

Dev, Back-End/CI & CD

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

TROLL's | 2021. 6. 3. 19:03

④
Jenkins를 이용해서
node.js CI/CD
자동 배포하는 방법
(NestJS)

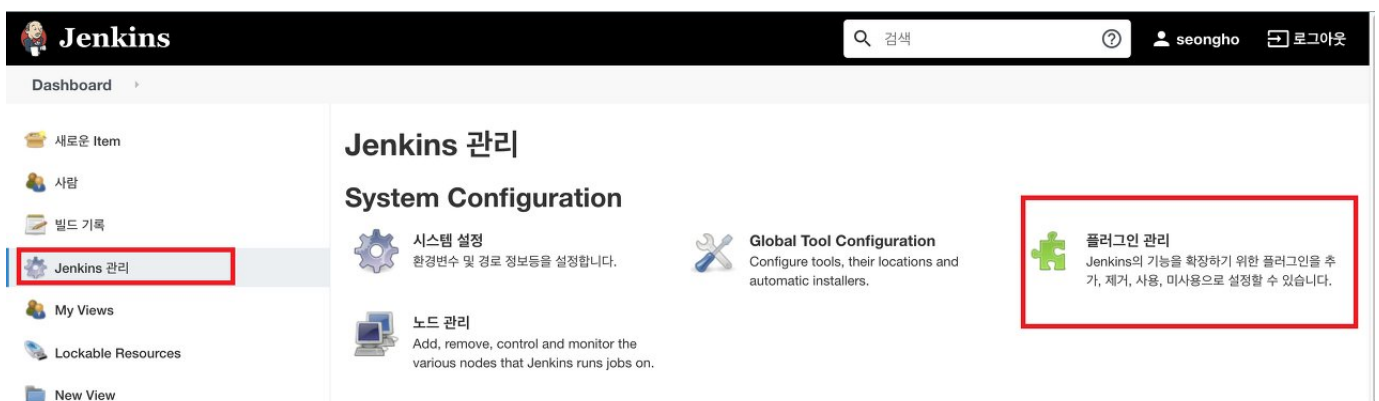
JenKins를 이용해서 node.js 배포하는 방법 (NestJS) 3편에 이어서

4편을 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](#)

● Install NodeJS Plugin on Jenkins



빌드 대기 목록

빌드 대기 항목이 없습니다.

빌드 실행 상태

1 대기 중

2 대기 중

Security

Configure Global Security
 Secure Jenkins; define who is allowed to access/use the system.

Manage Credentials
 Configure credentials

Configure Credential Providers
 Configure the credential providers and types

Manage Users
 Create/delete/modify users that can log in to this Jenkins

Status Information

시스템 정보
 문제 해결을 돕기위한 다양한 환경 정보를 보여줍니다.

System Log
 System log captures output from java.util.logging output related to

부하 통계
 Check your resource utilization and see if you need more computers for your builds.

- 브라우저를 통해 젠킨스 서버에 접속합니다.
Jenkins관리 -> 플러그인 관리를 클릭합니다.

Jenkins

검색

seongho

로그아웃

Dashboard

Plugin Manager

대시보드로 돌아가기

Jenkins 관리

NodeJS

업데이트된 플러그인 목록

설치 가능

설치된 플러그인 목록

고급

Install	Name	Version	Released
<input type="checkbox"/>	NodeJS npm NodeJS Plugin executes NodeJS script as a build step.	1.4.0	2 mo 5 days ago

Install without restart

Download now and install after restart

Update information obtained: 57 min ago

지금 확인

- 설치가능 탭에서 NodeJS를 검색한뒤에 체크박스를 체크하고 "지금 다운로드하고 재시작 후 설치하기"를 클릭합니다.

Jenkins

검색

seongho

로그아웃

Dashboard

Update Center

대시보드로 이동

Jenkins 관리

플러그인 관리

플러그인 설치/업그레이드 중

준비

- Checking internet connectivity
- Checking update center connectivity
- Success

Config File Provider

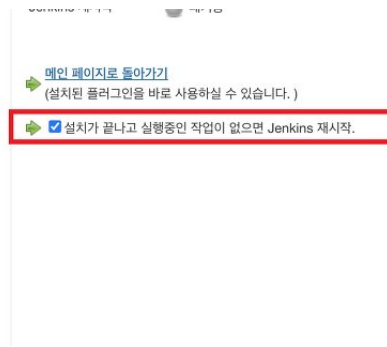
설치중

NodeJS

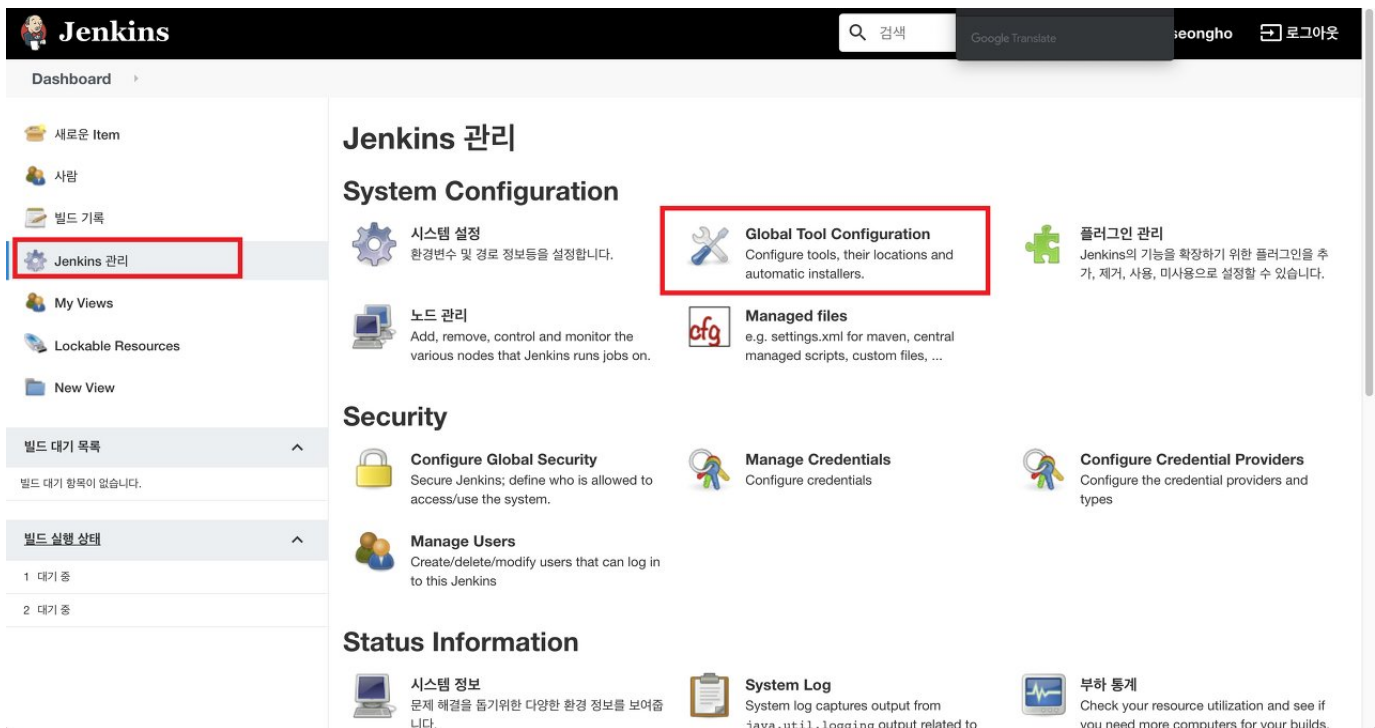
대기중

.jenkins 재시작

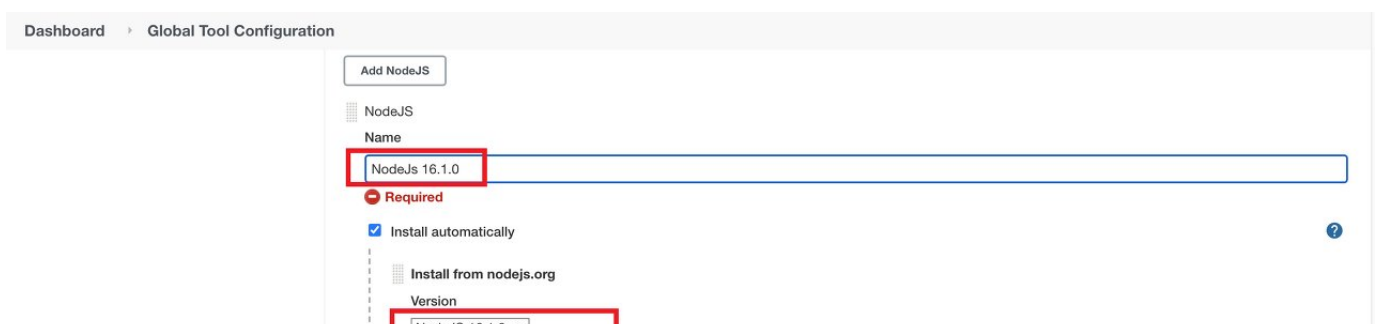
대기중



3. 플러그인 설치 화면으로 이동하면 "설치가 끝나고 실행중인 작업이 없으면 젠킨스 재시작"을 체크해줍니다.



4. 설치가 완료되면 재실행하여 Jenkins관리 -> Global Tool Configuration을 클릭합니다.



☐ Force 32bit architecture
For the underlying architecture, if available, force the installation of the 32bit package. Otherwise the build will fail

Global npm packages to install

Specify list of packages to install globally -- see npm install -g. Note that you can fix the packages version by using the syntax 'packageName@version'

Global npm packages refresh hours

Duration, in hours, before 2 npm cache update. Note that 0 will always update npm cache

Add Installer ▼

Delete Installer

Save Apply

5. Node JS탭에서 project에 쓸 node version을 선택합니다.
최신 버전으로 선택하고 name을 똑같이 입력해주고 저장합니다.

● Create a new Jenkins Job

Jenkins

검색

Google Translate

seongho

로그아웃

Dashboard

새로운 Item

사람

빌드 기록

Jenkins 관리

My Views

Lockable Resources

New View

빌드 대기 목록

빌드 실행 상태

Jenkins 관리

System Configuration

Security

Status Information

시스템 설정

환경변수 및 경로 정보등을 설정합니다.

노드 관리

Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Global Tool Configuration

Configure tools, their locations and automatic installers.

Managed files

e.g. settings.xml for maven, central managed scripts, custom files, ...

플러그인 관리

Jenkins의 기능을 확장하기 위한 플러그인을 추가, 제거, 사용, 미사용으로 설정할 수 있습니다.

Configure Global Security

Secure Jenkins; define who is allowed to access/use the system.

Manage Credentials

Configure credentials

Configure Credential Providers

Configure the credential providers and types

Manage Users

Create/delete/modify users that can log in to this Jenkins

시스템 정보

문제 해결을 돕기위한 다양한 환경 정보를 보여줍니다.

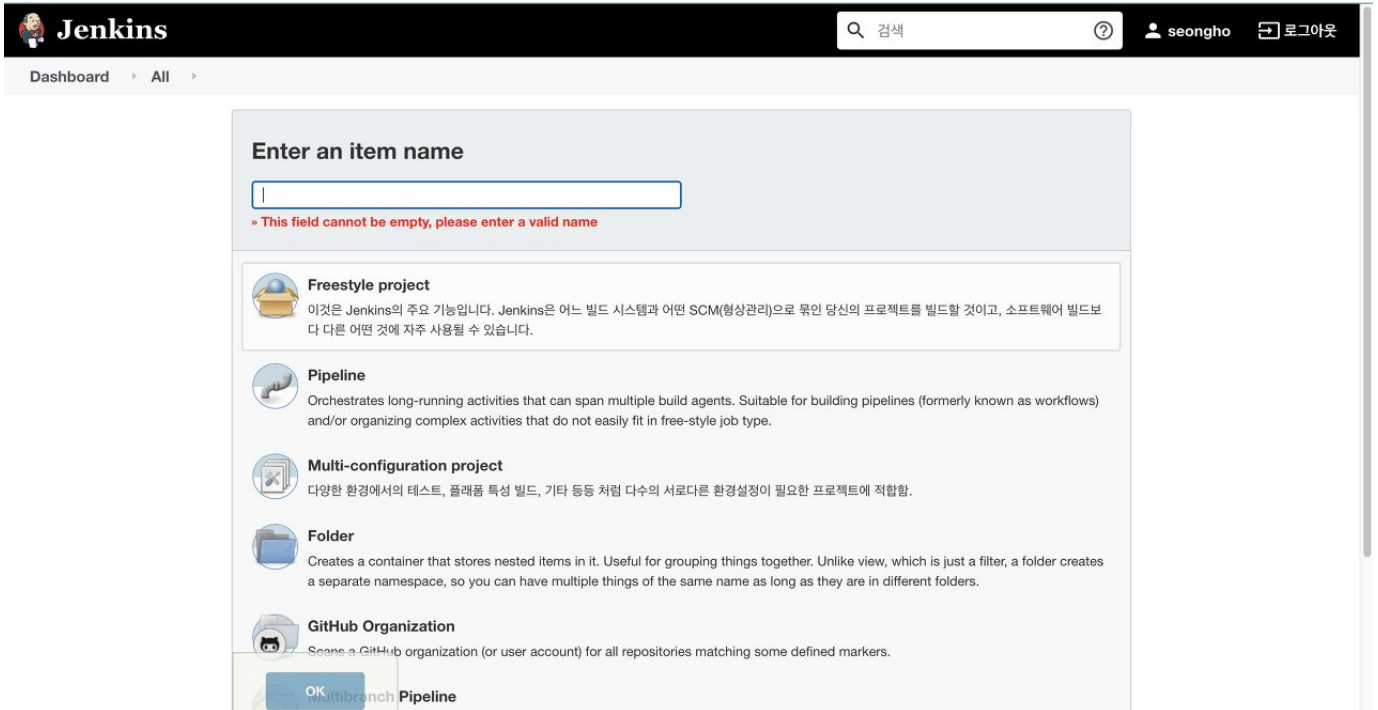
System Log

System log captures output from java.util.logging output related to

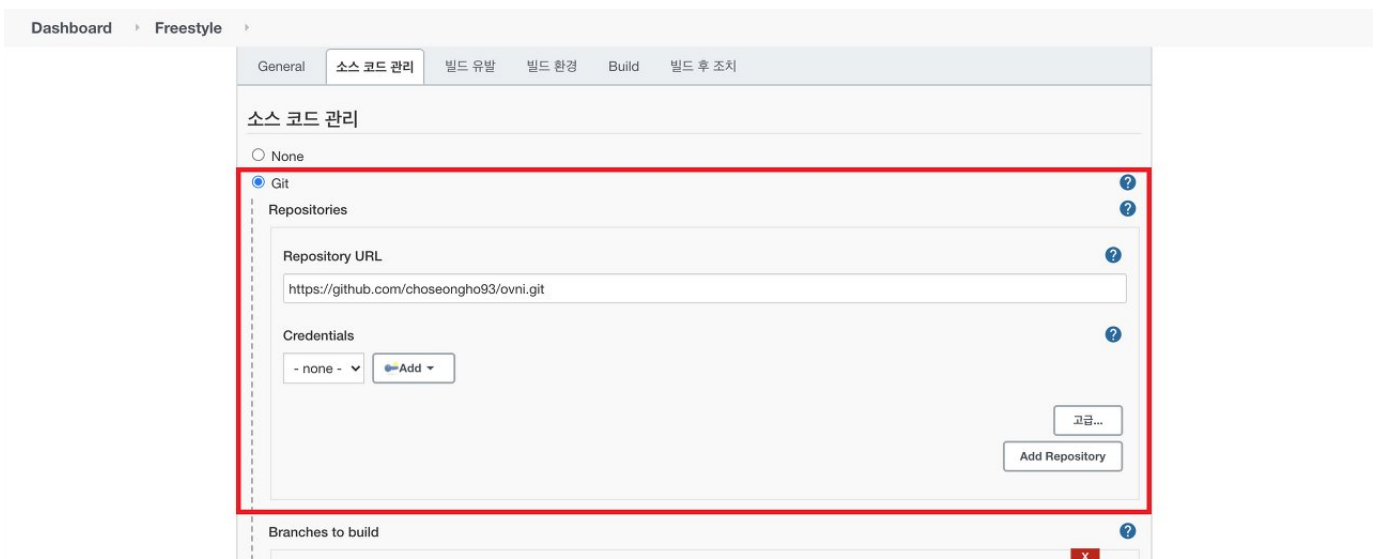
부하 통계

Check your resource utilization and see if you need more computers for your builds.

6. 새로운 Item을 클릭합니다.



7. 검색창에 깃헙에서 만든 프로젝트 이름을 입력해주고 Freestyle project 선택 해줍니다.

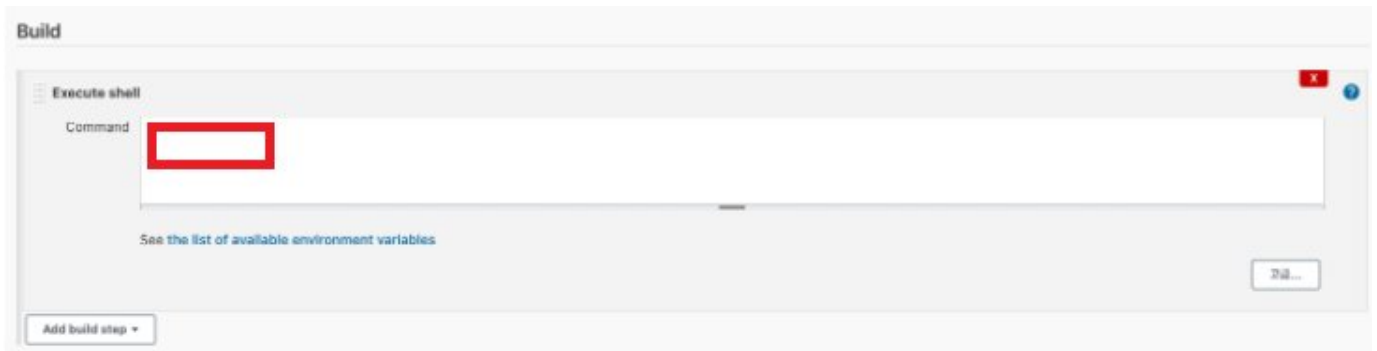


8. 소스 코드 관리 탭에서 GitHub 주소와 브랜치를 입력해줍니다.

만일, 소스코드 관리에 None만 있다면 위 NodeJS 플러그인 관리에서 설치했던것처럼 GitHub 플러그인을 별도로 설치를 동일하게 해줍니다.

9. GitHub hook trigger for GITScm polling에 체크해줍니다.

빌드환경에서 Provide Node & npm bin/ folder to PATH 체크한뒤에 Global Tool Configuration에서 만들었던 node 버전 설정을 선택합니다.

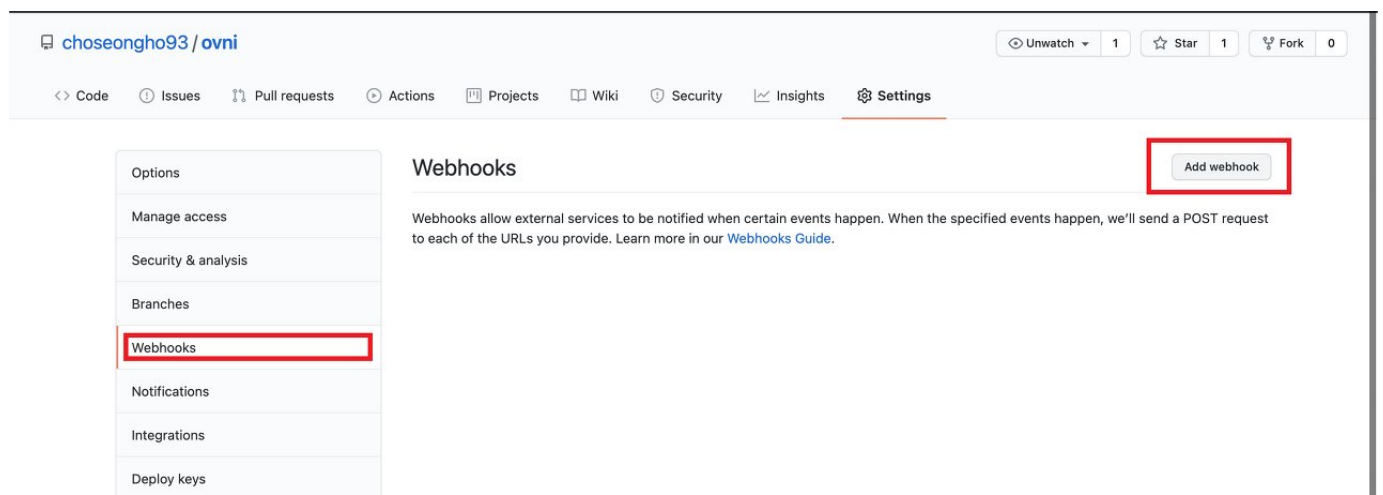


10. Build -> Execute shell 선택합니다.

```
npm install  
./script/deploy
```

위 처럼 입력하고 저장합니다.

● Add Git Webhook



Actions
Environments
Secrets

11. 깃헙 홈페이지로 이동해서 우측 상단에 프로필 클릭하고 setting을 클릭해줍니다.
그리고 webhooks 탭을 클릭하고 Add webhook 버튼을 클릭해줍니다.

Options

- Manage access
- Security & analysis
- Branches
- Webhooks**
- Notifications
- Integrations
- Deploy keys
- Actions
- Environments
- Secrets
- Pages
- Moderation settings

Webhooks / Add webhook

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

Payload URL *

Content type

application/x-www-form-urlencoded

Secret

Which events would you like to trigger this webhook?

- ☒ Just the push event.
- ☐ Send me everything.
- ☐ Let me select individual events.

☒ **Active**
We will deliver event details when this hook is triggered.

Add webhook

12. Payload URL에 `http://JENKINS.SERVER.IP:8080/github-webhook/` 을 입력하고 위와 같이 맞춰주고 추가해줍니다.

젠킨스로 CI/CD하는 방법에 대해 다음에 이어서 진행하겠습니다.

Ref: <https://soojae.tistory.com/25>
<https://12340zszs.tistory.com/33>
<https://cheese10yun.github.io/PM2/>

블로그 내용이 도움이 되었다면 아래 광고를 **클릭** 해주시면 감사하겠습니다. ^^

♡ 2  

구독하기



'[Dev. Back-End](#) > [CI & CD](#)' 카테고리의 다른 글

[Jenkins] job scheduling 설정하는 방법 (batch) (0)	2021.07.24
[Jenkins] Global 환경변수 등록하는 방법 (0)	2021.07.23
Jenkins를 Docker Container로 구축시에 TimeZone 설정하는 방법 (0)	2021.07.21
[Jenkins] docker를 이용해서 jenkins 설치하는 방법 (ubuntu) (0)	2021.07.11
[JenKins] ⑤ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)	2021.06.04
[JenKins] ④ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)	2021.06.03
[JenKins] ③ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)	2021.06.02
[JenKins] ② 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)	2021.06.01
[JenKins] ① 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)	2021.05.29

태그 #ci/cd, #Git 연동, #github, #jenkins, #jenkins 플러그인, #nodejs, #webhook, #자동배포,
#젠킨스, #플러그인

'[Dev. Back-End/CI & CD](#)' Related Articles

[Jenkins]

⑤

docker(도커)를 이용해서 jenkins 설치하는 방법

[Jenkins] docker를 이용해서 jenkins 설치하는 방법 (ubuntu)

Jenkins를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)

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

③ JenKins를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)

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

② JenKins를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)

[JenKins] ② 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)

☐ Secret

여러분의 소중한 댓글을 입력해주세요.

댓글달기

