

# JENKINS Native 배포하기

2023.02.03 부울경 1반 실습코치 김효은



# 목차

- ✓ JENKINS 설치
- ✓ JENKINS 초기 세팅
- ✓ Plugin 설치
- ✓ Webhook 연결
- ✓ Item 생성

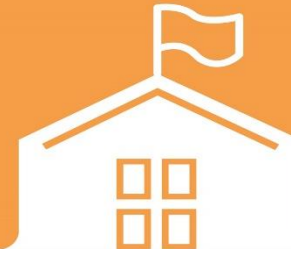


# 01

## EC2에 JENKINS 설치하기

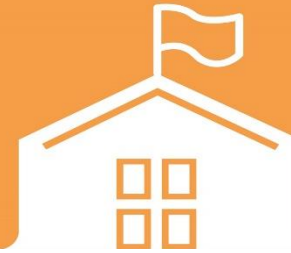


# Java 11 설치하기



- ✓ Jenkins는 JAVA 기반으로 개발된 소프트웨어
- ✓ 따라서 Java 설치가 필요함
- ✓ Jenkins LTS ver. 은 JAVA11 또는 JAVA17에서 동작하므로 해당 버전을 설치

# Java 11 설치하기



```
$ sudo apt update
```

```
$ sudo apt upgrade
```

```
$ sudo apt install openjdk-11-jre
```

```
$ java --version
```

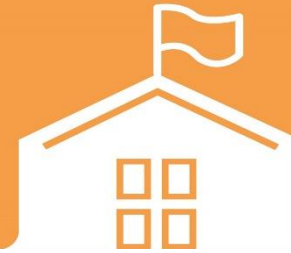
```
# apt package 업데이트
```

```
# apt package 업데이트 적용
```

```
# JAVA 11 다운로드
```

```
# java version 체크
```

# Jenkins 설치



- ✓ `$ curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo tee W /usr/share/keyrings/jenkins-keyring.asc > /dev/null`
- ✓ `$ echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]  
W https://pkg.jenkins.io/debian-stable binary/ | sudo tee  
W /etc/apt/sources.list.d/jenkins.list > /dev/null`
- ✓ `$ sudo apt-get update`
- ✓ `$ sudo apt-get install jenkins`

# Jenkins 설치

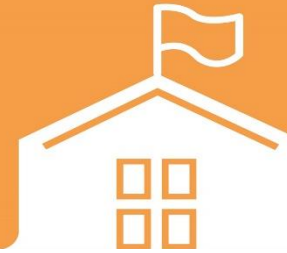


- ✓ `$ curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo tee W /usr/share/keyrings/jenkins-keyring.asc > /dev/null`
- ✓ `$ echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]  
W https://pkg.jenkins.io/debian-stable binary:amd64 | sudo tee  
W /etc/apt/sources.list.d/jenkins.list > /dev/null`
- ✓ `$ sudo apt-get update`
- ✓ `$ sudo apt-get install jenkins`

너무 길쭉?

우리 공식문서 한번 볼까요?

# Jenkins 설치



함께가요 미래로!  
Enabling People

## User Handbook

- [User Handbook Overview](#)
- **Installing Jenkins**
  - [Docker](#)
  - [Kubernetes](#)
  - [Linux](#)

## Long Term Support release

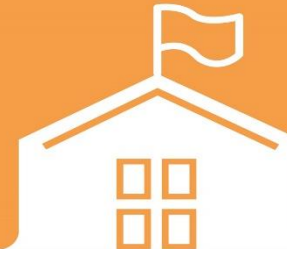
A [LTS \(Long-Term Support\) release](#) is chosen every 12 weeks from the stream of regular releases as the stable release for that time period. It can be installed from the [debian-stable apt repository](#).

```
curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo tee \  
  /usr/share/keyrings/jenkins-keyring.asc > /dev/null  
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \  
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \  
  /etc/apt/sources.list.d/jenkins.list > /dev/null  
sudo apt-get update  
sudo apt-get install jenkins
```

✓ 공식문서를 보는 습관을 들입시다 ☺



# Jenkins 서비스 상태 확인



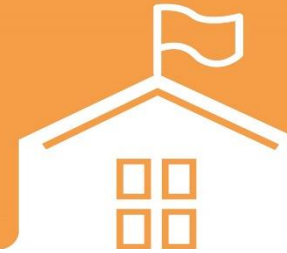
함께가요 미래로!  
Enabling People

```
ubuntu@ip-172-26-9-96:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-01-18 06:31:47 UTC; 22h ago
     Main PID: 177739 (java)
       Tasks: 45 (limit: 4710)
      Memory: 1.2G
     CGroup: /system.slice/jenkins.service
             └─177739 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

Jan 18 06:31:27 ip-172-26-9-96 jenkins[177739]: eed7401f6430481496dae1c4c3739e38
Jan 18 06:31:27 ip-172-26-9-96 jenkins[177739]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Jan 18 06:31:27 ip-172-26-9-96 jenkins[177739]: *****
Jan 18 06:31:27 ip-172-26-9-96 jenkins[177739]: *****
Jan 18 06:31:27 ip-172-26-9-96 jenkins[177739]: *****
Jan 18 06:31:47 ip-172-26-9-96 jenkins[177739]: 2023-01-18 06:31:47.598+0000 [id=29] INFO jenkins.InitReactorRunner$1#onAttained: Completed initialization
Jan 18 06:31:47 ip-172-26-9-96 jenkins[177739]: 2023-01-18 06:31:47.634+0000 [id=22] INFO hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up and running
Jan 18 06:31:47 ip-172-26-9-96 systemd[1]: Started Jenkins Continuous Integration Server.
Jan 18 06:31:50 ip-172-26-9-96 jenkins[177739]: 2023-01-18 06:31:50.293+0000 [id=46] INFO h.m.DownloadService$Downloadable#load: Obtained the updated war
Jan 18 06:31:50 ip-172-26-9-96 jenkins[177739]: 2023-01-18 06:31:50.294+0000 [id=46] INFO hudson.util.Retrier#start: Performed the action check update
lines 1-19/19 (END)
```

✓ \$ sudo systemctl status Jenkins

# Jenkins 포트 변경



```
$ jenkins --version
```

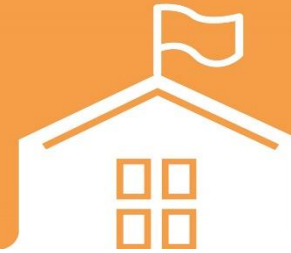
```
$ sudo vi /lib/systemd/system/jenkins.service
```

# Jenkins version 2.335 이후

```
$ sudo vi /etc/default/jenkins
```

# 2.335 이전

# Jenkins 포트 변경



```
# IP address to listen on for HTTP requests.  
# The default is to listen on all interfaces (0.0.0.0).  
#Environment="JENKINS_LISTEN_ADDRESS="
```

```
# Port to listen on for HTTP requests. Set to -1 to disable.  
# To be able to listen on privileged ports (port numbers less than 1024),  
# add the CAP_NET_BIND_SERVICE capability to the AmbientCapabilities  
# directive below.  
Environment="JENKINS_PORT=9090"
```

- ✓ Environment에서 PORT 수정 후
- ✓ \$ sudo service jenkins restart

# 02

## JENKINS 초기 세팅



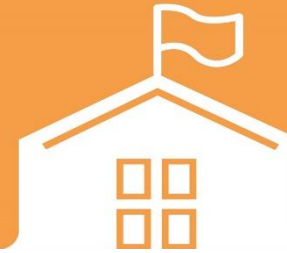


# Jenkins 초기 비밀번호 확인

```
ubuntu@ip-172-26-9-96:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword  
[REDACTED]
```

- ✓ `$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword`
- ✓ 초기 비밀번호는 세팅할 때 필요하므로 적어 두기

# Jenkins 웹페이지 접속



함께가요 미래로!  
Enabling People

- ✓ 서버주소: 포트번호

ex. <http://i8e123.p.ssafy.io:9090>

- ✓ 앞에서 본 초기 비밀번호 입력

```
$ sudo cat
```

Getting Started

## Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

```
/var/lib/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

# Jenkins 웹페이지 접속



함께가요 미래로!  
Enabling People

- ✓ 서버주소: 포트번호

ex. <http://i8e123.p.ssafy.io:9090>

- ✓ 앞에서 본 초기 비밀번호 입력

Getting Started

## Unlock Jenkins

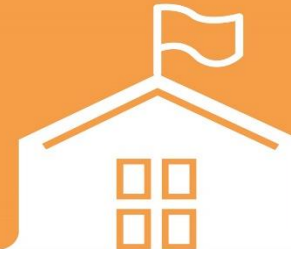
To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

```
/var/lib/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

# Jenkins 웹페이지 접속



함께가요 미래로!  
Enabling People

## ✓ Suggested plugins 설치

Getting Started

## Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

### Install suggested plugins

Install plugins the Jenkins community finds most useful.

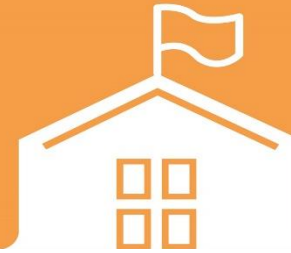
### Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.277.4



# Jenkins 웹페이지 접속



함께가요 미래로!  
Enabling People

## Create First Admin User

계정명:

암호:

암호 확인:

이름:

이메일 주소:

✓ 계정명 & 암호로  
Jenkins에 접속하므로  
잊어버리면 안됨!!



# 환경변수 추가1. JAVA

✓ `$ sudo vim ~/.bashrc`

case: Java 17 에서 진행하는  
스프링 프로젝트를 진행할 경우

✓ # 아래 내용 추가

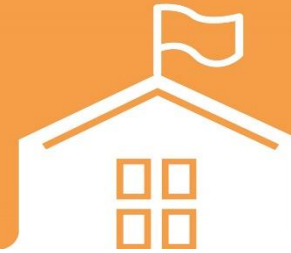
✓ `export JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64`

✓ `export PATH="$PATH:$JAVA_HOME/bin"`

✓ `$ source ~/.bashrc`

# 변경사항 적용

# 환경변수 추가1. JAVA



✓ `$ which java`

# java 실제 주소

ex. `/usr/bin/java`

✓ `$ readlink -f /usr/bin/java`

# 실제 디렉토리

ex. `/usr/lib/jvm/java-11-  
openjdk-amd64/bin/java`

✓ `$ echo $JAVA_HOME`

# JAVA 환경변수 체크



## 환경변수 추가2. GIT

✓ `$ sudo apt-get install git`

또는

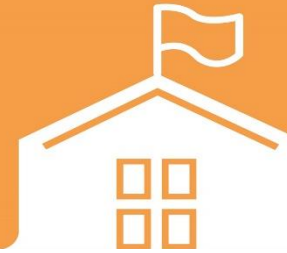
✓ `$ sudo apt install git`

# git 설치

✓ `$ git --version`

# git 설치(버전) 체크

# 환경변수 추가2. Jenkins-GIT



함께가요 미래로!  
Enabling People

Jenkins 관리 >

Global Tool Configuration

Git

Git installations

≡ Git

Name

git

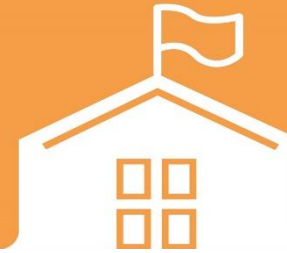
Path to Git executable ?

/usr/bin/git

☐ Install automatically ?

Add Git ▾

# 그외 환경변수 추가



✓ JDK, Gradle, npm.. 등 배포에 필요한 패키지들

1. EC2에 설치하고

2. EC2 또는 JENKINS Global Tool Configuration 환경변수 등록하기  
(npm..)                      (Gradle, Maven, Ant 등)

# 03

## Plugin 설치하기



# 플러그인 추가 방법



함께가요 미래로!  
Enabling People

Jenkins

Dashboard > Jenkins 관리

+ 새로운 Item

사람

빌드 기록

**Jenkins 관리**

My Views

빌드 대기 목록

빌드 대기 항목이 없습니다.

빌드 실행 상태

1 대기 중

2 대기 중

**Jenkins 관리**

Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#).

Set up agent Set up cloud Dismiss

**System Configuration**

시스템 설정  
환경변수 및 경로 정보등을 설정합니다.

Global Tool Configuration  
Configure tools, their locations and automatic installers.

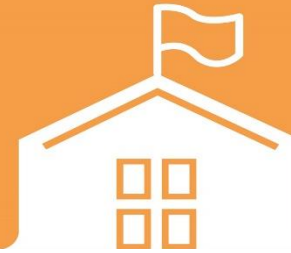
**플러그인 관리**  
Jenkins의 기능을 확장하기 위한 플러그인을 추가, 제거, 사용, 미사용으로 설정할 수 있습니다.

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

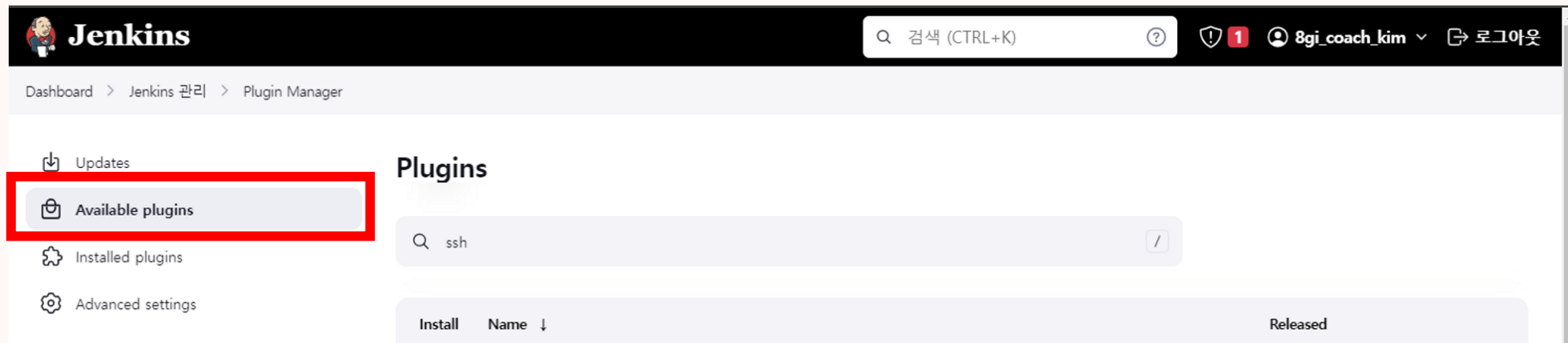
✓ Jenkins 관리 > 플러그인 관리



# 플러그인 추가 방법

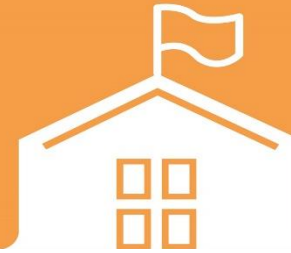


함께가요 미래로!  
Enabling People



- ✓ Available plugins
- ✓ 상단에서 키워드 검색해서 찾기

# 필요 플러그인



## Publish Over SSH 1.24

Send build artifacts over SSH

[Report an issue with this plugin](#)

✓ 서버에 빌드 파일을 옮겨 주는 플러그인

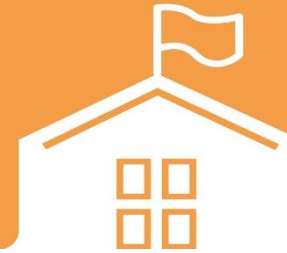
## Post build task 1.9

Allows to execute a batch/shell task depending on the build log output.

[Report an issue with this plugin](#)

✓ 빌드 후 조치를 세팅하기 위한 플러그인

# 필요 플러그인



## Generic Webhook Trigger Plugin 1.86.2

Can receive any HTTP request, extract any value from Bitbucket, Jira and many more.

[Report an issue with this plugin](#)

✓ Gitlab webhook을 활용하기 위한 plugin

## GitLab Plugin 1.6.0

This plugin allows **GitLab** to trigger Jenkins builds and display their results in the GitLab UI.

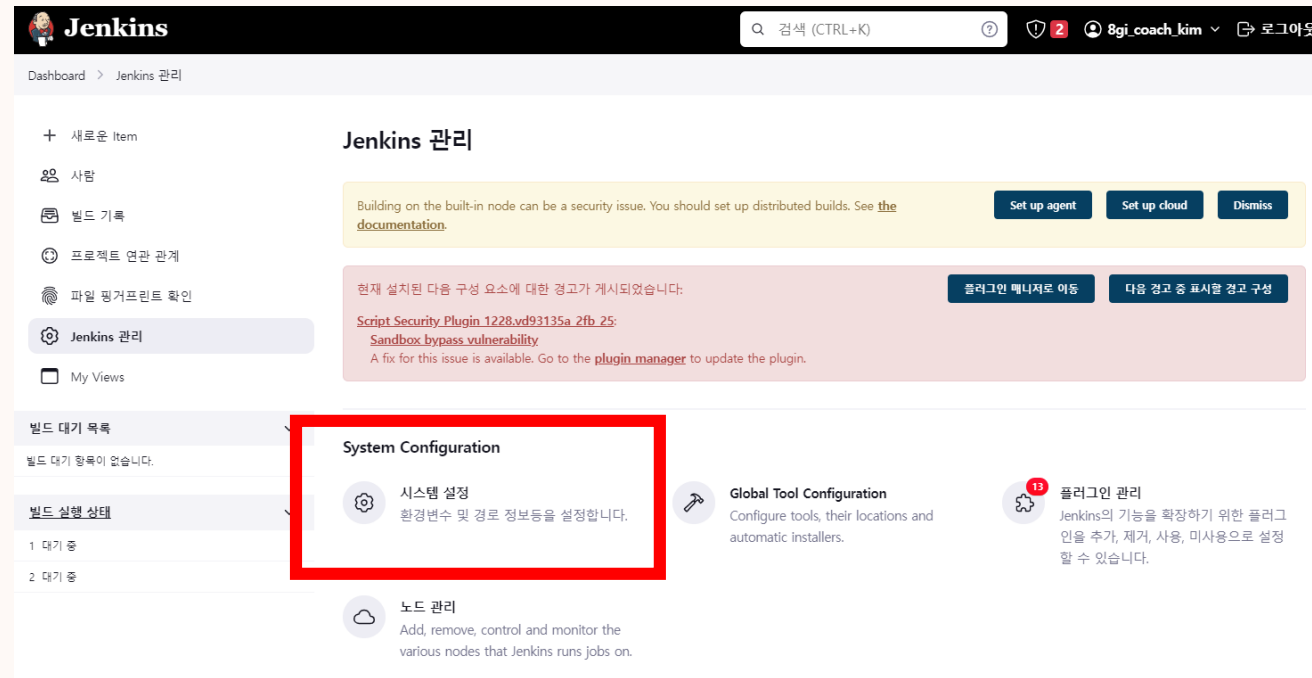
[Report an issue with this plugin](#)

This plugin is up for adoption! We are looking for new maintainers. Visit our [Adopt a Plugin](#)

# Publish over SSH 세팅



함께가요 미래로!  
Enabling People



✓ dashboard > jenkins 관리 > 시스템 설정 > Publish over SSH > 하단에 추가

# Publish over SSH 세팅



함께가요 미래로!  
Enabling People

SSH Servers

SSH Server

Name ?

coach21

본인이 식별하기 위한 Name

Hostname ?

.p.ssafy.io

EC2 서버 주소

Username ?

ubuntu

서버 계정명(SSAFY 서버의 경우 ubuntu로 모두 동일)

Remote Directory ?

/home/jenkins/workspace/

빌드된 파일을 위치시킬 디렉토리(반드시 **실재** 경로)

The base directory for this configuration.

This directory must exist, it will not be created.

The publisher will only be able to place files below this directory.

Note that nothing prevents an exec from creating, deleting or moving files anywhere that the user has permission to.

(from [Publish Over SSH](#))



# Publish over SSH 세팅

☒ Use password authentication, or use a different key ?

고급 버튼을 눌러 Use password authentication 체크

Passphrase / Password ?

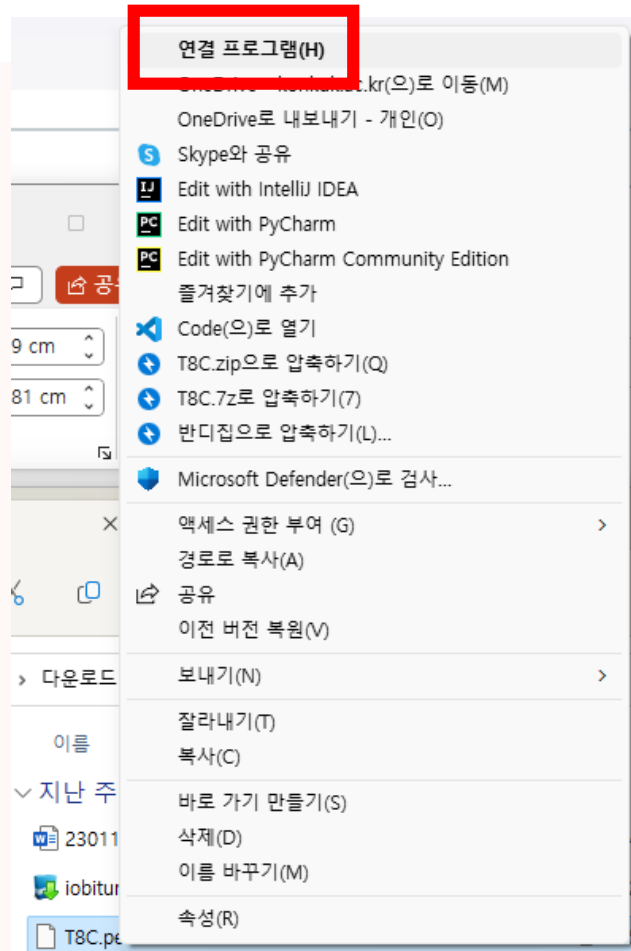
Path to key ?

Key ?

```
-----BEGIN RSA PRIVATE KEY-----  
MIIEpQIBAAKCAQEAOj6QpcDQUNR9q+StnJ4iOqz2LHkU4G4KIAxvH3/88DOXVf6W  
h3TJ2ArFCf1gbzgebeFf8LhnXKe6VDoHvtBk5sbn9SJ1/JB+RZkUbCbHiDHDNU91
```

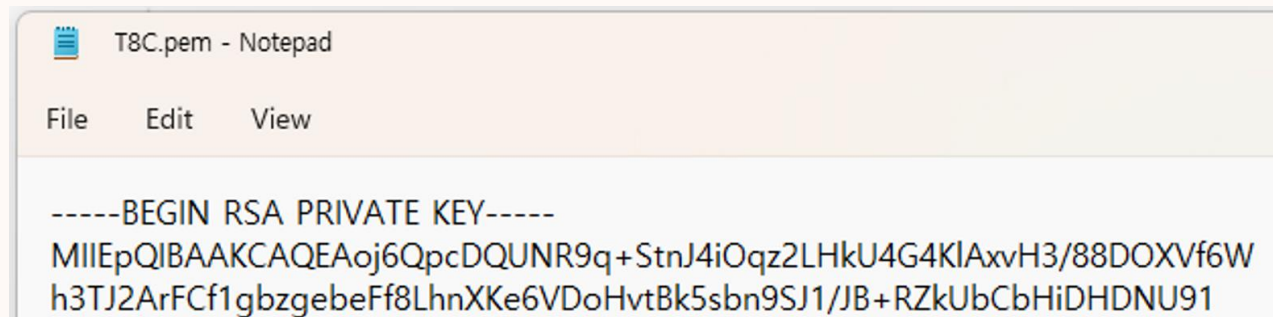
지금받은 pem key 파일 내용 붙여넣기

# Pem Key 내용 확인하기



지금받은 pem 키 > 오른쪽 마우스

> 연결프로그램 Notepad로 열기



# Publish over SSH 세팅



함께가요 미래로!  
Enabling People

SSH Servers

SSH Server

Name ?

coach21

Hostname ?

.p.ssafy.io

Username ?

ubuntu

Remote Directory ?

/home/jenkins/workspace/



고급...

Success

Test Configuration

추가

Test Configuration 클릭

> Success가 뜨면 세팅 끝!

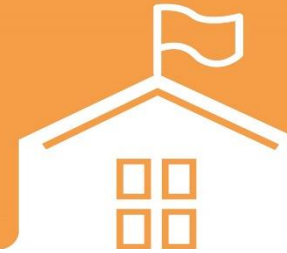


# 04

## Webhook 연결하기



# New Item 생성



함께가요 미래로!  
Enabling People

Enter an item name

» Required field

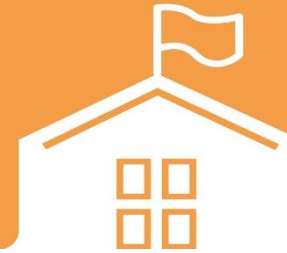


## Freestyle project

이것은 Jenkins의 주요 기능입니다. Jenkins은 어느 빌드 시스템과 어떤 SCM(형상관리)으로 묶인 당신의 프로젝트를 빌드할 것이고, 소프트웨어 빌드보다 다른 어떤 것에 자주 사용될 수 있습니다.

젠킨스 new item 이름 & Freestyle project 클릭 후 OK

## 소스코드 관리

함께가요 미래로!  
Enabling People

Git ?

Repositories ?

Repository URL ?

`https://lab.ssafy.com/ssafy-coach-8th/bool_coach_21.git`

Credentials ?

`8gi_coach21@ssafy.com/*****`

+ Add

고급...

Add Repository

빌드 유발할 Git repository 입력

Add 해서 Confidential 추가

# 소스코드 관리-Credential 추가



Kind

Username with password

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Username ?

8gi\_coach21@ssafy.com

☐ Treat username as secret ?

Password ?

\*\*\*\*\*

ID ?

private|token

Unacceptable characters

Description ?

Add

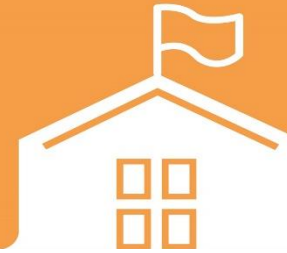
Cancel

SSAFY git 아이디 입력

SSAFY git 비밀번호 또는 private token 입력

Credential을 구별할 수 있는 임의의 ID 설정

# 소스코드 관리



함께가요 미래로!  
Enabling People

Branches to build ?

Branch Specifier (blank for 'any') ?

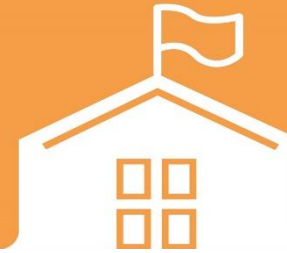
\*/master

Add Branch

Repository browser ?

(자동)

빌드를 유발시키는 **이벤트가 발생할** Branch 설정



## 빌드 유발

- ☐ 빌드를 원격으로 유발 (예: 스크립트 사용) ?
- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☒ Build when a change is pushed to GitLab. GitLab webhook URL: http://[redacted].ssafy.io:8080/project/cicdtest ?

Enabled GitLab triggers

- ☒ Push Events
- ☐ Push Events in case of branch delete
- ☒ Opened Merge Request Events

- ☐ Build only if new commits were pushed to Merge Request ?
- ☐ Accepted Merge Request Events
- ☐ Closed Merge Request Events

Rebuild open Merge Requests

Never

- ☒ Approved Merge Requests (EE-only)
- ☒ Comments

Comment (regex) for triggering a build ?

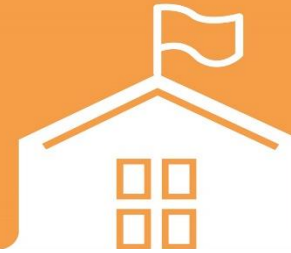
Jenkins please retry a build

고급...

저장

Apply

빌드를 유발시키는 이벤트 설정



Secret token ?

a1babac

Generate

Clear

고급 버튼 누르면 나오는 Secret token 생성하기

# Gitlab 웹훅 연결하기



함께가요 미래로!  
Enabling People

Project repository

> Setting

> Webhooks

ssafy-coach-8th > test > Webhook Settings

Search page

### Webhooks

Webhooks enable you to send notifications to web applications in response to events in a group or project. We recommend using an [integration](#) in preference to a webhook.

**URL**  
http://p.ssafy.io:8080/project/cicdtest

**Secret token**  
.....

Used to validate received payloads. Sent with the request in the `X-Gitlab-Token` HTTP header.

**Trigger**

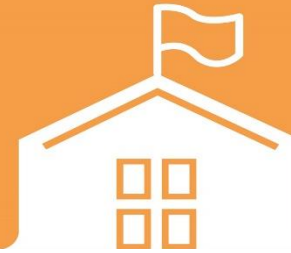
- ☒ Push events  
main  
Push to the repository.
- ☐ Tag push events  
A new tag is pushed to the repository.
- ☐ Comments  
A comment is added to an issue or merge request.
- ☐ Confidential comments  
A comment is added to a confidential issue.
- ☐ Issues events  
An issue is created, updated, closed, or reopened.
- ☐ Confidential issues events  
A confidential issue is created, updated, closed, or reopened.
- ☒ Merge request events  
A merge request is created, updated, or merged.
- ☐ Job events

URL은 hook 연결할 젠킨스 프로젝트 주소

Secret token은 앞에서 생성한 key



# Gitlab 웹훅 연결하기



함께가요 미래로!  
Enabling People

The screenshot displays the GitLab Project Hooks configuration page. At the top, a blue notification bar states "Hook executed successfully: HTTP 200". Below this, the "Project Hooks (1)" section lists a single hook with the URL "http://[redacted].ssafy.io:8080/project/cicdtest". The hook is configured for "Push Events" and "Merge Requests Events", with "SSL Verification: enabled". To the right of the hook entry are buttons for "Test", "Edit", and "Delete". A red box highlights the "Test" button, with a red arrow pointing from it to the notification bar. Another red arrow points from the "Test" button to a log entry at the bottom. The log entry shows a green checkmark, the ID "#4", the timestamp "2023. 1. 25. 오후 4:46", and the message "Started by GitLab push by 김호은(실습코치)".

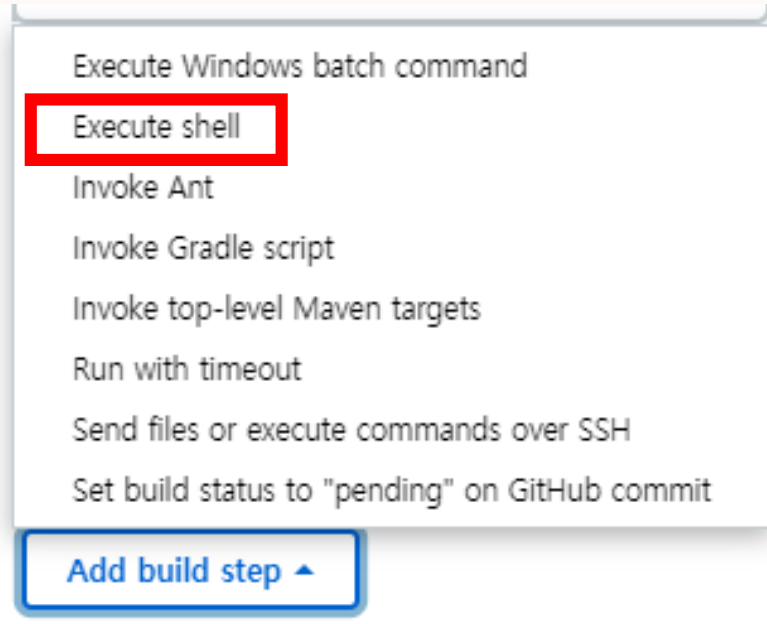
생성한 Webhook  
Test 해서 성공여부 확인하기

# 05

## 이제 진짜 배포하기

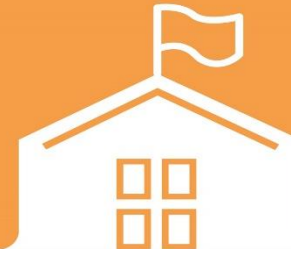


# Build Steps



빌드 명령어를 직접 작성하기 위해  
Execute shell 활용

# Build Steps



함께가요 미래로!  
Enabling People

≡ Execute shell ?

Command

See [the list of available environment variables](#)

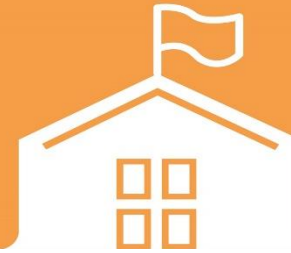
```
export JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64
export PATH="$PATH:$JAVA_HOME/bin"
cd /var/lib/jenkins/workspace/cicdtest/cicdtest_be
chmod +x ./gradlew
./gradlew clean bootJar
```

고급...

## 빌드를 위한 shell 명령어 입력

Jenkins에서 수행할 환경변수 추가

# Build Steps



≡ Execute shell ?

Command

See [the list of available environment variables](#)

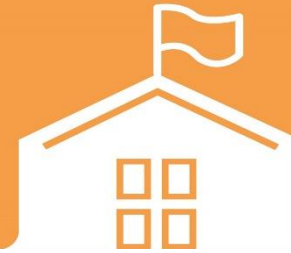
```
export JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64
export PATH="$PATH:$JAVA_HOME/bin"
cd /var/lib/jenkins/workspace/cicdtest/cicdtest_be
chmod +x ./gradlew
./gradlew clean bootJar
```

고급...

## 빌드를 위한 shell 명령어 입력

Gradlew가 있는 경로로 이동

# Build Steps



함께가요 미래로!  
Enabling People

≡ Execute shell ?

Command

See [the list of available environment variables](#)

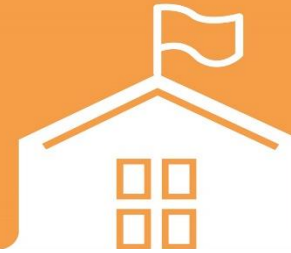
```
export JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64
export PATH="$PATH:$JAVA_HOME/bin"
cd /var/lib/jenkins/workspace/cicdtest/cicdtest_be
chmod +x ./gradlew
./gradlew clean bootJar
```

권한을 부여하고, 빌드 실행

고급...

빌드를 위한 shell 명령어 입력

# 빌드 후 조치

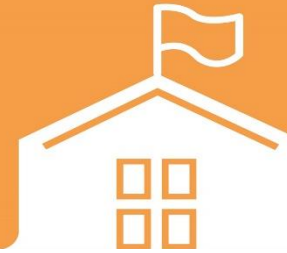


- Aggregate downstream test results
- Archive the artifacts
- Build other projects
- Post build task
- Publish JUnit test result report
- Record fingerprints of files to track usage
- Git Publisher
- Accept GitLab merge request on success
- Add note with build status on GitLab merge requests
- Add vote for build status on GitLab merge requests
- E-mail Notification
- Editable Email Notification
- Publish build status to GitLab
- Send build artifacts over SSH**
- Set GitHub commit status (universal)
- Set build status on GitHub commit [deprecated]
- Delete workspace when build is done

빌드 후 조치 추가 ▲

Send build artifacts over SSH 클릭  
빌드 된 파일을 서버로 옮기기 위함

# 빌드 후 조치



SSH Publishers

SSH Server

Name ?

coach21

고급...

Transfers

Transfer Set

Source files ?

cicdtest\_be/build/libs/\*.jar

Remove prefix ?

cicdtest\_be/build/libs/

Remote directory ?

Exec command ?

cd /home/jenkins/workspace  
nohup ./build.sh > /dev/null 2>&1 &

고급...

All of the transfer fields (except for Exec timeout) support substitution of [Jenkins environment variables](#)

publish over SSH configuration에 등록된 서버 선택

옮길 파일 경로, jar 파일만 이동!

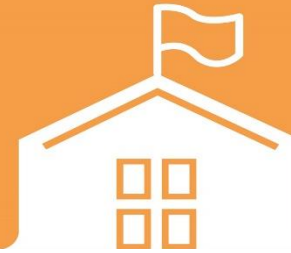
파일명만 추출하기 위한 작업

빌드 파일들이 위치한 workspace dir에서

Build.sh 쉘파일을 실행하라는 명령어



# 빌드 후 조치



- ✓ `cd /home/jenkins/workspace`
- ✓ `nohup ./build.sh > /dev/null 2>&1 &`

build.sh가 가 있는 경로로 이동해서

build.sh라는 쉘파일을 실행하고,  
백그라운드에서 계속 실행되도록 하는 명령어

Exec command ?

```
cd /home/jenkins/workspace  
nohup ./build.sh > /dev/null 2>&1 &
```

All of the transfer fields (except for Exec timeout) support substitution of [Jenkins environment variables](#)

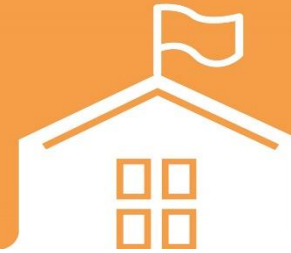
고급...



# 빌드 후 조치-셸파일 뜯어보기

```
#!/bin/bash  
  
fuser -k 8081/tcp  
fileName=$(ls *.jar)  
  
file=./$fileName  
  
java -jar $file &
```

build.sh 내용을 살펴보면  
가동 되고 있는 port를 중단시키고  
Jar file을 찾아서  
실행하라는 명령어로 작성함



#44

2023. 1. 26. 오후 7:38

빌드를 성공적으로 마치면  
초록 체크가 떠 있음!



## Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

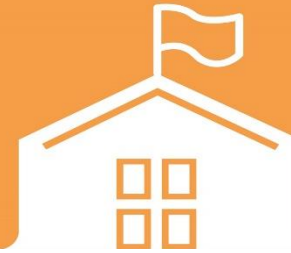
Fri Jan 27 00:38:28 UTC 2023

There was an unexpected error (type=Not Found, status=404).

작업이 성공하면

서버 주소 : 포트 접속 시 확인 가능!

## 마치며

함께가요 미래로!  
Enabling People

❗ #26	2023. 1. 26. 오후 5:54		
❗ #25	2023. 1. 26. 오후 5:50		
❗ #24	2023. 1. 26. 오후 6:52	❗ #35	2023. 1. 26. 오후 6:52
		❗ #34	2023. 1. 26. 오후 6:34
✅ #23	2023. 1. 26. 오후 7:38	Started by GitLab pu	✅ #44
❗ #22	2023. 1. 26. 오후 7:37		✅ #43
❗ #21	2023. 1. 26. 오후 7:36	Started by GitLab pu	✅ #42
❗ #20	2023. 1. 26. 오후 7:35	❗ #32	2023. 1. 26. 오후 7:35
❗ #19	2023. 1. 26. 오후 7:31	❗ #31	2023. 1. 26. 오후 7:31
❗ #18	2023. 1. 26. 오후 7:25	❗ #30	2023. 1. 26. 오후 7:25
		❗ #29	2023. 1. 26. 오후 7:15
		❗ #28	2023. 1. 26. 오후 7:09
		❗ #27	2023. 1. 26. 오후 7:00
			✅ #36

젠킨스 많이 힘들죠?

괜찮아요..

원래 이상한 할아버지예요...

# 감사합니다

