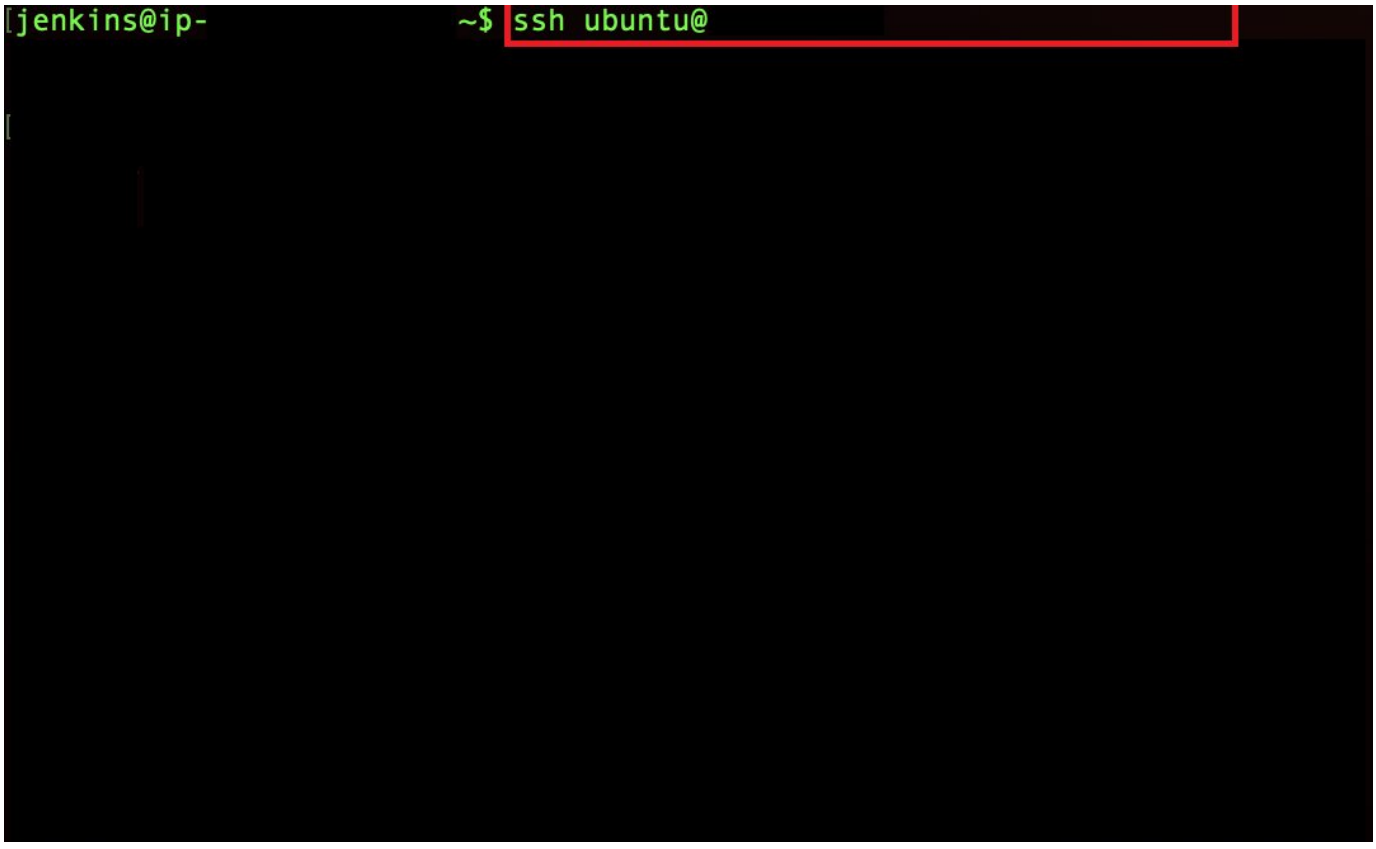
4. vim으로 해당 파일을 들어가서 위에서 복사한 것을 붙여넣기합니다.

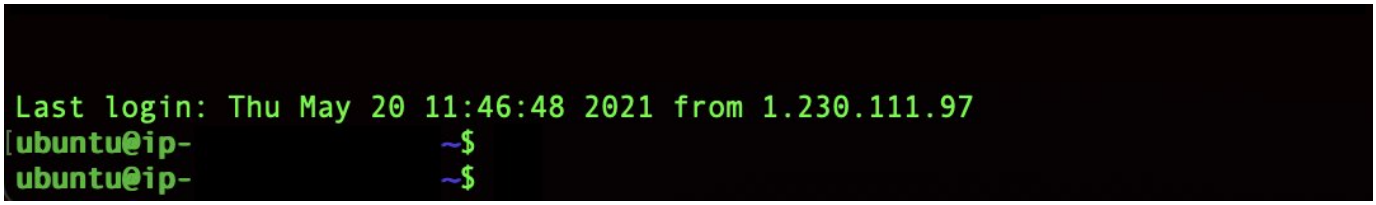


$ chmod 700 ~/.ssh

$ chmod 600 ~/.ssh/*

5. .ssh 펄더 권한을 줍니다.





$ ssh ubuntu@NODE.APP.SERVER.IP

6. 젠킨스 서버로 가서 node서버에 접속이 가능한지 확인합니다.

위와 같이 ssh접속이 되면 성공한것입니다.

● 자동 배포 세팅

```
#!/bin/sh
ssh ubuntu@[redacted] <<EOF
    cd ~/ovni
    git pull origin master
    curl -o-   https://raw.githubusercontent.com/creationix/nvm/v0.33.8/install.sh    | bash
    . ~/.nvm/nvm.sh
    nvm install 16.2.0
    npm install
    npm install -g typescript
    npm install -g ts-node
    npm install -g pm2
    pm2 install typescript
    pm2 restart ecosystem.config.js
    exit
EOF
```

7. node 앱에 배포를 위한 shell script가 필요하기 때문에 script 디렉토리 아래 deploy파일을 생성하여 위처럼 입력합니다. (확장자 없이)

GitHub 주소 : https://github.com/choseongho93/Jenest/blob/master/script/deploy

※ deploy파일 내용 설명

위에서 젠킨스 서버가 Node서버로 SSH접속할수있도록 입력해줍니다.

검정박스는 Node서버 IP주소를 입력하시면 됩니다.

기본 디렉토리가 /home/ubuntu로 되어있고, cd를 통해 프로젝트명으로 이동합니다.
git pull을 받고 npm을 설치해주는 과정을 거치고 ts를 js로 변환할수있도록 설치를 진행합니다.
node프로세스 관리도구인 pm2를 설치하여 run해주도록 ecosystem.config.js파일을 실행합니다.

```
$ chmod +x script/deploy
```

8. 방금 만든 depoy파일 권한을 부여합니다.

```
module.exports = {
    apps: [{
      name    : "onvi",
      script  : "npm",
      args    : "start"
    }]
  }
```

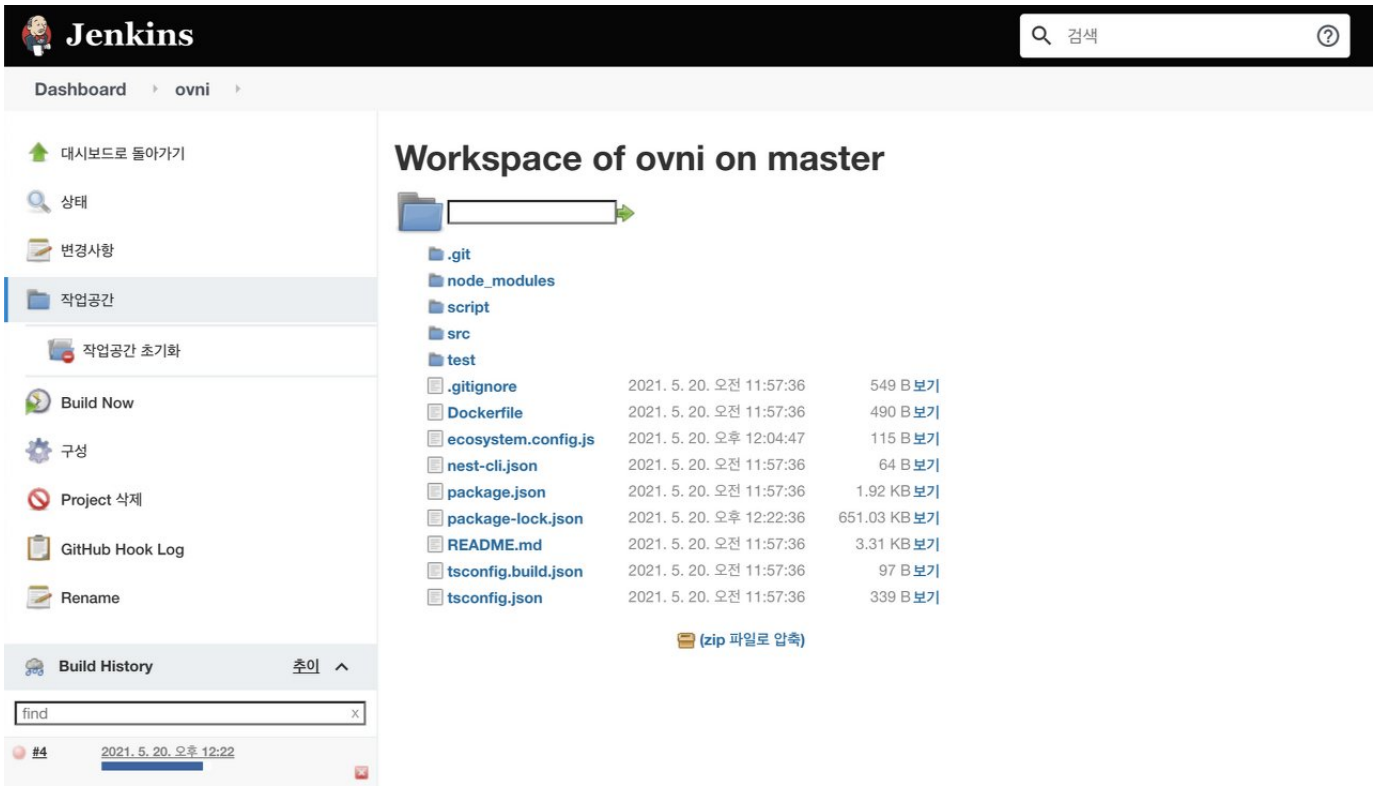
9. 루트 디렉토리 아래 ecosystem.config.js파일을 만들어주고 위와 같이 내용을 넣어줍니다.
GitHub 주소 : https://github.com/choseongho93/Jenest/blob/master/ecosystem.config.js

● 테스트

```ts
import { Injectable } from '@nestjs/common';

@Injectable()
export class AppService {
  getHello(): string {
    return 'Hello World! ovni!';
  }
}
```

10. src/app.service.ts 파일을 위와 같이 텍스트를 수정해줍니다.
GitHub에서 Push합니다.



11. 푸시를 하면 젠킨스에서 좌측 하단에 배포중임을 알수있고, 그에따른 로그도 확인가능합니다.
(초록색이 뜨면 정상 배포되었다는 의미입니다.)

12. Node서버 IP주소:3000으로 접속하면 자동배포가 성공된것을 확인할 수 있습니다.

※ 최종 GitHub 주소 : https://github.com/choseongho93/Jenest

Ref: https://soojae.tistory.com/25
https://12340zszs.tistory.com/33
https://cheese10yun.github.io/PM2/
dlawogus (임재현) - velog

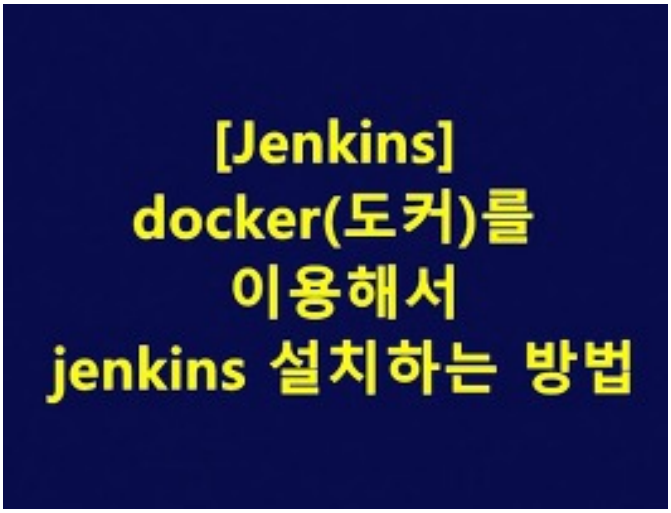
'Dev. Back-End > CI & CD' 카테고리의 다른 글

[Jenkins] job scheduling 설정하는 방법 (batch) (0) 2021.07.24

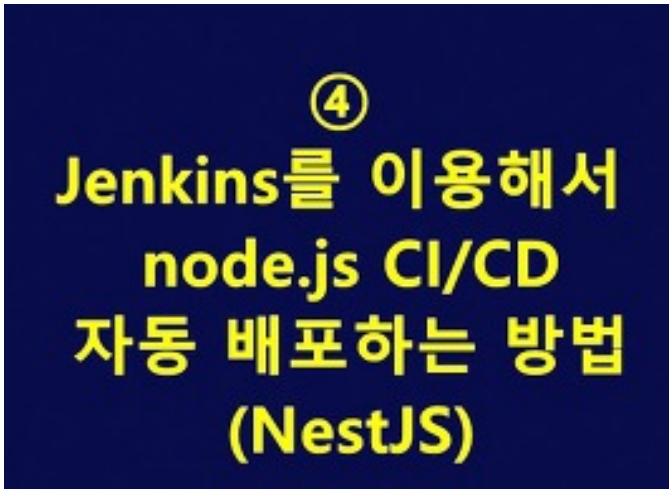
태그 #cd, #CI, #github, #javascript, #NestJS, #node, #Typescript, #빌드, #자동배포, #젠킨스

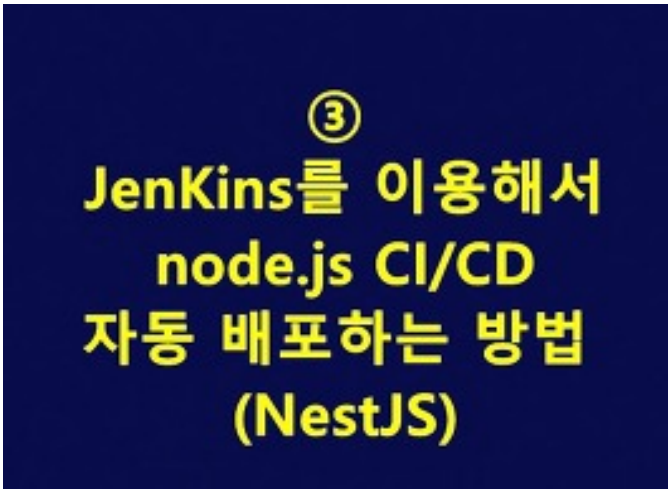
'Dev. Back-End/CI & CD' Related Articles

Jenkins를 Docker Container로 구축시에 TimeZone 설정하는 방법



[Jenkins] docker를 이용해서 jenkins 설치하는 방법 (ubuntu)





[JenKins] ④ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)

[JenKins] ③ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)

이름 암호 Secret

여러분의 소중한 댓글을 입력해주세요.

댓글달기