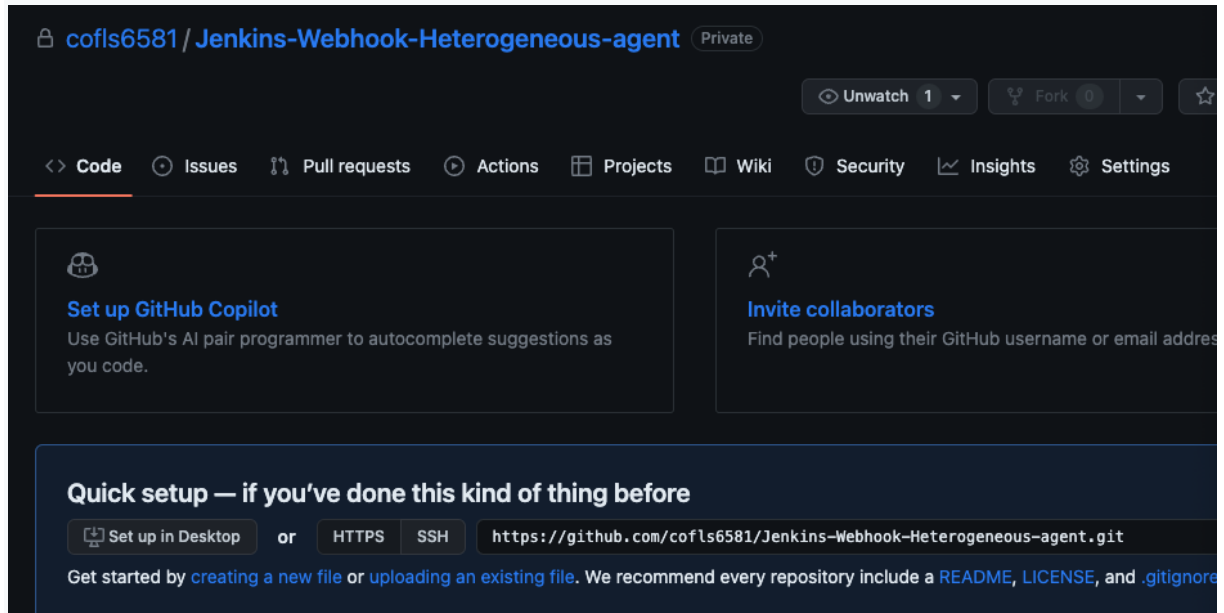


소프트웨어공학

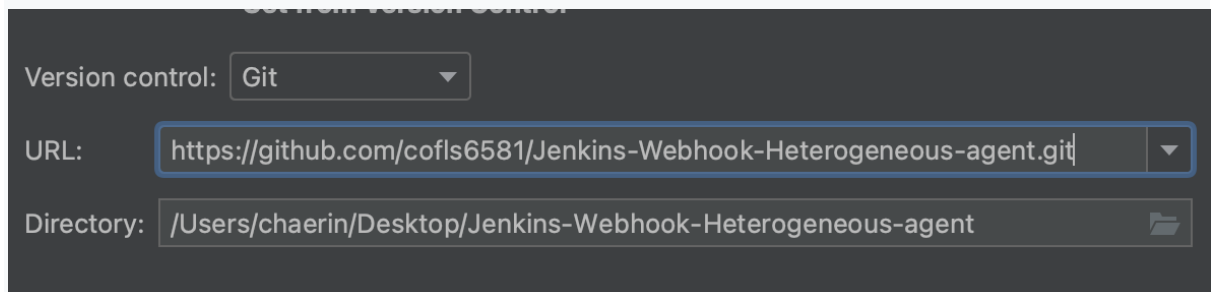
## Jenkinsfile with Github Webhook for heterogeneous agent 과제

컴퓨터공학전공 B911151 이채린

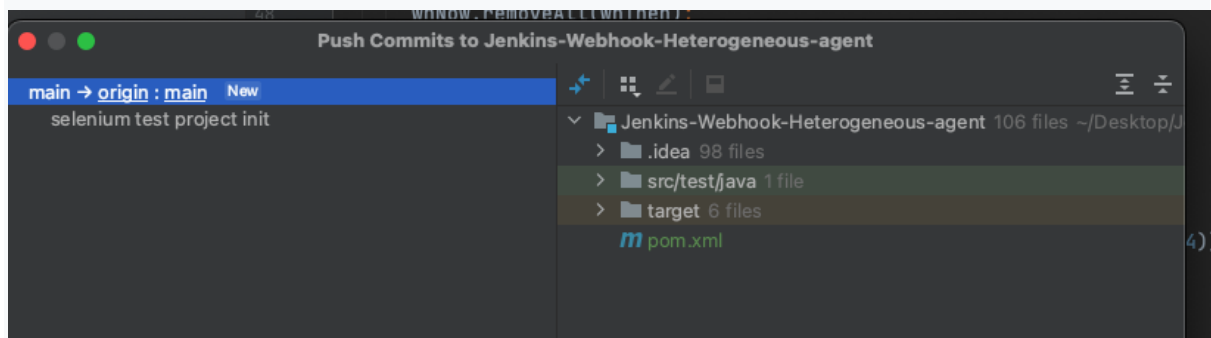
(1) IntelliJ 등 IDE에서 작업한 코드를 IDE에서 각자의 Github repo로 자동 업로드합니다.

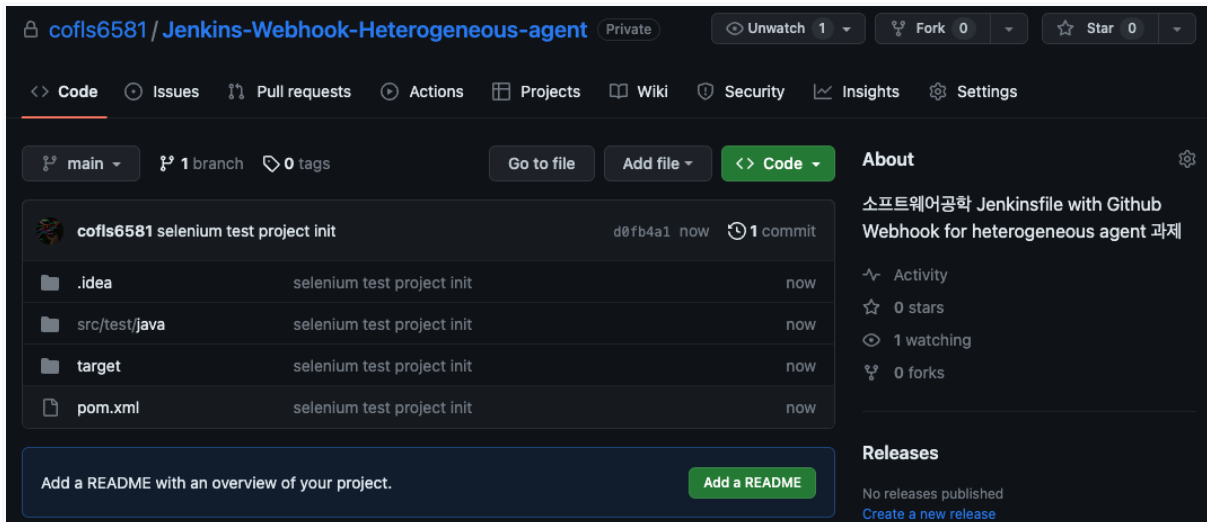


레포지토리 생성



인텔리제이 IDE에서 자동 업로드





(2) Github repo에 Jenkinsfile을 업로드합니다.

Jenkinsfile에 들어갈 ec2 agent 생성 과정

<input checked="" type="checkbox"/>	jenkins agent	i-0147d4818374dc822	실행 중	t2.micro
-------------------------------------	---------------	---------------------	------	----------

생성한 ec2에 jdk 11, maven, git 설치

```
ubuntu@ip-172-31-38-82:~$ java -version
openjdk version "11.0.19" 2023-04-18
OpenJDK Runtime Environment (build 11.0.19+7-post-Ubuntu-0ubuntu120.04.1)
OpenJDK 64-Bit Server VM (build 11.0.19+7-post-Ubuntu-0ubuntu120.04.1, mixed mode, sharing)
```

```
ubuntu@ip-172-31-38-82:~$ mvn -version
Apache Maven 3.6.3
Maven home: /usr/share/maven
Java version: 11.0.19, vendor: Ubuntu, runtime: /usr/lib/jvm/java-11-openjdk-amd64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "5.15.0-1036-aws", arch: "amd64", family: "unix"
```

```
ubuntu@ip-172-31-38-82:~$ git --version
git version 2.25.1
```

jenkins user 생성

```
ubuntu@ip-172-31-38-82:~$ sudo useradd -m jenkins
```

jenkins 홈 디렉토리 생성

```
root@ip-172-31-38-82:/home# sudo -u jenkins mkdir jenkins/.ssh
```

mac ssh 키 생성

```
x chaerin@chaelin-ui-MacBookPro ~ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/chaerin/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /Users/chaerin/.ssh/id_rsa
Your public key has been saved in /Users/chaerin/.ssh/id_rsa.pub
```

```
sh-3.2# cd .ssh
sh-3.2# ls
id_rsa      id_rsa.pub  known_hosts  known_hosts.old
```

id\_rsa.pub을 ec2 authorized\_keys에 저장

```
root@ip-172-31-38-82:/home/jenkins/.ssh# sudo -u jenkins vi authorized_keys
```

mac에서 ssh 연결 확인

```
chaerin@chaelin-ui-MacBookPro ~$ ssh jenkins@54.180.24.58
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1036-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon Jun  5 19:56:32 UTC 2023

System load:  0.08               Processes:            106
Usage of /:   31.4% of 7.57GB    Users logged in:     1
Memory usage: 23%               IPv4 address for eth0: 172.31.38.82
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

42 updates can be applied immediately.
28 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

2 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

New release '22.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
```

mac에서 젠킨스 ec2 연결위한 추가 설정

```
chaerin@chaelin-ui-MacBookPro ~$ cp .ssh/known_hosts .jenkins/.ssh
```

demoAgent 노드 생성 완료

demoAgent log [Jenkins]

localhost:8080/computer/demoAgent/log

> Nodes > demoAgent > Log

```
[06/06/23 05:08:02] [SSH] Checking java version of /home/jenkins/jdk/bin/java
Couldn't figure out the Java version of /home/jenkins/jdk/bin/java
sh: 1: /home/jenkins/jdk/bin/java: not found

[06/06/23 05:08:02] [SSH] Checking java version of java
[06/06/23 05:08:02] [SSH] java -version returned 11.0.19.
[06/06/23 05:08:02] [SSH] Starting sftp client.
[06/06/23 05:08:02] [SSH] Copying latest remoting.jar...
[06/06/23 05:08:03] [SSH] Copied 1,370,647 bytes.
Expanded the channel window size to 4MB
[06/06/23 05:08:03] [SSH] Starting agent process: cd "/home/jenkins" && java -jar remoting.jar -workDir /home/jenkins -jar-cache /home/jenkins/remoting/jarCache
Jun 05, 2023 8:08:03 PM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/jenkins/remoting as a remoting work directory
Jun 05, 2023 8:08:03 PM org.jenkinsci.remoting.engine.WorkDirManager setupLogging
INFO: Both error and output logs will be printed to /home/jenkins/remoting
<==[JENKINS REMOTING CAPACITY]==>channel started
Remoting version: 3107.v665000b_51092
Launcher: SSHLauncher
Communication Protocol: Standard in/out
This is a Unix agent
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by jenkins.slaves.StandardOutputSwapper$ChannelSwapper to constructor java.io.FileDescriptor(int)
WARNING: Please consider reporting this to the maintainers of jenkins.slaves.StandardOutputSwapper$ChannelSwapper
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Evacuated stdout
Agent successfully connected and online
```

Nodes

+ New Node

S	이름 ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Mac OS X (x86_64)	In sync	759.35 GB	0 B	759.35 GB	0ms
	demoAgent		N/A	N/A	N/A	N/A	N/A
	수집된 데이터	13 ms	17 ms	11 ms	46 min	8 ms	5 ms

생성한 agent 들어간 Jenkinsfile 생성

Jenkins-Webhook-Heterogeneous-agent / Jenkinsfile

Go to file

cofls6581 feat: create Jenkinsfile

aa53896 · now History

Code

Blame

17 lines (17 loc) · 399 Bytes

Raw

```
1 pipeline {
2   agent {
3     label "demoAgent"
4   }
5   stages {
6     stage('git') {
7       steps {
8         git branch: 'main', url: 'https://github.com/cofls6581/Jenkins-Webhook-Heterogeneous-agent.git'
9       }
10    }
11    stage('mvn') {
12      steps {
13        sh "mvn -Dmaven.test.failure.ignore=true clean test"
14      }
15    }
16  }
17 }
```

(3) Pipeline 프로젝트는 Github webhook을 사용하여 IDE에서 업로드된 변경된 코드가 commit 되면 자동으로 Agent에서 test가 수행되는 것을 확인합니다.

ngrok 설치 후 8080 http로 ngrok 생성

```
ngrok http 8080

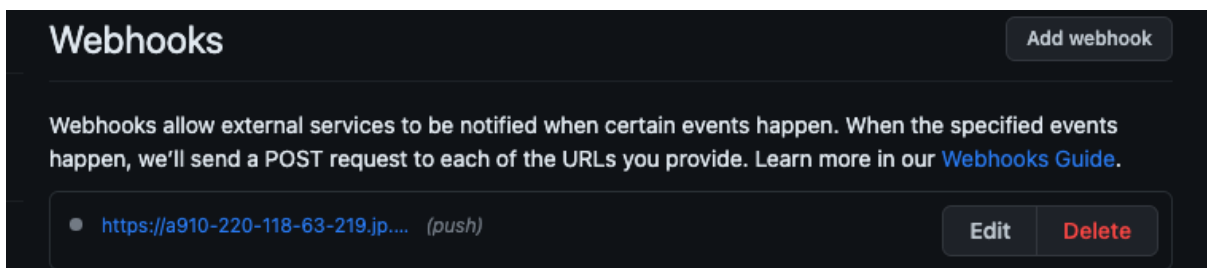
ngrok
(Ctrl+C to quit)

Announcing ngrok-go: The ngrok agent as a Go library: https://ngrok.com/go

Session Status      online
Session Expires     1 hour, 59 minutes
Terms of Service     https://ngrok.com/tos
Version             3.3.0
Region              Japan (jp)
Latency              37ms
Web Interface        http://127.0.0.1:4040
Forwarding            https://a910-220-118-63-219.jp.ngrok.io -> http://localhost:8080

Connections          ttl    opn    rt1    rt5    p50    p90
0                    0      0      0.00   0.00   0.00   0.00
```

생성한 ngrok로 깃헙 레포지토리에 웹훅 설정



젠킨스 파이프라인 생성

웹훅 트리거 설정+레포지토리의 Jenkinsfile로 스크립트 실행

#### Pipeline

##### Definition

Pipeline script from SCM

##### SCM ?

Git

##### Repositories ?

###### Repository URL ?

https://github.com/cofls6581/2023\_swe.git

###### Credentials ?

- none -

Add

고급

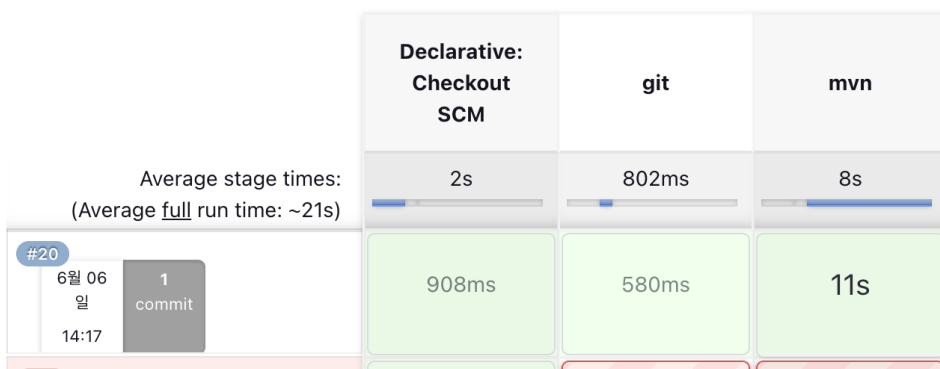
Script Path ?

Jenkinsfile

인텔리제이 IDE에서 push 후 작동 확인

## Pipeline test

### Stage View



콘솔 출력

```
Started by GitHub push by cofls6581
Obtained Jenkinsfile from git https://github.com/cofls6581/2023_swe.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on demoAgent in /home/jenkins/workspace/test
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
The recommended git tool is: git
No credentials specified
Fetching changes from the remote Git repository
Checking out Revision 277c2ada581c62961376ddaab86462bbd12d4285 (refs/remotes/origin/main)
Commit message: "fix: 레포지토리명 변경"
```