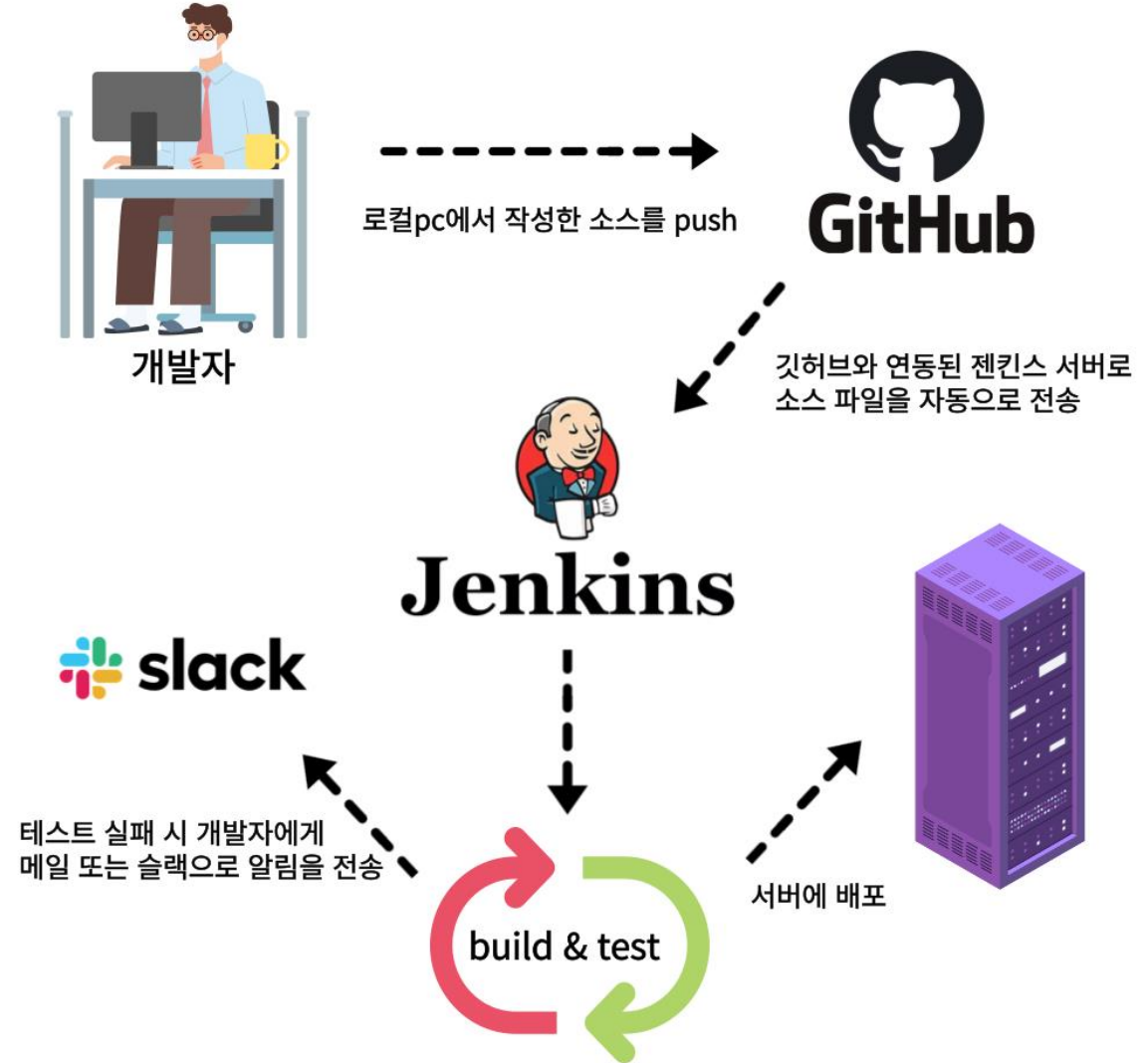
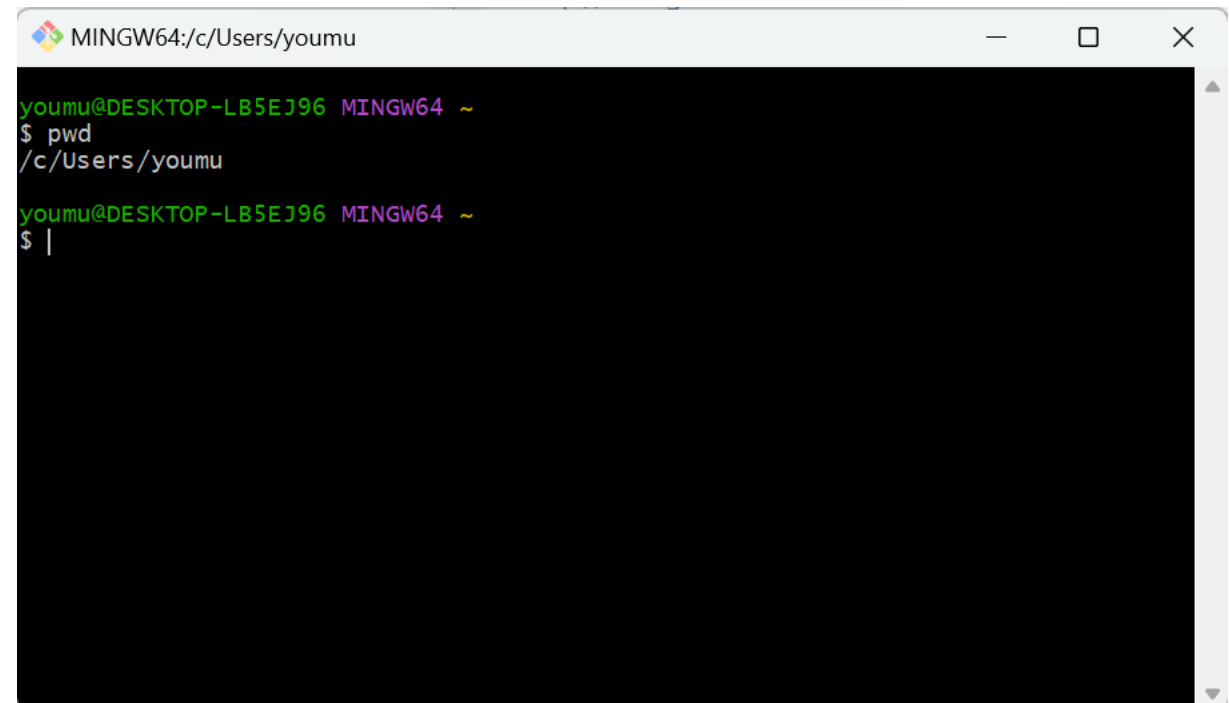
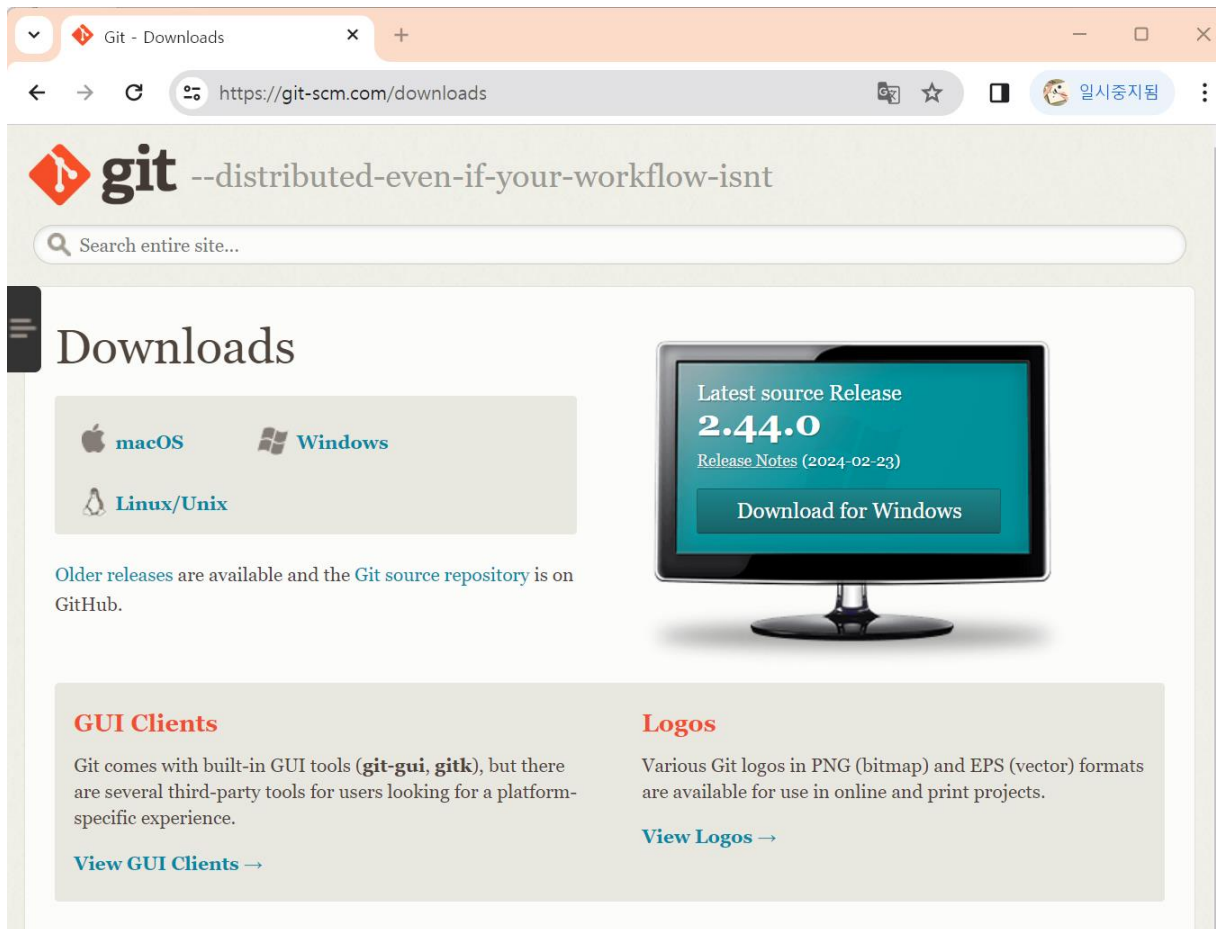


CI/CD 실습



GIT 설치



GIThub

- Git을 기반으로 한 웹 기반 호스팅 서비스
- Git의 모든 기능을 제공할 뿐만 아니라, 프로젝트 협업을 위한 추가적인 기능들을 제공.(버그 추적, 기능 요청, 작업 관리, 위키, 그리고 소셜 네트워킹 같은 기능들이 포함)

깃(GIT)

- 개발 버전 관리 시스템(Version Control System)
- 설치 후 자신의 컴퓨터(로컬)에서 동작
- 개발 중 기능 수정/업그레이드시 각 단계 버전별 저장, 복원 등 가능
- 버전 파일을 공유하면 협업 가능

오프라인

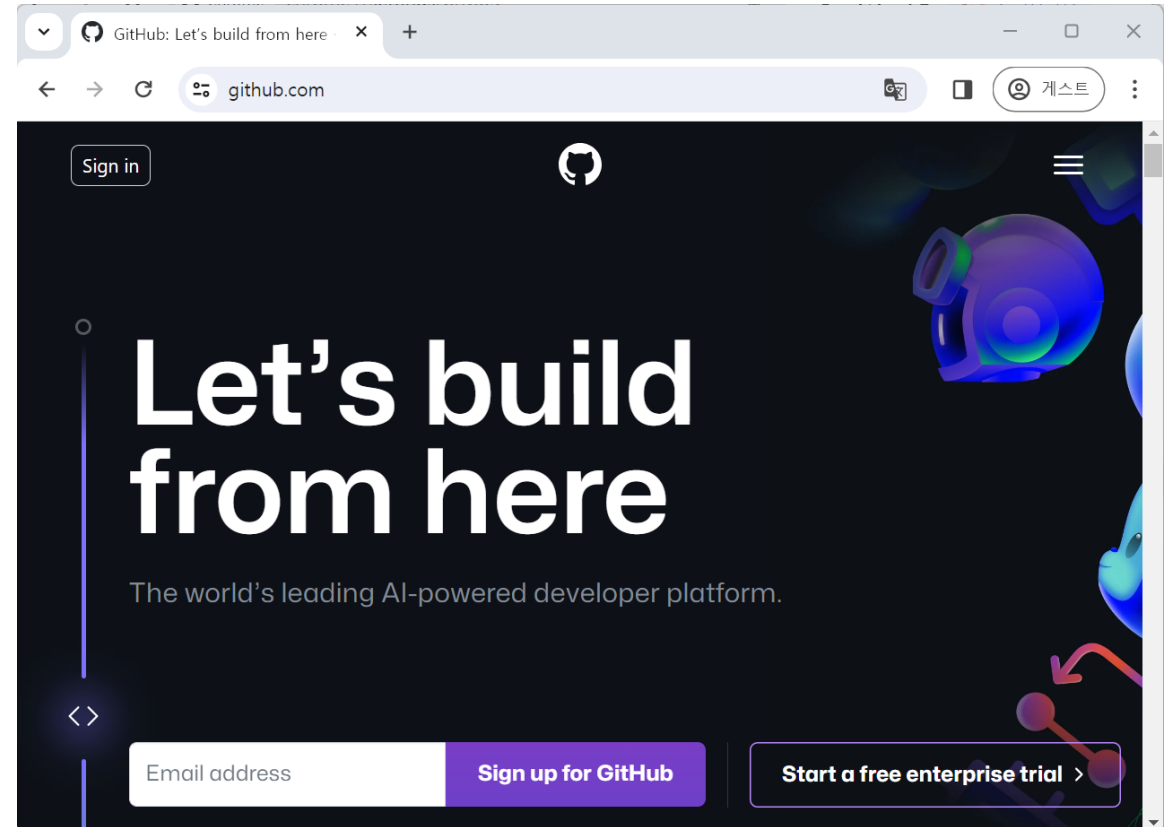
깃허브(GITHUB)

- GIT을 기반으로 한 웹 호스팅 서비스
- GIT을 클라우드로 옮겨놓은 것
- 온라인 상에서 버전관리 가능
- 다수 개발자가 한 프로젝트를 언제든지 개발할 수 있어 효율적인 협업 가능

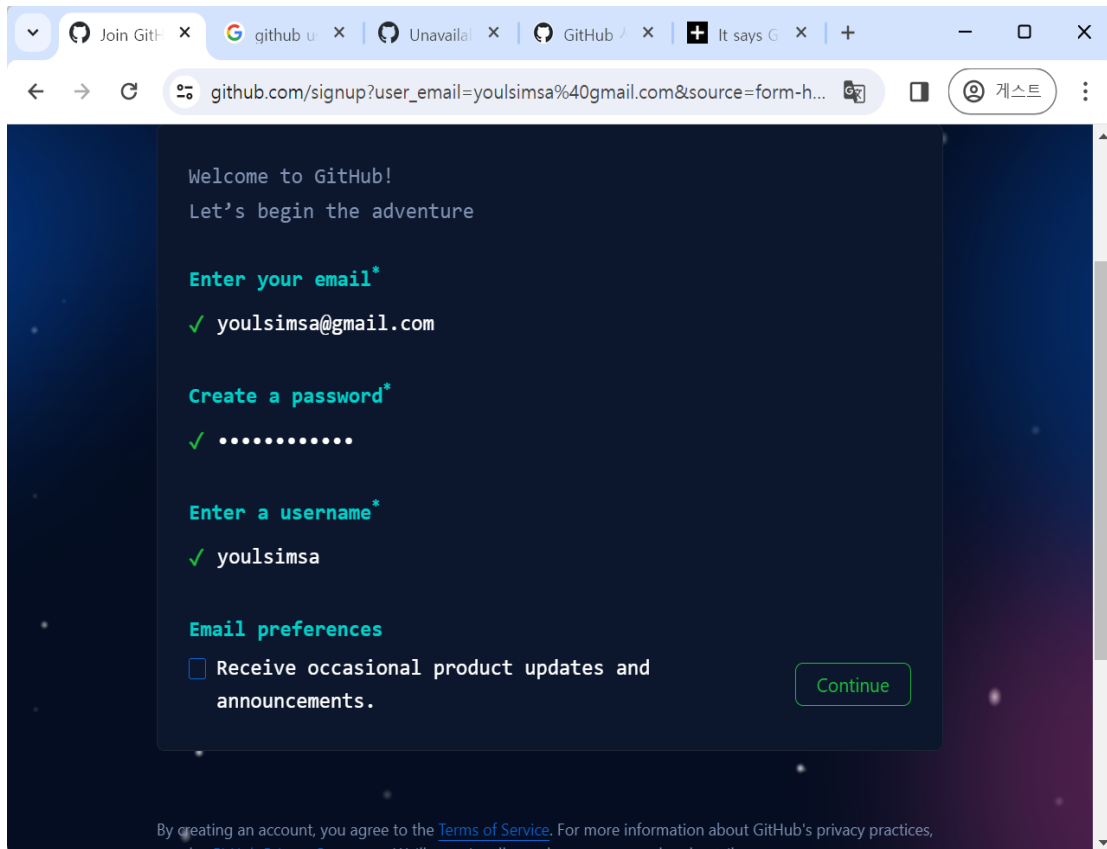
 온라인

GitHub 가입

- 1. 깃헙 가입 페이지 이동
- - 다음 깃헙 가입 페이지로 이동하자.
- <https://github.com/>
- - "Sign up for GitHub"
버튼 클릭



GitHub 가입



Join GitHub

github.com/signup?user_email=yousimsa%40gmail.com&source=form-h...

Welcome to GitHub!
Let's begin the adventure

Enter your email*

✓ youlsimsa@gmail.com

Create a password*

✓

Enter a username*

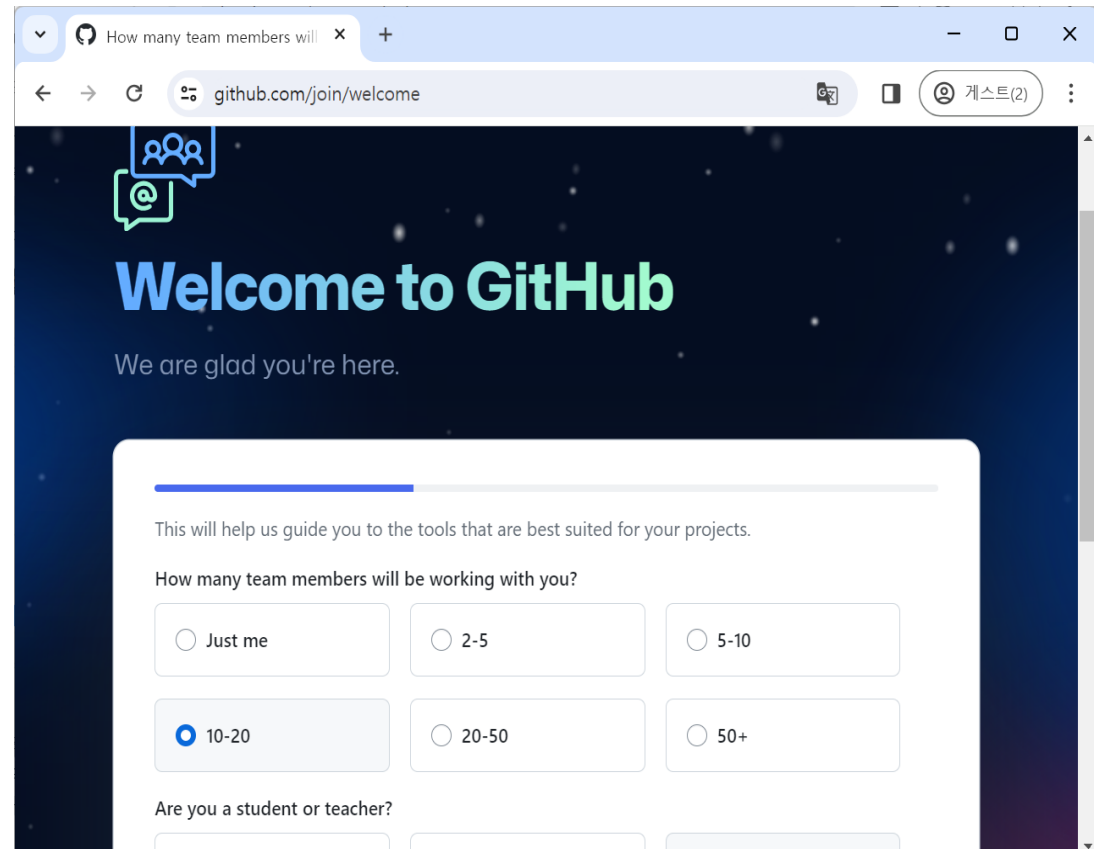
✓ youlsimsa

Email preferences

☐ Receive occasional product updates and announcements.

Continue

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll never sell your account details to anyone.



How many team members will

github.com/join/welcome

Welcome to GitHub

We are glad you're here.

This will help us guide you to the tools that are best suited for your projects.

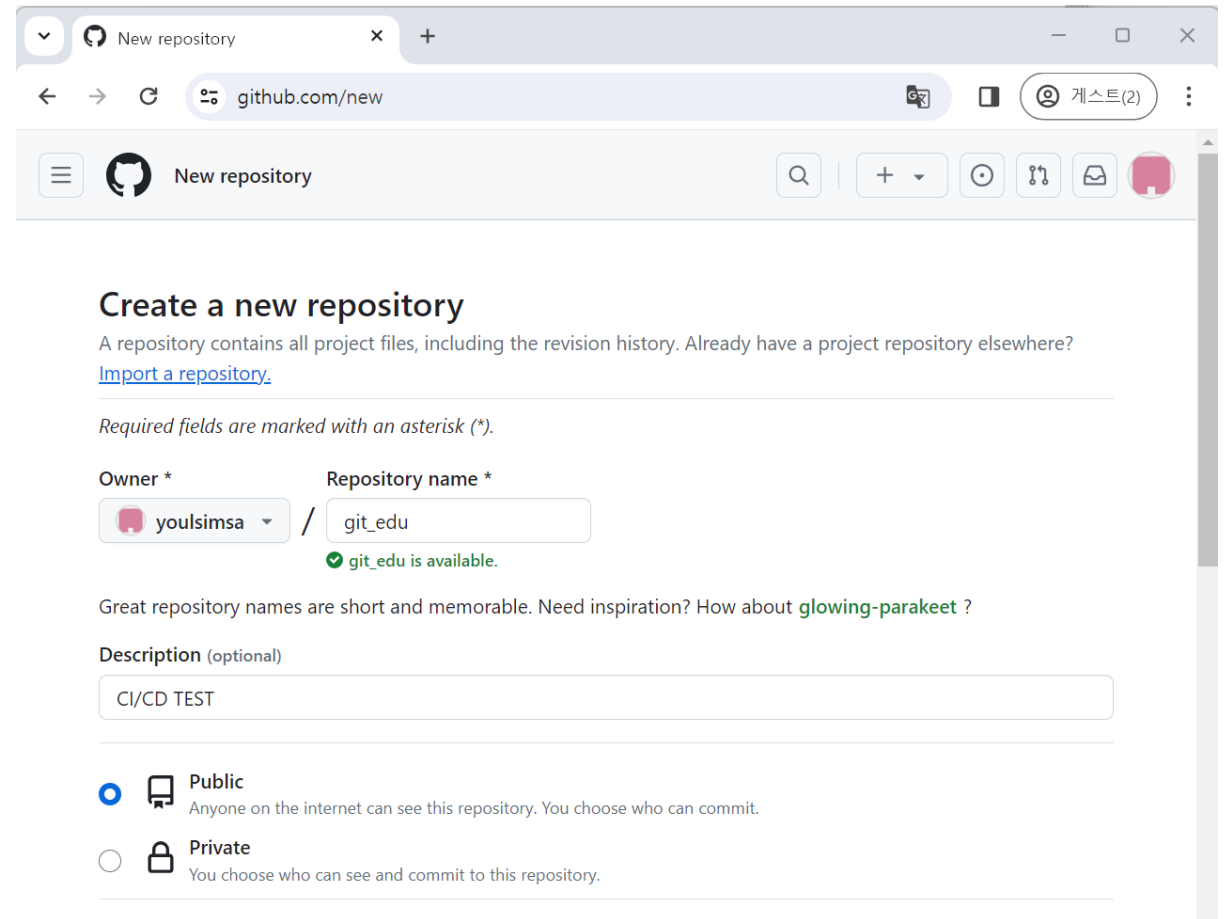
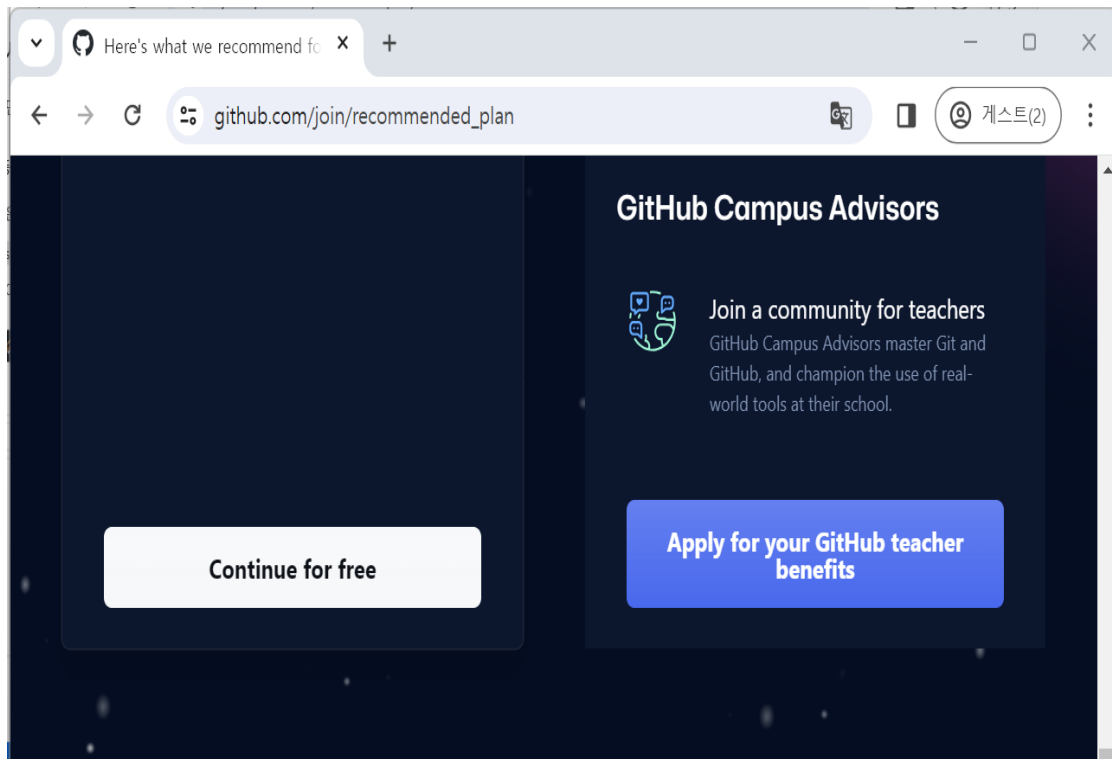
How many team members will be working with you?

☐ Just me ☐ 2-5 ☐ 5-10

☒ 10-20 ☐ 20-50 ☐ 50+

Are you a student or teacher?

GitHub 가입



GIThub REPO생성

The image shows a multi-step process for creating a new repository on GitHub. It features three overlapping browser window screenshots. The top window shows the 'New repository' page with the URL 'https://github.com/new'. The middle window shows the user's profile page 'YYouri' with the URL 'https://github.com/YYouri?tab=repositories'. The bottom window shows the 'Create a new repository' form. The form includes fields for 'Owner' (YYouri), 'Repository name' (git_cicd), and a 'Description' field. It also has radio buttons for 'Public' and 'Private' visibility, a checkbox for 'Add a README file', a dropdown for '.gitignore template', and a dropdown for 'License'. A green 'Create repository' button is at the bottom right.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * / Repository name *

YYouri / git_cicd

git_cicd is available.

Great repository names are short and memorable. Need inspiration? How about [laughing-broccoli](#) ?

Description (optional)

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

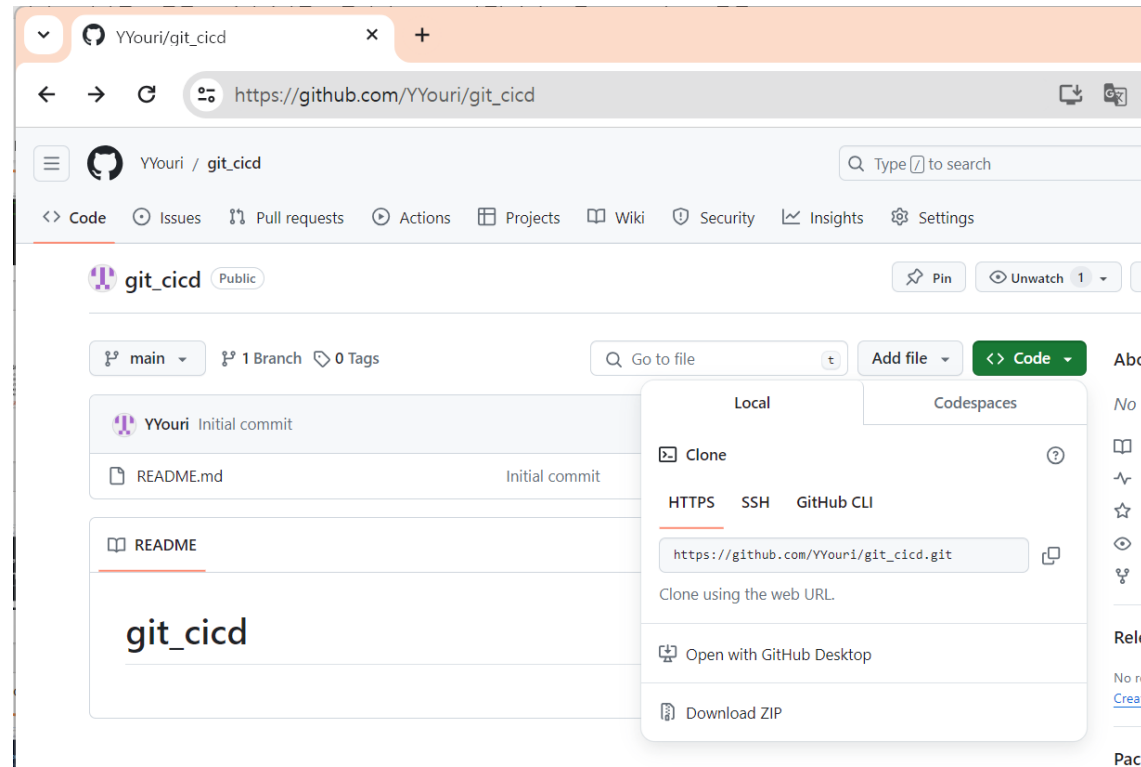
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set `main` as the default branch. Change the default name in your settings.

☐ You are creating a public repository in your personal account.

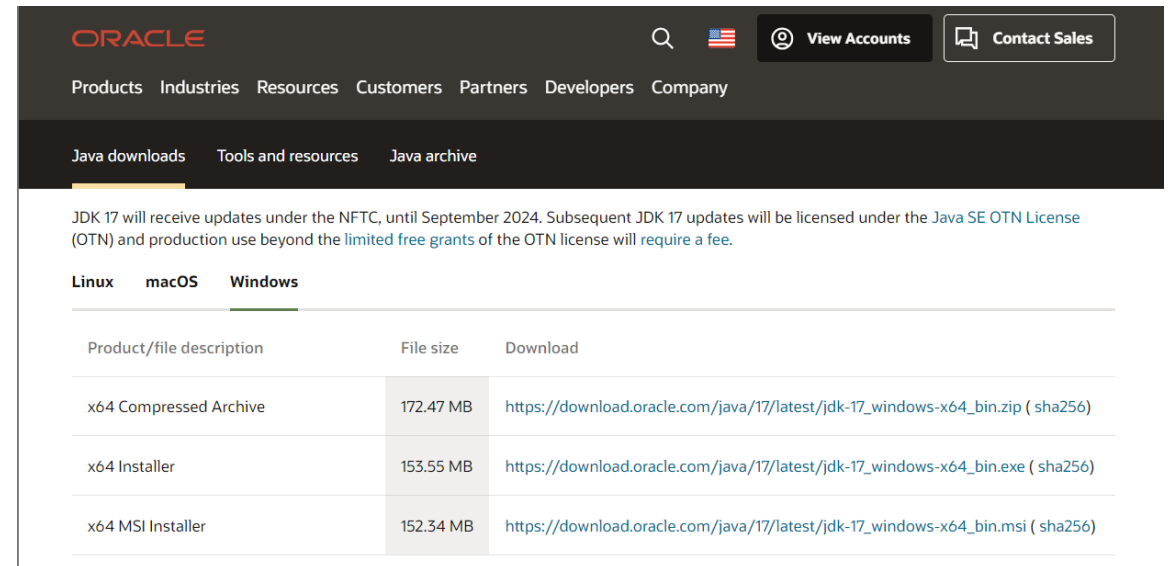
[Create repository](#)

GIThub REPO생성



개발환경 준비

- 1. JDK 17다운로드
- <https://www.oracle.com/java/technologies/downloads/#jdk17-windows>

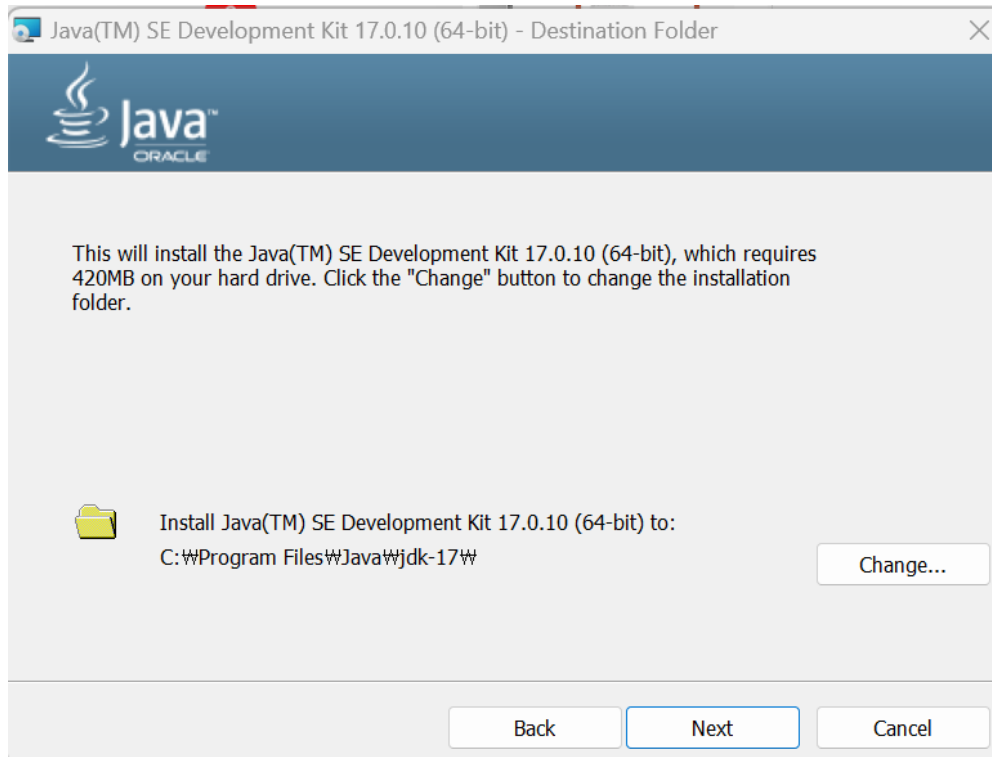


The screenshot shows the Oracle Java Downloads page for JDK 17. The page has a dark header with the Oracle logo, a search icon, a US flag, and links for 'View Accounts' and 'Contact Sales'. Below the header is a navigation bar with 'Products', 'Industries', 'Resources', 'Customers', 'Partners', 'Developers', and 'Company'. The main content area has tabs for 'Java downloads', 'Tools and resources', and 'Java archive'. A notice states that JDK 17 will receive updates under the NFTC until September 2024, and subsequent updates will be licensed under the Java SE OTN License (OTN) and production use beyond the limited free grants of the OTN license will require a fee. Below the notice are tabs for 'Linux', 'macOS', and 'Windows'. The 'Windows' tab is selected, showing a table with three columns: 'Product/file description', 'File size', and 'Download'.

Product/file description	File size	Download
x64 Compressed Archive	172.47 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip (sha256)
x64 Installer	153.55 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe (sha256)
x64 MSI Installer	152.34 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi (sha256)

개발환경 준비

2. JDK 17 설치



개발환경 준비

2. JDK 17 설치

Java version 이 다를 경우 시스템 변수 확인

```
Windows PowerShell
PS C:\Users\youmu> java
Usage: java [options] <mainclass> [args...]
        (to execute a class)
or java [options] -jar <jarfile> [args...]
        (to execute a jar file)
or java [options] -m <module>[/<mainclass>] [args...]
        (to execute the main class in a module)
or java [options] --module <module>[/<mainclass>] [args...]
        (to execute the main class in a module)
or java [options] <sourcefile> [args]
        (to execute a single source-file program)

Arguments following the main class, source file, -jar <jarfile>
-m or --module <module>/<mainclass> are passed as the arguments
to the main class.

where options include:

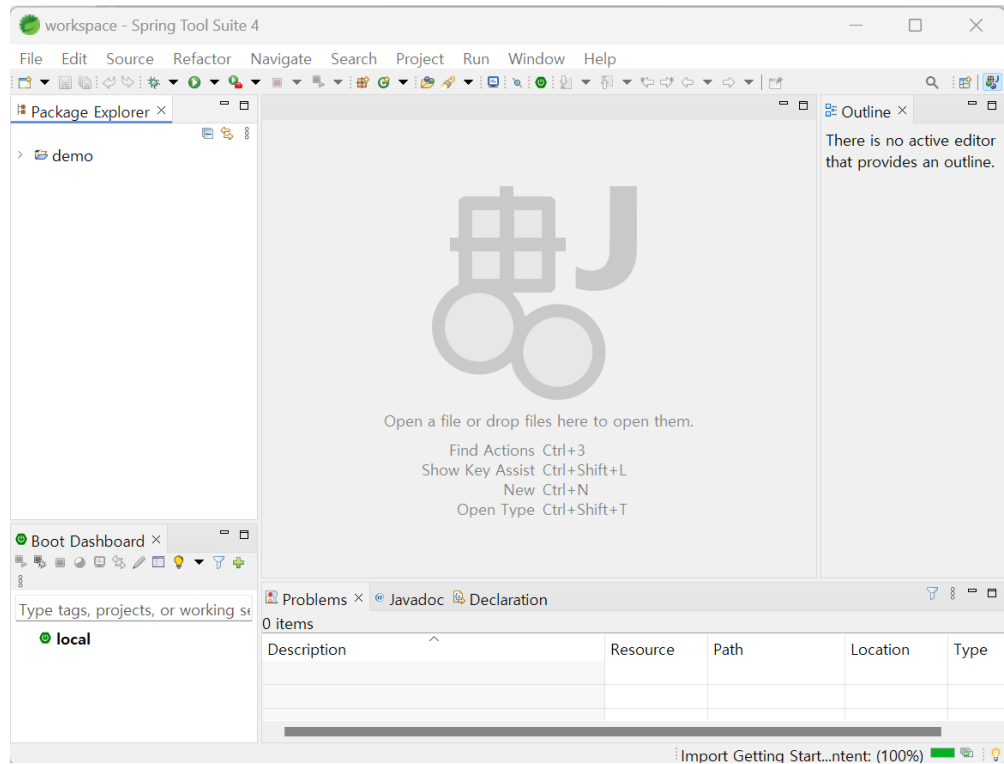
  -cp <class search path of directories and zip/jar files>
  -classpath <class search path of directories and zip/jar files>
  --class-path <class search path of directories and zip/jar files>
```



개발환경 준비

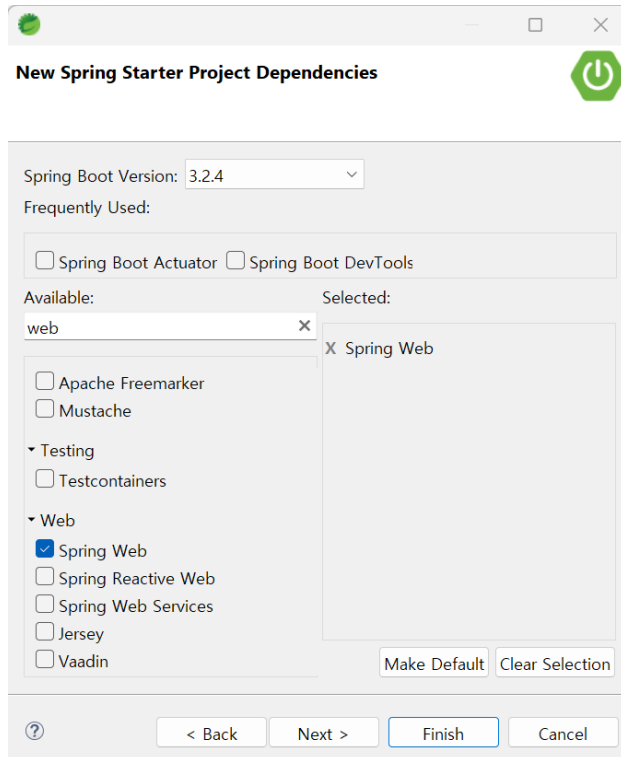
- IDE설치 (eclipse.org, spring.io/tools)
- Eclipse : 범용적 사용
 - Eclipse IDE for Java Developers는 자바 개발을 위한 IDE로 순수한 자바만을 지원
 - Eclipse IDE for Enterprise Java Developers는 자바 웹 개발을 위한 IDE로 순수 자바를 포함
- STS : 스프링 프레임워크에 특화된 IDE

개발환경 준비



Spring boot

Problem 발생 시 Gradle-refresh gradle project



New Spring Starter Project Dependencies

Spring Boot Version: 3.2.4

Frequently Used:

☐ Spring Boot Actuator ☐ Spring Boot DevTools

Available: Selected:

web X Spring Web

☐ Apache Freemarker
☐ Mustache

Testing

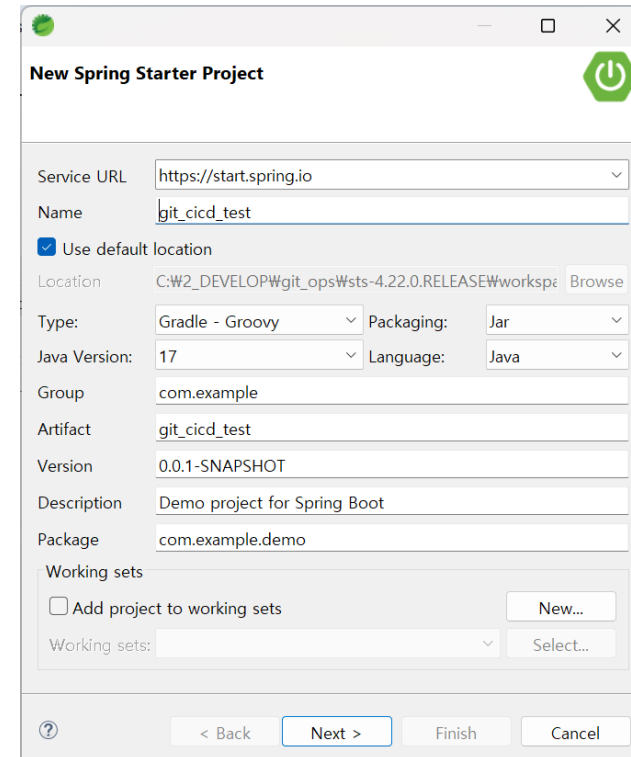
☐ Testcontainers

Web

☒ Spring Web
☐ Spring Reactive Web
☐ Spring Web Services
☐ Jersey
☐ Vaadin

Make Default Clear Selection

< Back Next > Finish Cancel



New Spring Starter Project

Service URL: https://start.spring.io

Name: git_cicd_test

☒ Use default location

Location: C:\W2_DEVELOP\git_ops\Wsts-4.22.0.RELEASE\workspace Browse

Type: Gradle - Groovy Packaging: Jar

Java Version: 17 Language: Java

Group: com.example

Artifact: git_cicd_test

Version: 0.0.1-SNAPSHOT

Description: Demo project for Spring Boot

Package: com.example.demo

Working sets

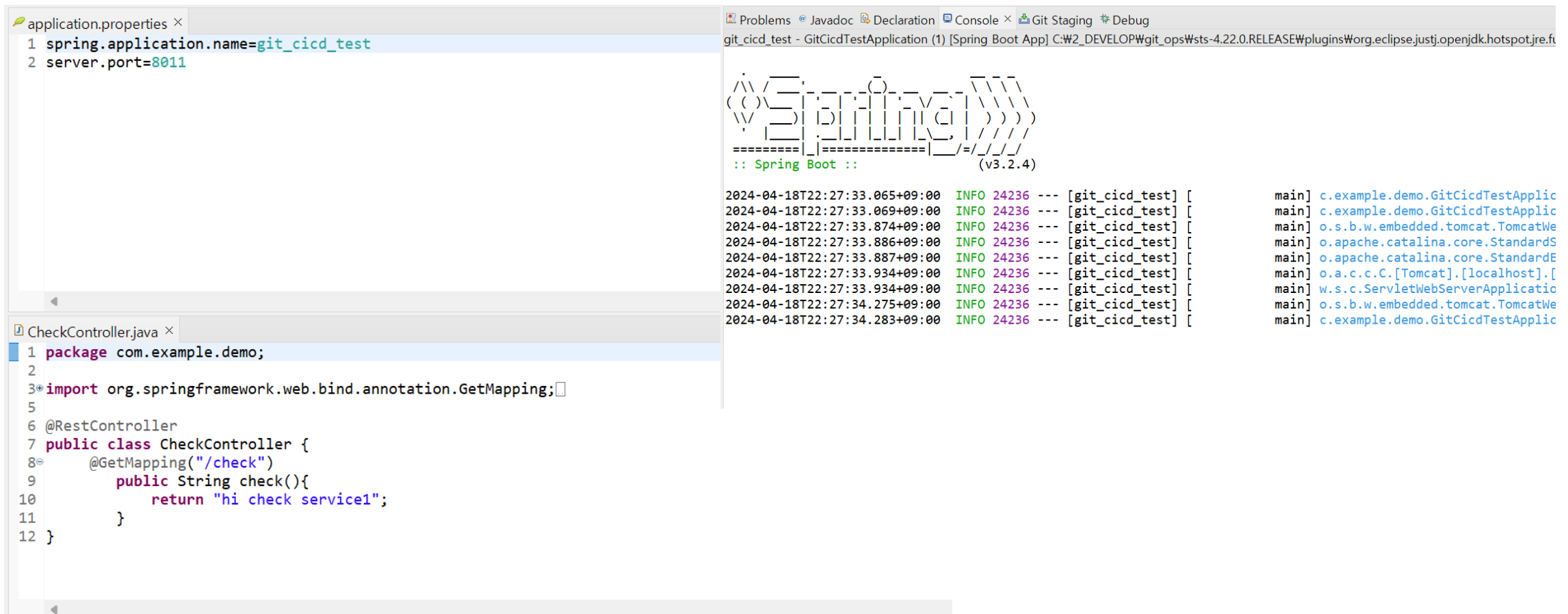
☐ Add project to working sets New...

Working sets: Select...

< Back Next > Finish Cancel

Spring boot

Run as – spring boot app



The screenshot displays an IDE with two main panels. The left panel shows the 'application.properties' file with the following content:

```
1 spring.application.name=git_cicd_test
2 server.port=8011
```

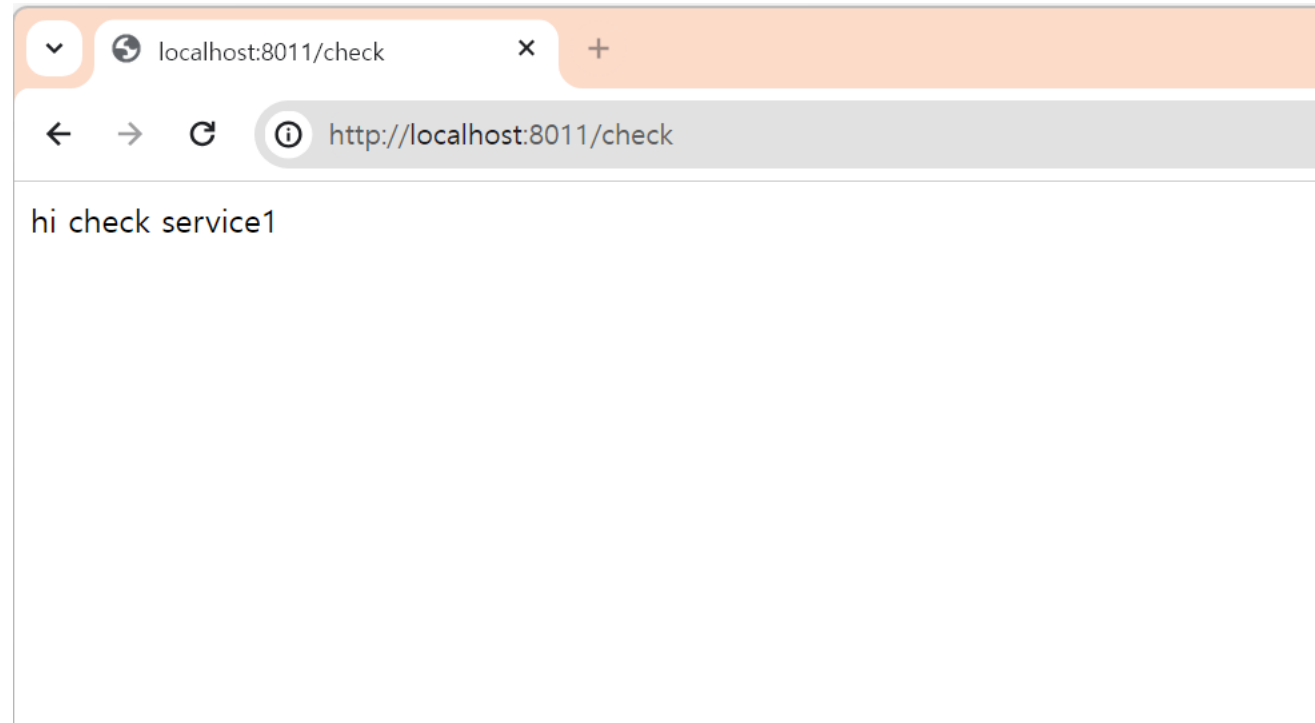
The right panel shows the 'CheckController.java' file with the following content:

```
1 package com.example.demo;
2
3 import org.springframework.web.bind.annotation.GetMapping;
4
5
6 @RestController
7 public class CheckController {
8     @GetMapping("/check")
9     public String check(){
10         return "hi check service1";
11     }
12 }
```

The right panel also shows the 'Console' output, which includes the Spring Boot logo and the following log messages:

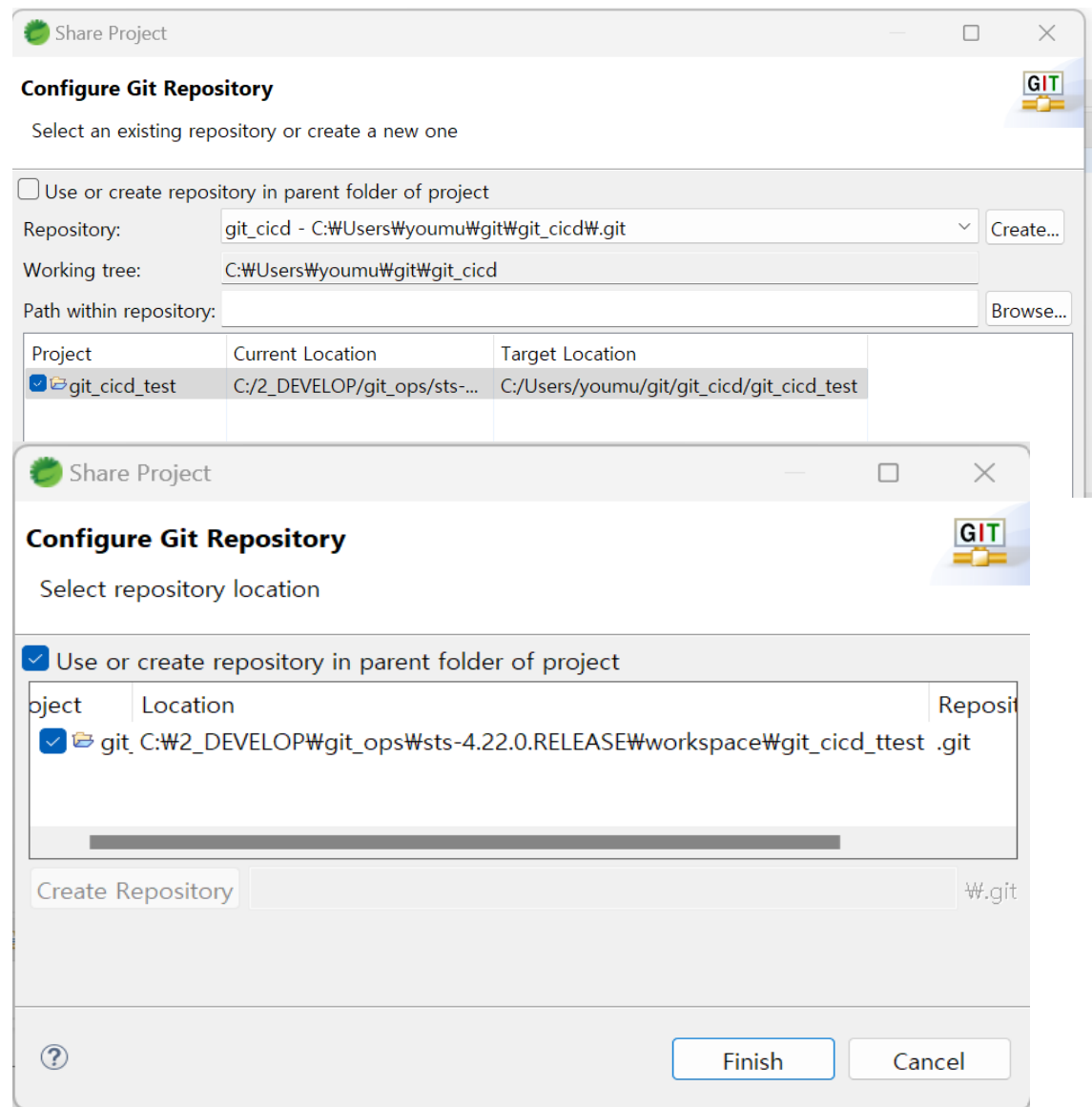
```
git_cicd_test - GitCicdTestApplication (1) [Spring Boot App] C:\W2_DEVELOP\git_ops\Wsts-4.22.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.f
:: Spring Boot :: (v3.2.4)
2024-04-18T22:27:33.065+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:33.069+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:33.874+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:33.886+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:33.887+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:33.934+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:33.934+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:33.934+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:34.275+09:00 INFO 24236 --- [git_cicd_test] [
2024-04-18T22:27:34.283+09:00 INFO 24236 --- [git_cicd_test] [
main] c.example.demo.GitCicdTestApplic
main] c.example.demo.GitCicdTestApplic
main] o.s.b.w.embedded.tomcat.TomcatWe
main] o.apache.catalina.core.StandardS
main] o.apache.catalina.core.StandardE
main] o.a.c.c.C.[Tomcat].[localhost].[
main] w.s.c.ServletWebServerApplicatio
main] o.s.b.w.embedded.tomcat.TomcatWe
main] c.example.demo.GitCicdTestApplic
```

Spring boot



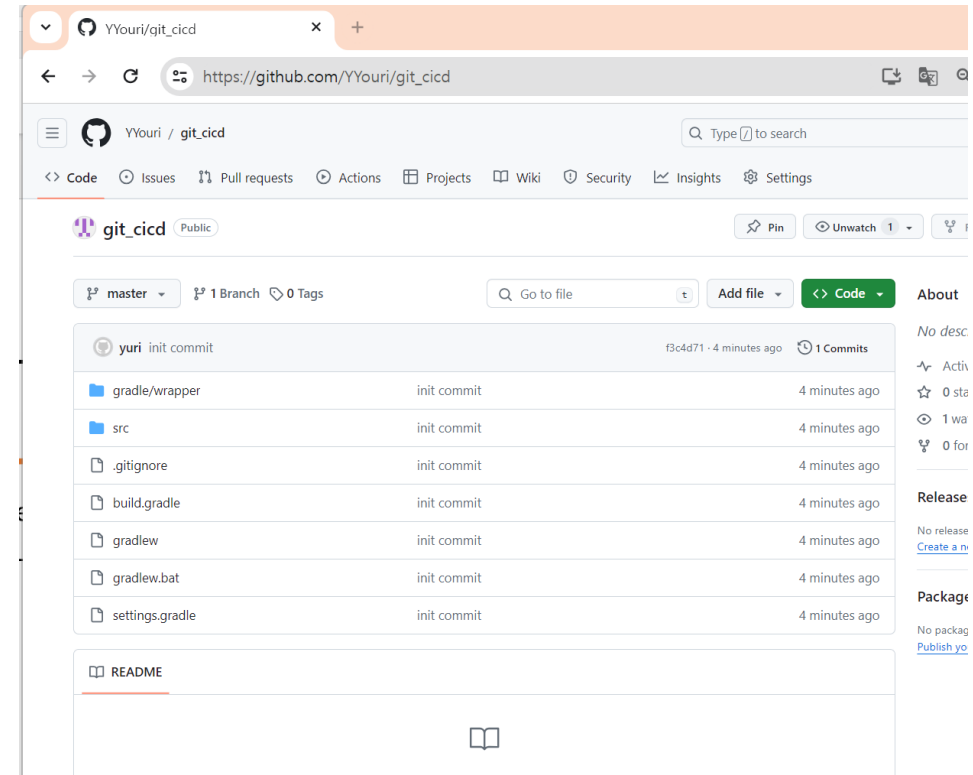
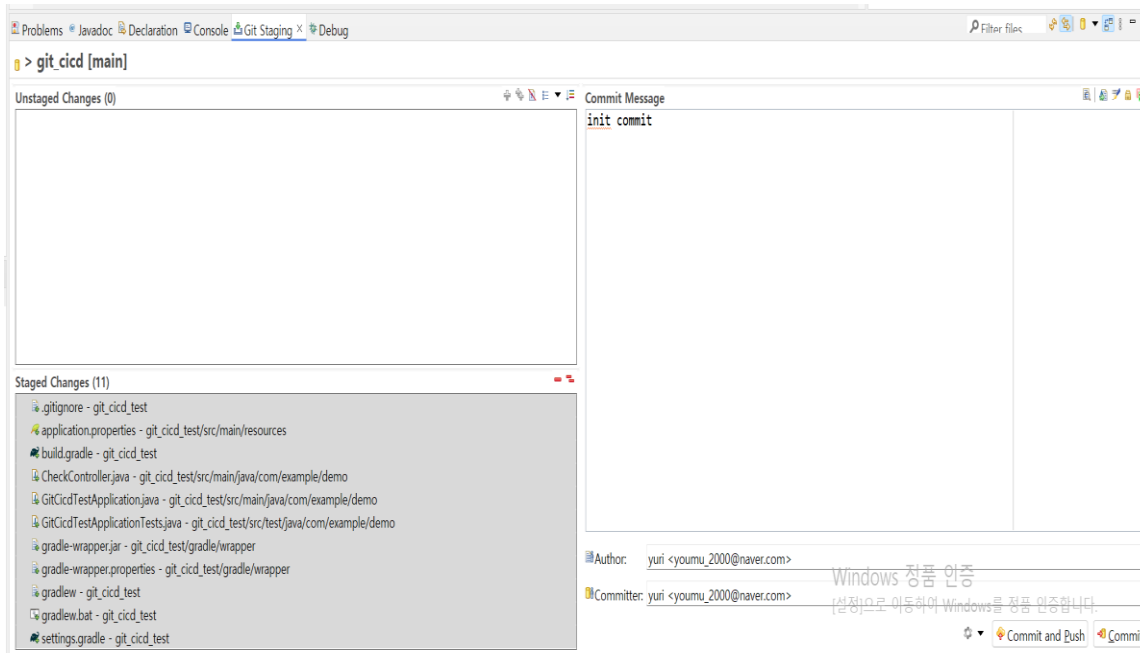
GITHUB 연동

- Package Explorer 이동
- 프로젝트 – team-share project

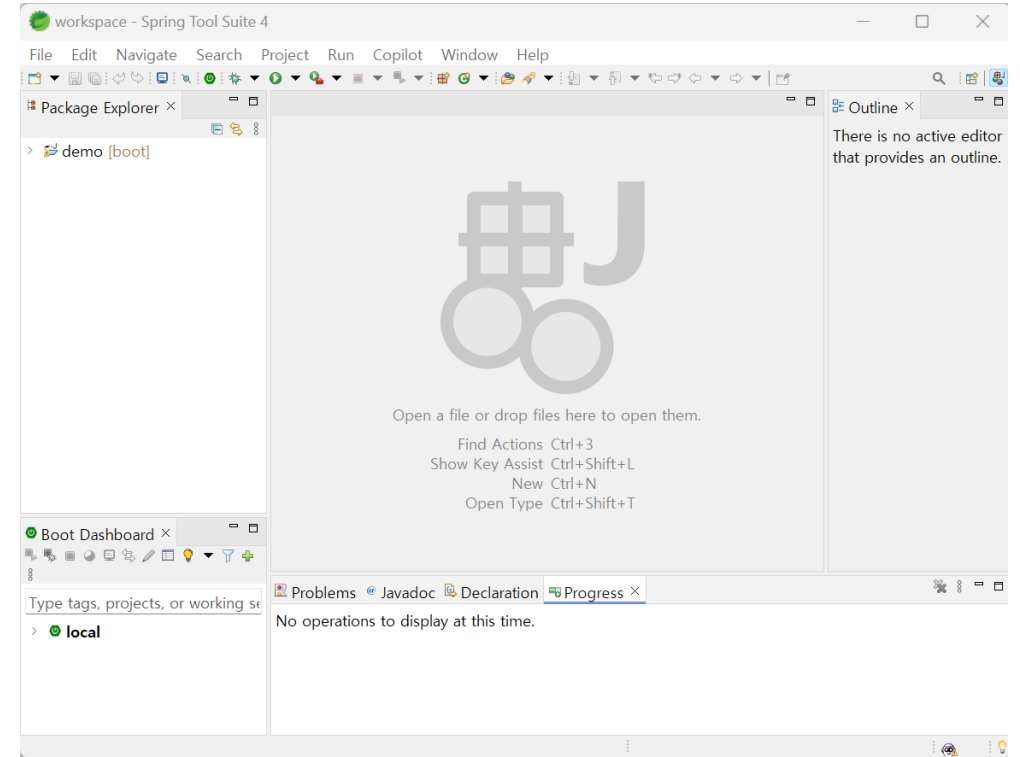
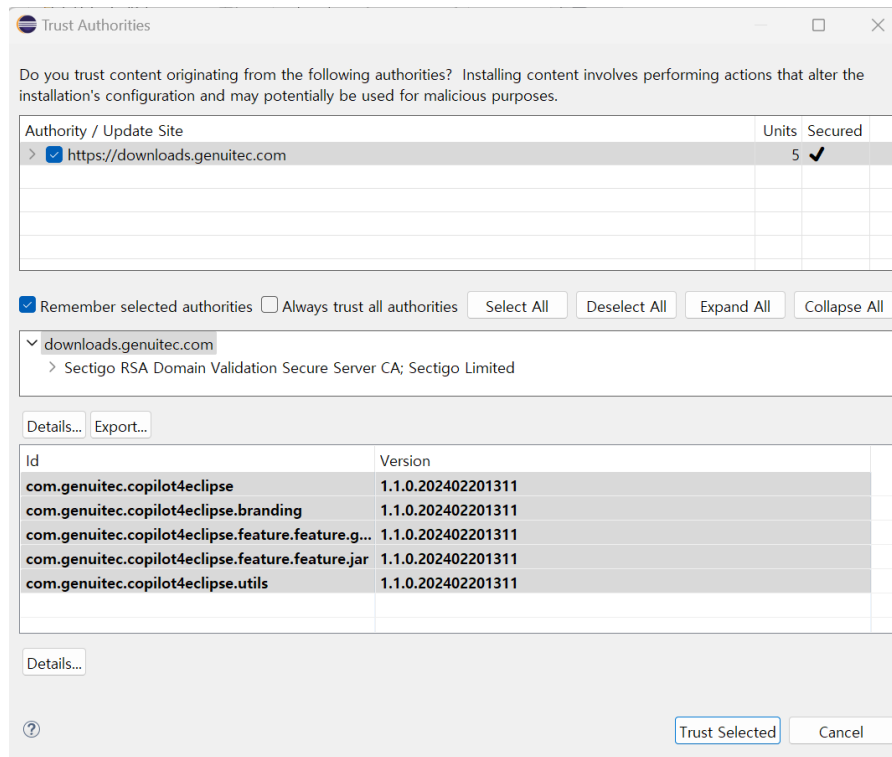
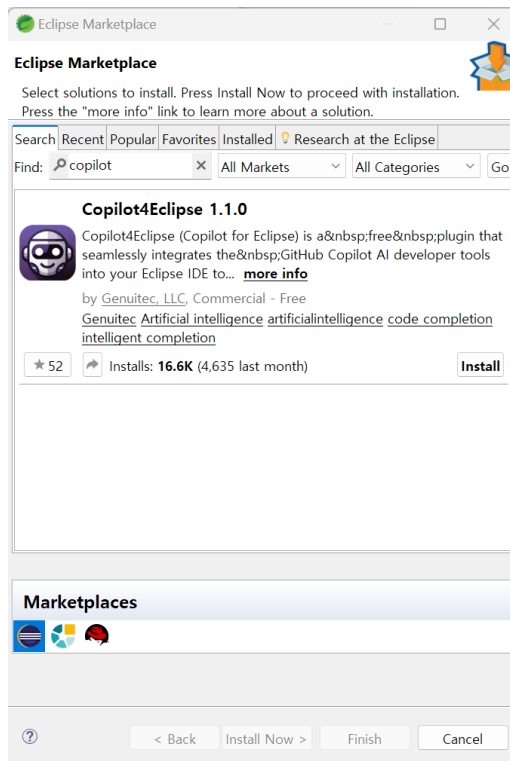


GITHub 연동

- Git Perspective 이동
- Push- branch



코파일럿 설치

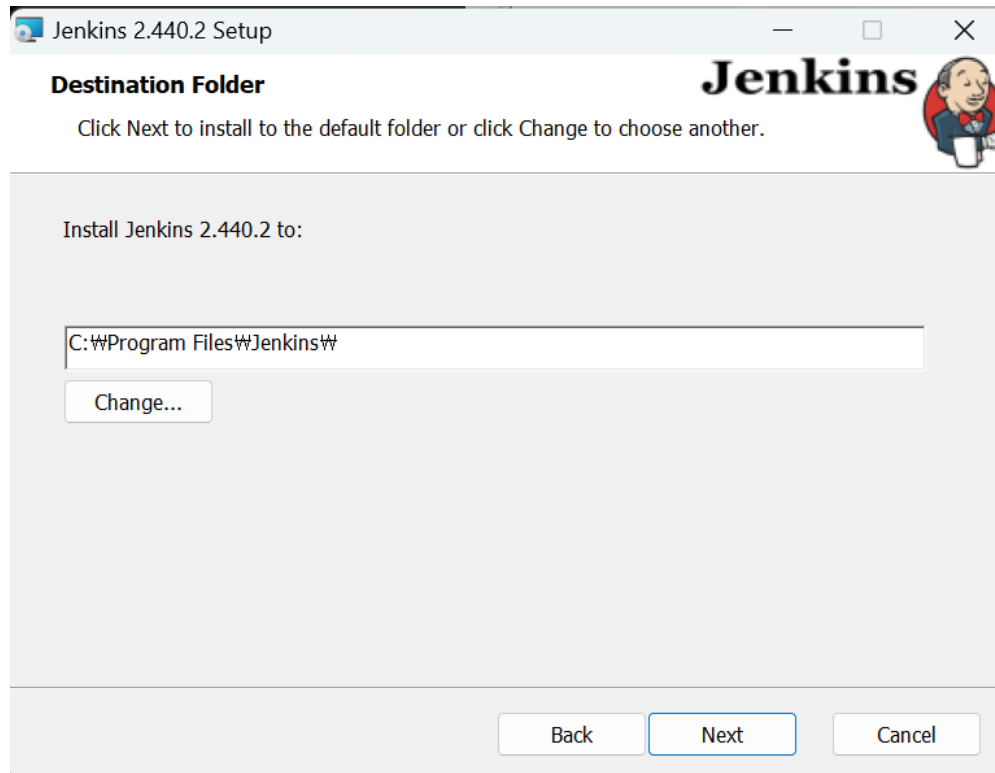


GitHub-WebHooks –Jenkins



Jenkins 설치

- 1. Jenkins 다운로드 : <https://www.jenkins.io/>
- 2. Jenkins 설치



Jenkins 설치

- 2. Jenkins 설치

Jenkins 2.440.2 Setup

Service Logon Credentials

Enter service credentials for the service.

Jenkins 2.440.2 installs and runs as an independent Windows service. To operate in this manner, you must supply the user account credentials for Jenkins 2.440.2 to run successfully.

Logon Type:

☒ Run service as LocalSystem (not recommended)

☐ Run service as local or domain user:

Account:

Password:

Test Credentials

Back Next Cancel


Jenkins 2.440.2 Setup

Port Selection

Choose a port for the service.

Please choose a port.

Port Number (1-65535):

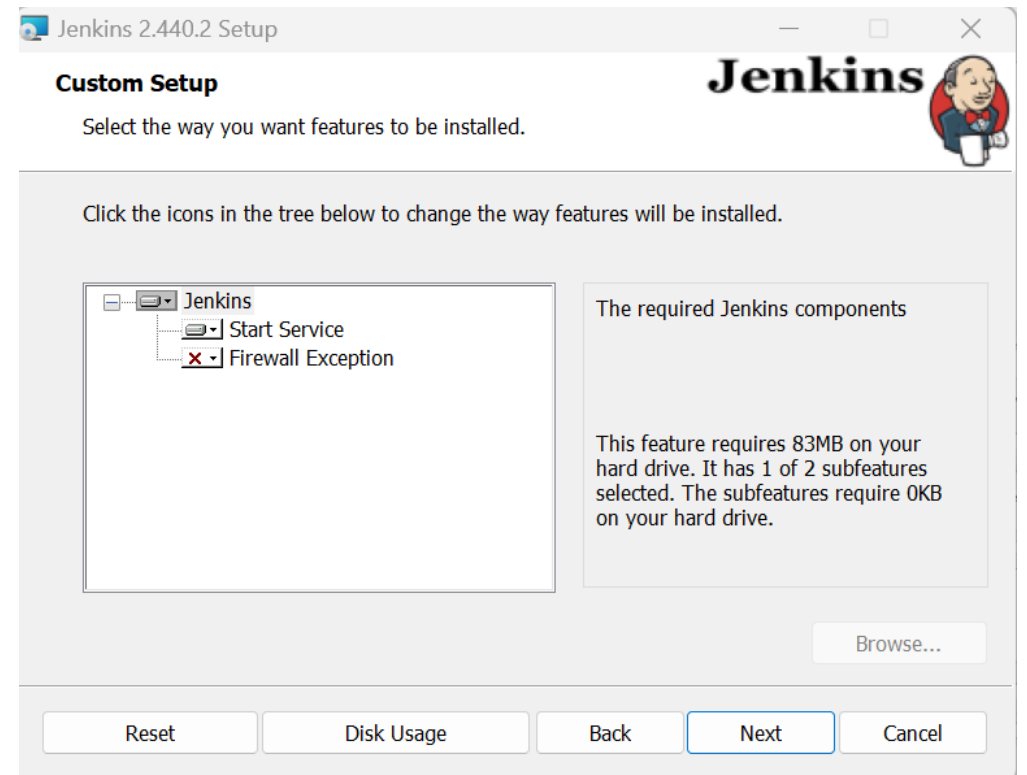
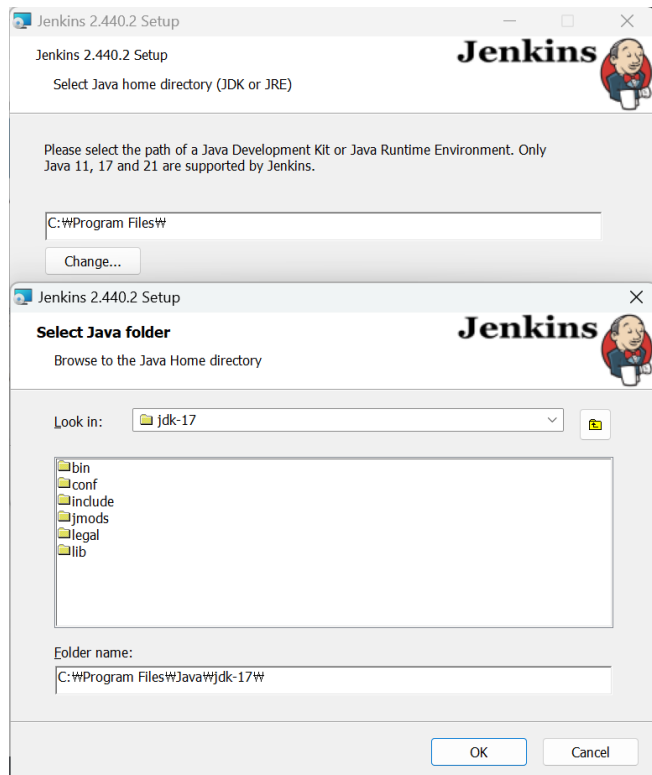
Test Port 

It is recommended that you accept the selected default port.

Back Next Cancel

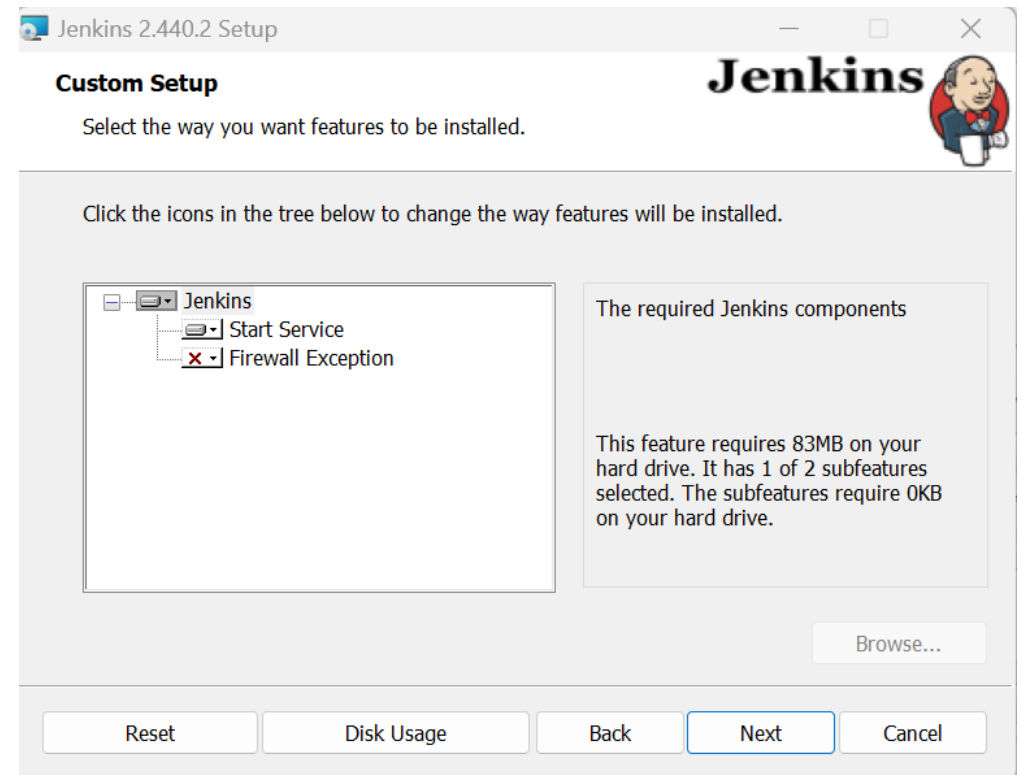
Jenkins 설치

- 2. Jenkins 설치



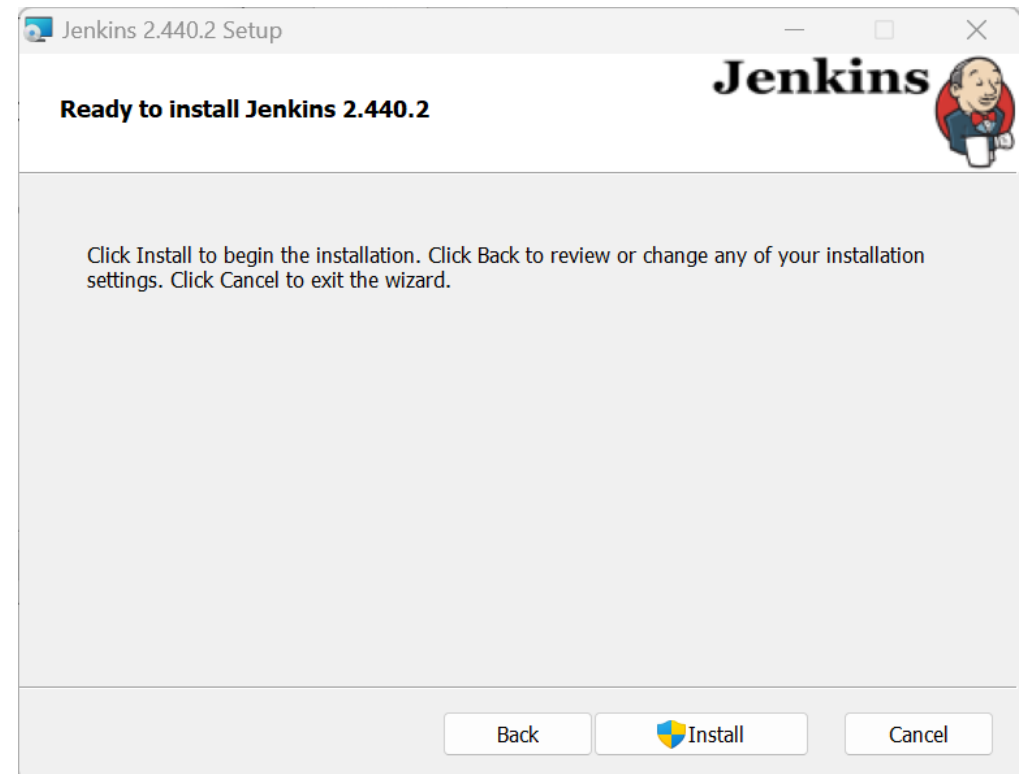
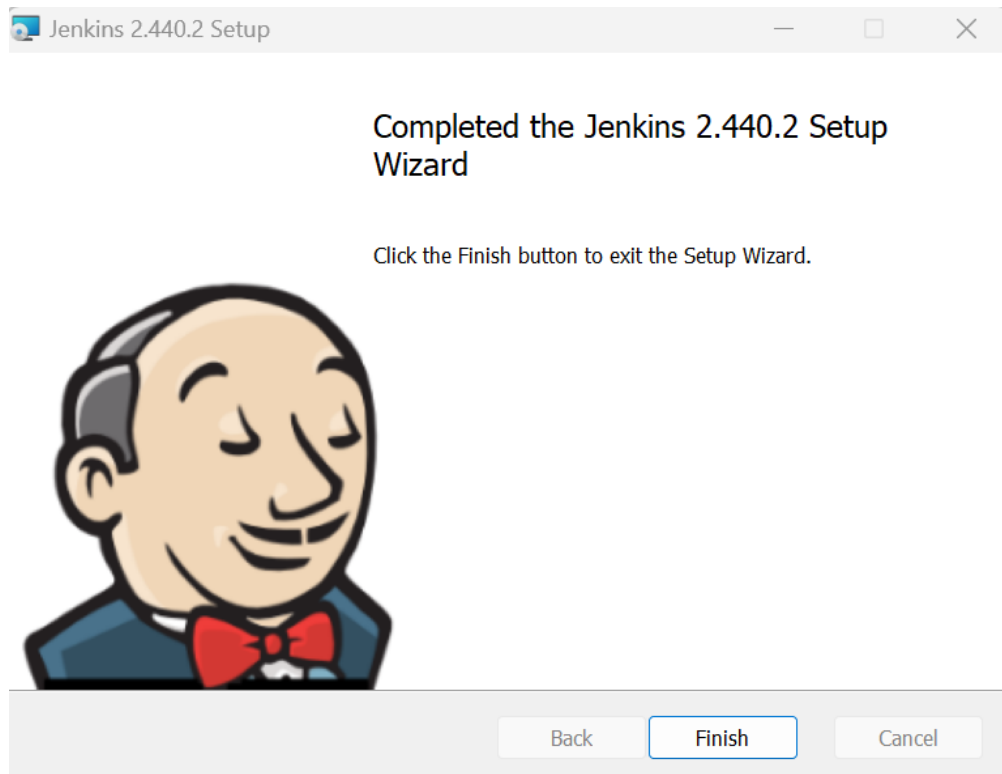
Jenkins 설치

- 2. Jenkins 설치



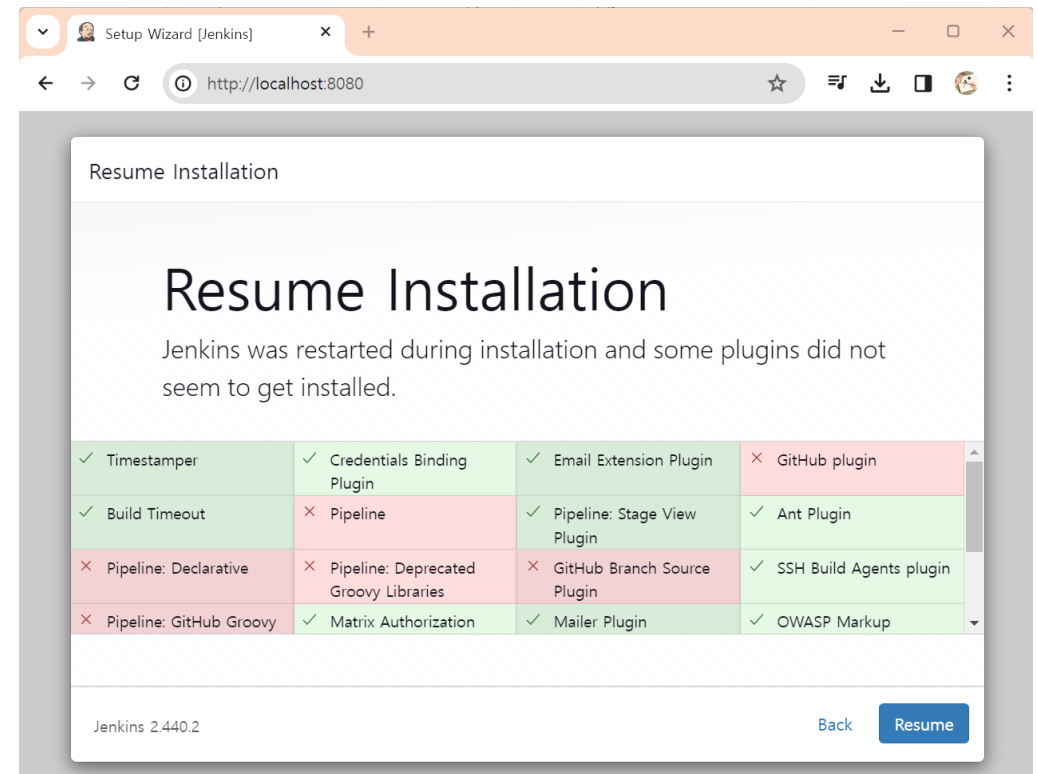
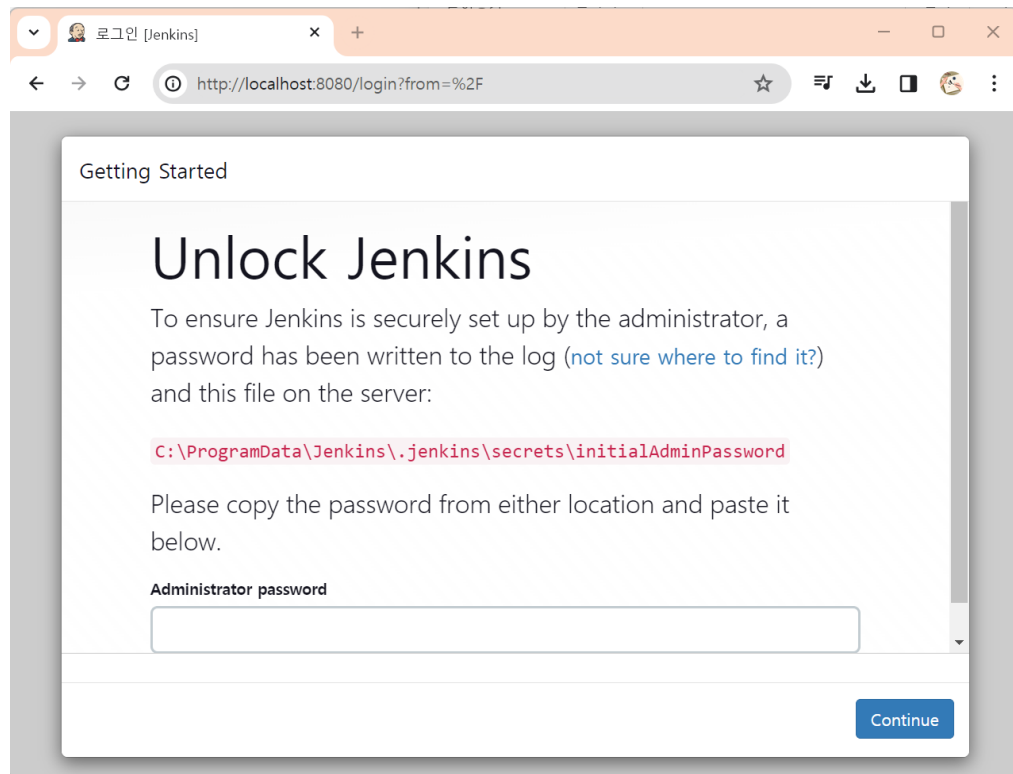
Jenkins 설치

- 2. Jenkins 설치



Jenkins 설치

• 2. Jenkins 설치



Jenkins 설치

• 2. Jenkins 설치

- User.home: C:\ProgramData\Jenkins\Jenkins
 - 플러그인 및 설정 정보 위치
- Install.hom: C:\Program Files\Jenkins
 - 설치 경로로 서비스 시작 및 종료 가능

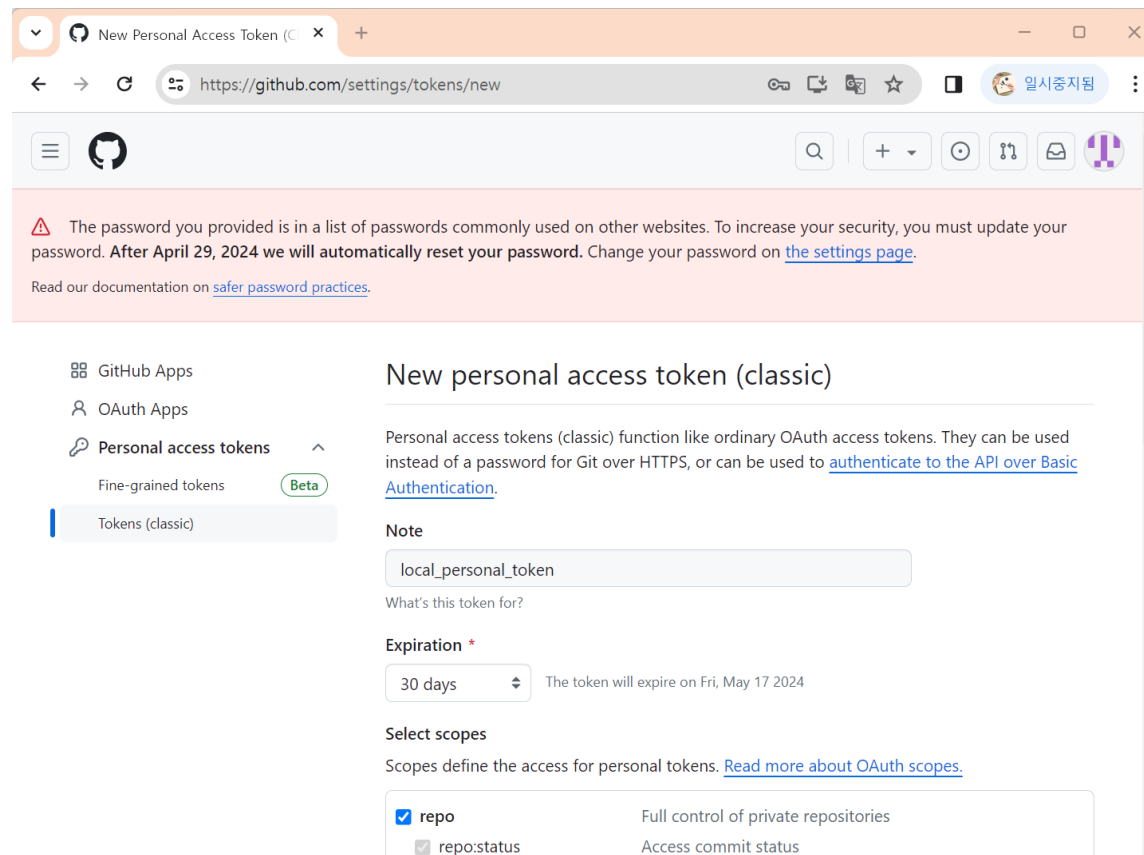
The screenshot shows the 'Getting Started' section of the Jenkins Setup Wizard. The main heading is 'Create First Admin User'. Below this, there are three input fields: '계정명' (Username) with the value 'git', '암호' (Password) with three dots, and '암호 확인' (Confirm Password) with three dots. At the bottom left, it says 'Jenkins 2.440.2'. At the bottom right, there are two buttons: 'Skip and continue as admin' and 'Save and Continue'.

The screenshot shows the 'Getting Started' section of the Jenkins Setup Wizard. The main heading is 'Instance Configuration'. Below this, there is a 'Jenkins URL:' label and a text input field containing 'http://localhost:8080/'. Below the input field, there is a paragraph of text explaining the Jenkins URL. At the bottom left, it says 'Jenkins 2.440.2'. At the bottom right, there are two buttons: 'Not now' and 'Save and Finish'.

GitHub –Jenkins

❖ GitHub 액세스 토큰(access token) 생성

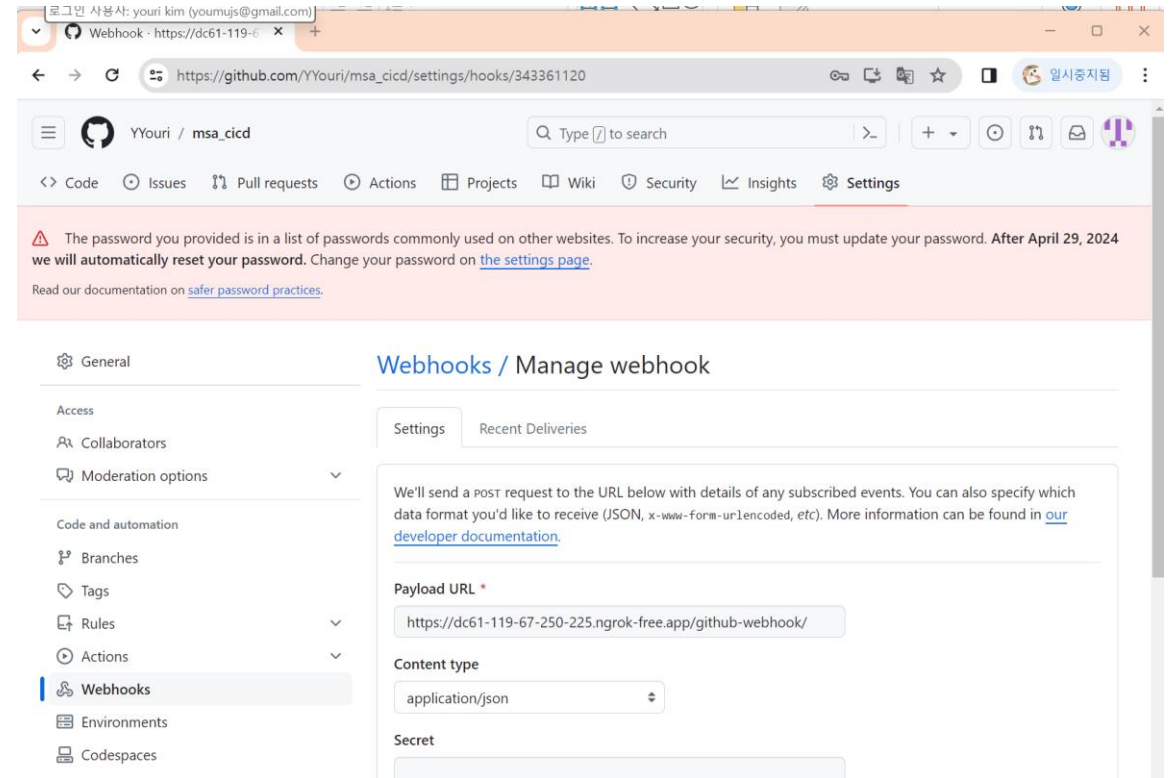
- Settings > Developer settings > Personal access tokens
- 화면에 보이는 Generate new token 버튼 확인
- 만료 시간은 무제한으로 설정
- 해당 토큰으로 접근할 수 있는 scope를 지정
(repo, admin:repo_hook scope 선택)
- 토큰 생성 완료
(토큰 분실 시 다시 생성 해야 하니
다른 곳에 저장 해놓기)



GitHub –Jenkins

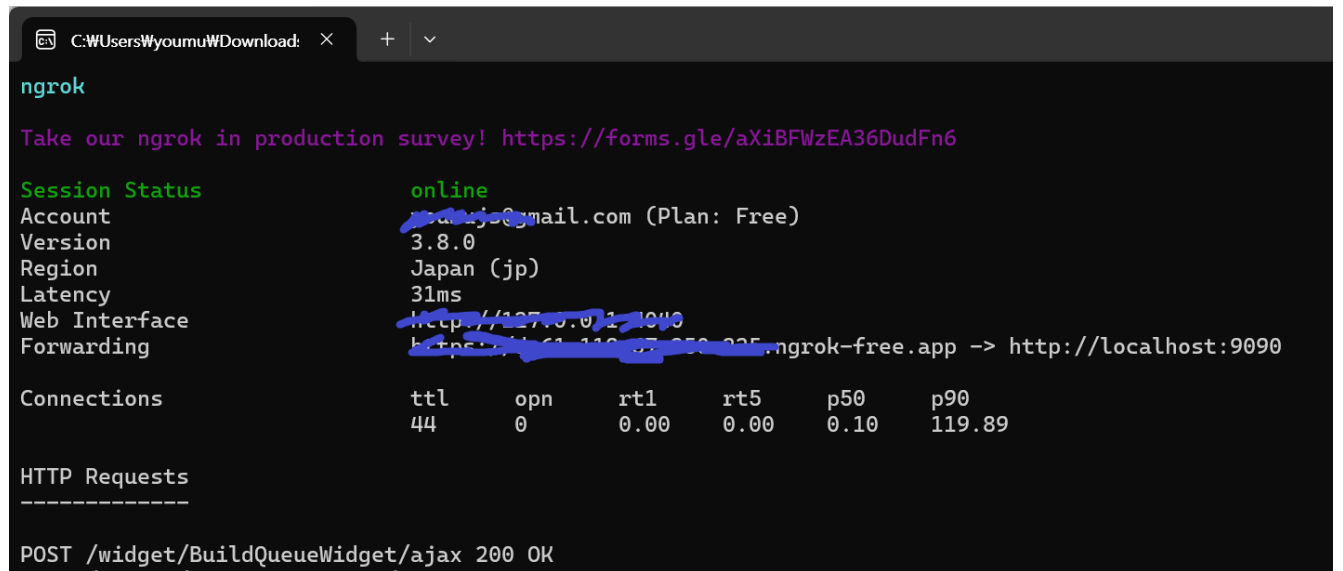
❖ GitHub 개인 레포지토리 webhook 설정

- Settings > Webhooks
- Add webhook 버튼을 클릭
- ngrok 어플리케이션을 통해 외부에서 접근할 수 있는 도메인을 사용



ngrok 설치

- 로컬 컴퓨터의 개발 환경을 인터넷으로 공유해주는 툴로
- Github에서 Webhook을 사용하려면 Jenkins에 DNS 설정이 되어있어서 외부에서도 연결이 가능
- <https://ngrok.com/download>
- ngrok.exe http 9090



```
C:\Users\Wyoumu\Downloads: X + v
ngrok
Take our ngrok in production survey! https://forms.gle/aXiBFWzEA36DudFn6

Session Status      online
Account             [redacted]@gmail.com (Plan: Free)
Version             3.8.0
Region              Japan (jp)
Latency              31ms
Web Interface       http://127.0.0.1:1090
Forwarding           https://[redacted].ngrok-free.app -> http://localhost:9090

Connections      ttl    opn    rt1    rt5    p50    p90
                  44     0      0.00   0.00   0.10   119.89

HTTP Requests
-----
POST /widget/BuildQueueWidget/ajax 200 OK
```

참고) err_ngrok_4018 발생의 경우

- <https://dashboard.ngrok.com/get-started/your-authtoken> 따라 설정

GIThub –Jenkins

- 3. 정보 준비 -인증정보를 기입
- Username = github id
- Password = github access token

New credentials [Jenkins] x +

http://localhost:8080/manage/credentials/store/system/domain/_/newCredentials

Jenkins

검색 (CTRL+K) ?

admin ? 로그아웃

Dashboard > Jenkins 관리 > Credentials > System > Global credentials (unrestricted) >

New credentials

Kind

Username with password

Scope ?

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

Username ?

yyouri

☐ Treat username as secret ?

Password ?

.....

ID ?

personal_token

Create

CI/CD 구축

- 3. 설정(소스트리 설치 경우 git 경로)

Dashboard > Jenkins 관리 > Tools

≡ Git

Name

Default

Path to Git executable ?

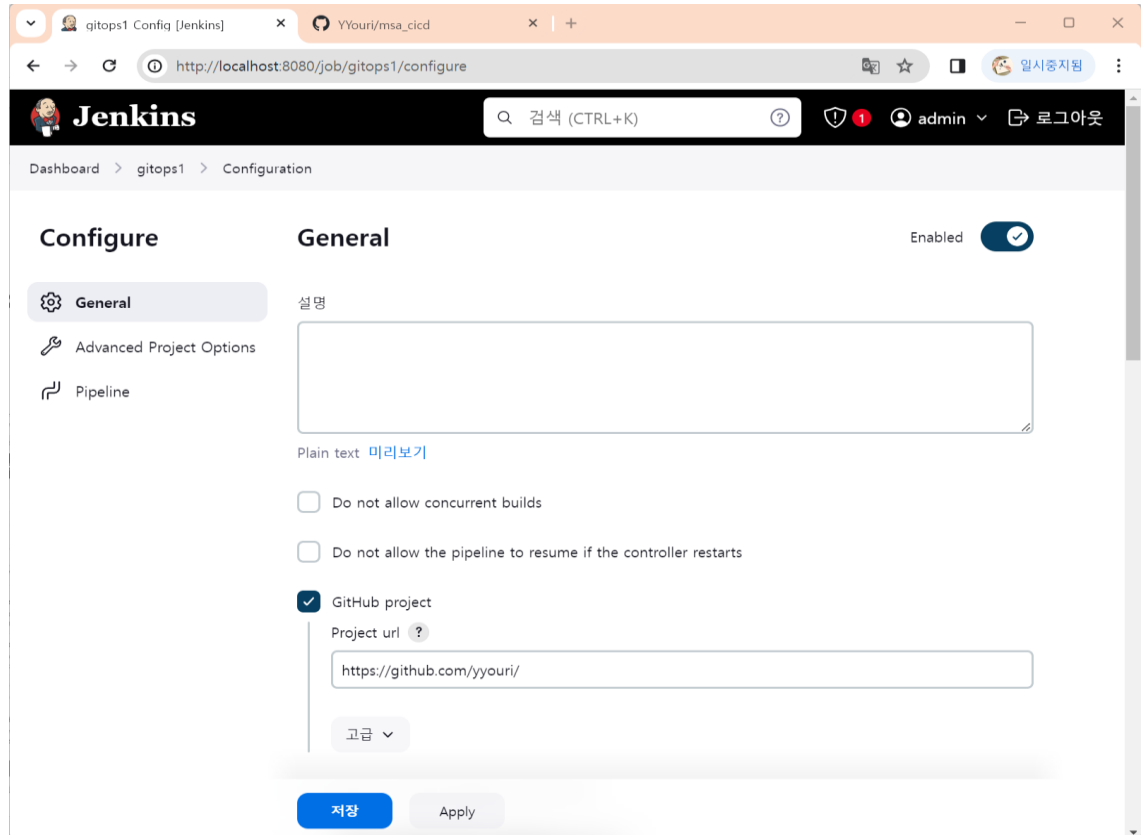
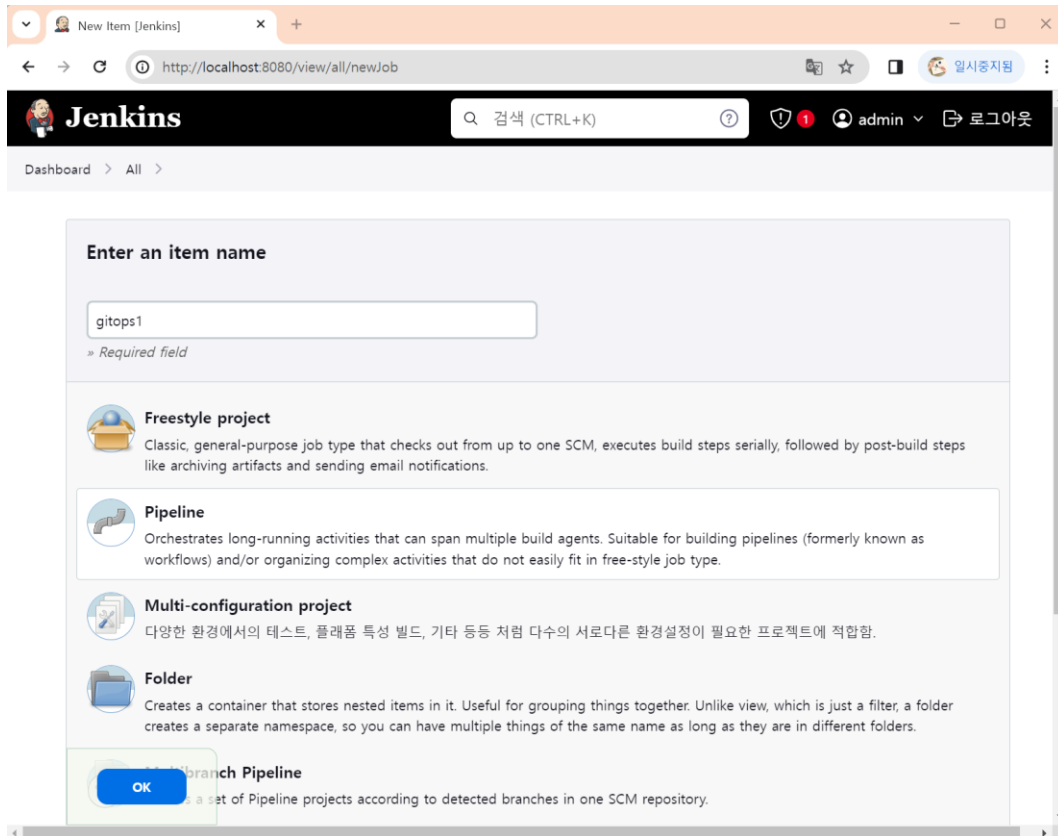
C:\Users\Wyoumu\AppData\Local\Atlassian\SourceTree\git_local\bin\git.exe

☐

Install automatically ?

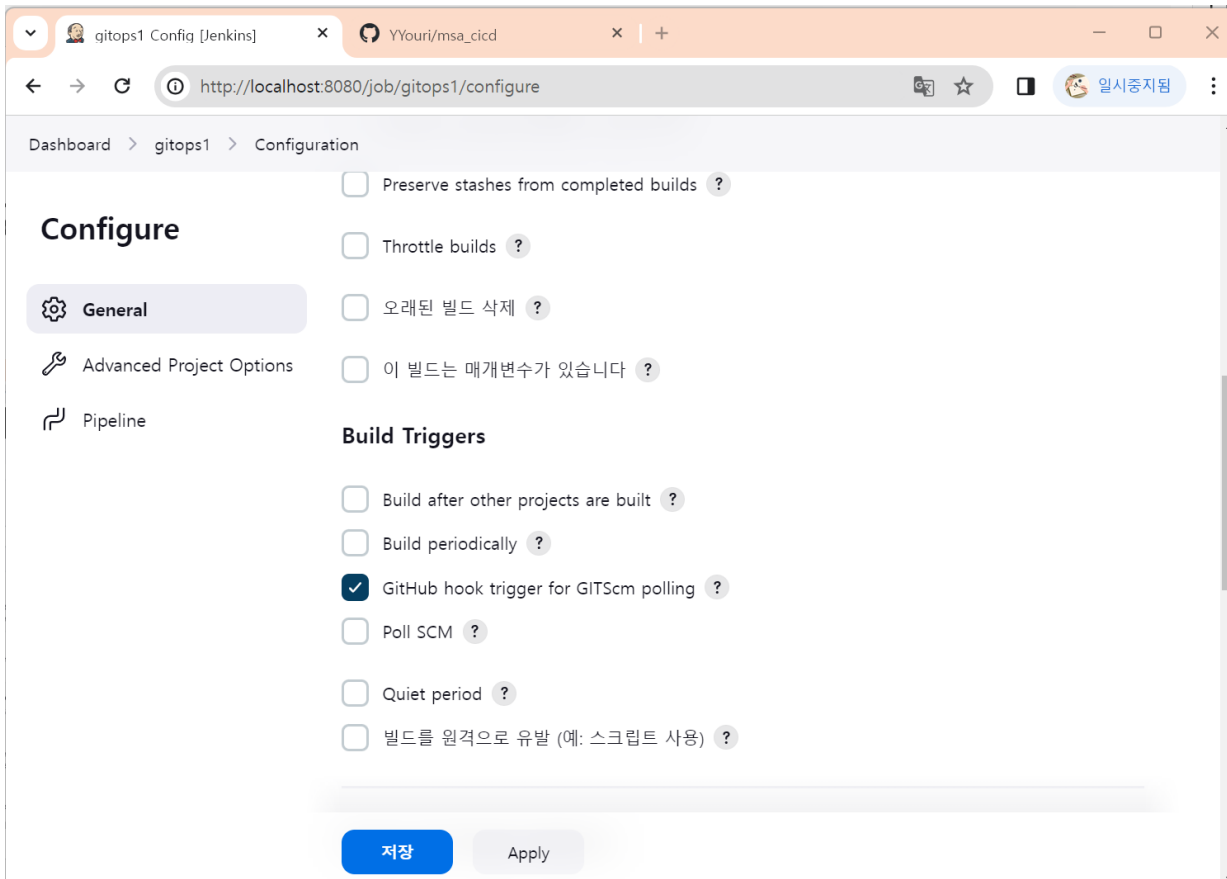
CI/CD 구축

• 4. 파이프라인 만들기



CI/CD 구축

• 4. 파이프라인 만들기



Dashboard > gitops1 > Configuration

Configure

- General (selected)
- Advanced Project Options
- Pipeline

☐ Preserve stashes from completed builds ?

☐ Throttle builds ?

☐ 오래된 빌드 삭제 ?

☐ 이 빌드는 매개변수가 있습니다 ?

Build Triggers

☐ Build after other projects are built ?

☐ Build periodically ?

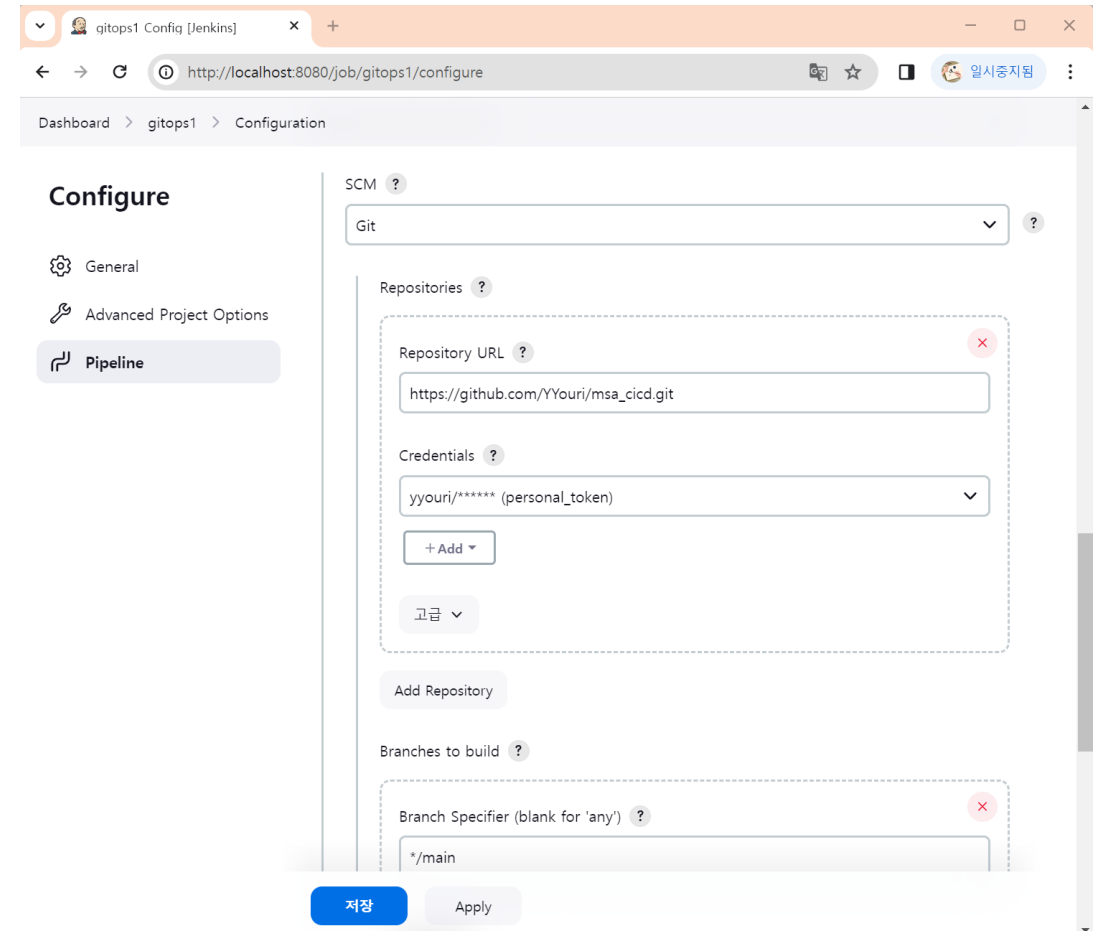
☒ GitHub hook trigger for GITScm polling ?

☐ Poll SCM ?

☐ Quiet period ?

☐ 빌드를 원격으로 유발 (예: 스크립트 사용) ?

저장 Apply



Dashboard > gitops1 > Configuration

Configure

- General
- Advanced Project Options
- Pipeline (selected)

SCM ?

Git

Repositories ?

Repository URL ?

https://github.com/YYouri/msa_cicd.git

Credentials ?

yyouri/***** (personal_token)

+ Add

고급

Add Repository

Branches to build ?

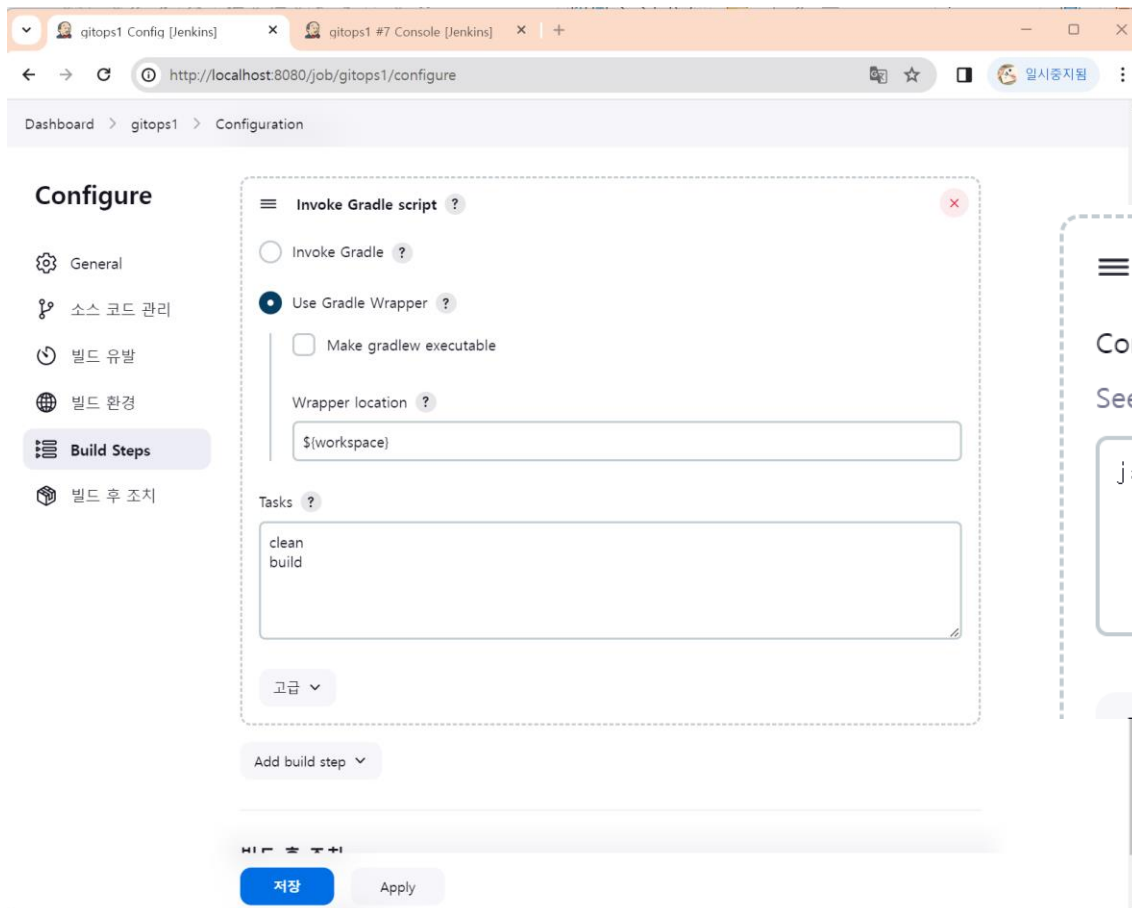
Branch Specifier (blank for 'any') ?

*/main

저장 Apply

CI/CD 구축

• 4. 파이프라인 만들기



Execute Windows batch command ?

Command

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

```
java -jar "%WORKSPACE%\build\libs\git_flow-0.0.1-SNAPSHOT.jar"
```

CI/CD 구축

- 4. 파이프라인 만들기

All +

S	W	Name ↓	최근 성공	최근 실패
		gitops1	13 min #24	5 min 27 sec #26

아이콘: S M L

