

표준프레임워크
CI 및 Nexus 환경 구축 가이드
Version 4.2

Index

1. 표준프레임워크 EGOVCI 패키지 설치	3
1.1 개요.....	3
1.2 EGOVCI 압축풀기.....	3
1.3 EGOVCI시스템 구성.....	3
1.4 CI 시스템 구동 (START/STOP)	4
2. NEXUS 설정 정보.....	6
2.1 NEXUS 서버구동(START/STOP)	6
2.2 NEXUS 저장소 설정	7
2.3 NEXUS 저장소에 라이브러리 추가.....	7
2.4 NEXUS 실행 및 추가된 라이브러리 확인.	8
2.5 NEXUS저장소를 프로젝트에 반영.....	8
2.6 NEXUS 저장소에 3 RD PARTY 라이브러리 추가.....	9
3. CI (SVN & JENKINS) 설정.....	11
3.1 SUBVERSION 설치 (선택사항)	11
3.2 SVN REPOSITORIES 설정 및 사용자 등록.	12
3.2.1 저장소 설정 및 등록	12
3.2.2 사용자 등록	13
3.2.3 SVN 서버 시작/종료.....	14
3.2.4 Eclipse 프로젝트를 SVN서버에 등록.	14
3.3 JENKINS 설정.....	16
3.3.1 Jenkins 기본 시스템 설정	17
3.3.2 새 작업 생성	20
3.3.3 작업 Bulid.....	24

1. 표준프레임워크 eGovCI 패키지 설치

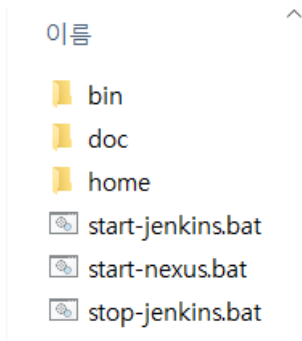
1.1 개요

본 가이드는 전자정부 표준프레임워크 센터에서 제공하는 eGovCI 구성 및 설치 가이드이며, 아래와 같은 구성으로 이루어져 있다.

- eGovCI 패키지 (윈도우용 exe 압축파일 또는 리눅스용 tar.gz 압축파일)
- Nexus repository 구성 정보.
- CI 관련 구성 정보 (SVN 포함, 리눅스용 패키지는 미포함 - 별도 설치필요).
- 기본 접속 설정 URL
- JENKINS : http://<ip_address>:8080/jenkins
- NEXUS : http://<ip_address>:8081/nexus

1.2 eGovCI 압축풀기

전자정부 표준프레임워크에서 제공하는 eGovCI ("eGovCI-X.X.X_XXbit") 파일을 "C:\\"에 압축을 푼다. 압축을 푼 구조는 아래와 같다. (리눅스의 경우 /home/<사용자> 디렉토리 권장)



1.3 eGovCI 시스템 구성

eGovCI 패키지는 별도의 모듈이 필요하지 않도록, 아래와 같은 구성으로 제공하고 있다. 주요 모듈로는 JAVA JDK, Tomcat configurations, Maven 등으로 구성되어 있다.

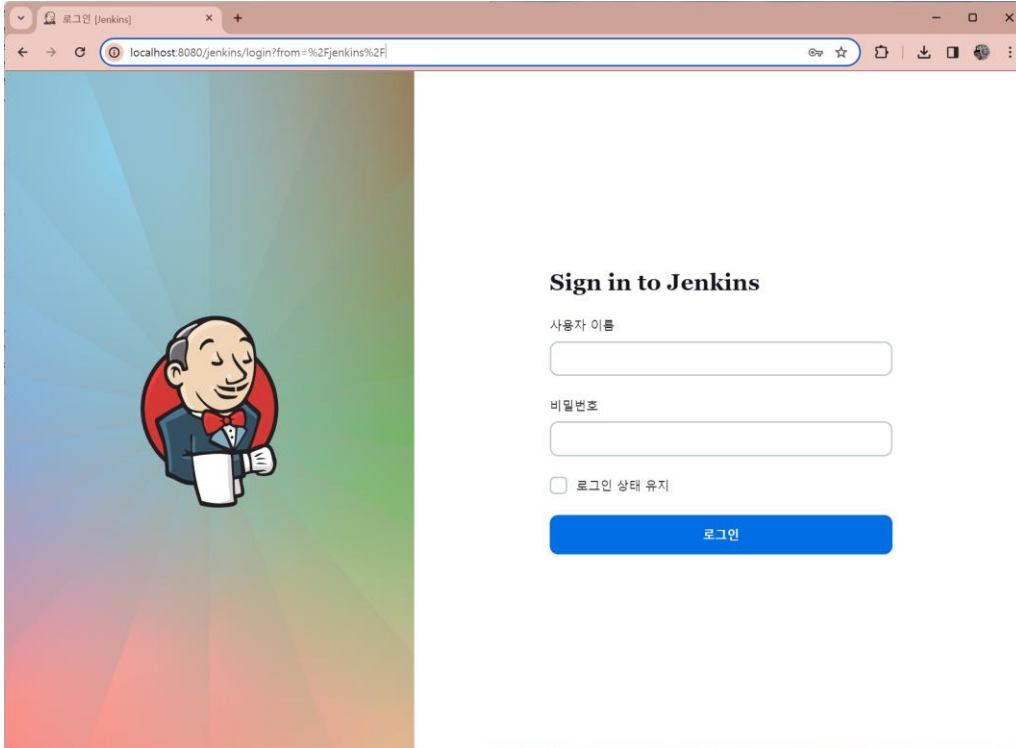
- Ant-1.10.14 (under /bin dictory)
- Maven-3.9.6 (under /bin dictory)
- Tomcat-9.0.85 (under /bin dictory)
- OpenJDK1.8 / 17 (under /bin dictory)
- VisualSVN-Server-5.3.2.msi (under /bin dictory) - 별도 설치형 파일 제공 (리눅스 제외)

구동 후, 아래와 같은 주소로 CI 웹 페이지를 확인 할 수 있다.

(<http://localhost:8080/jenkins/>)

계정 ID : admin ,

계정 비밀번호 : <eGovCI_경로>/home/jenkins/secrets/initialAdminPassword 파일 확인



** CI 서버 기본 포트는 '8080'으로 설정되어 있으며, 아래와 같이 별도 포트도 변경하여 사용가능 하다. (<eGovCI_home>\bin\apache-tomcat-jenkins\conf\server.xml)

```

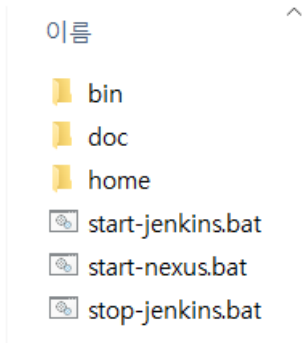
server - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
<!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Documentation at :
Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
Java AJP Connector: /docs/config/ajp.html
APR (HTTP/AJP) Connector: /docs/apr.html
Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
-->
<Connector port="8080" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" />
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
port="8080" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" />
-->
<!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443
This connector uses the NIO implementation that requires the JSSE
-->
server - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
    
```

2. Nexus 설정 정보

2.1 Nexus 서버구동(Start/Stop)

CI 시스템 구성 이후, “start-nexus.bat”을 사용하여 Nexus 서버 구동 할 수 있으며, 실행된 콘솔창을 닫으면 Nexus 서버를 멈출 수 있다. (별도의 stop nexus 파일 없음)

(! 반드시 관리자 권한으로 실행)



구동 이후, 아래와 같은 콘솔 창으로 nexus 을 시작을 확인 할 수 있다.

기본 접속 URL : <http://localhost:8081/nexus/>

```

Sonatype Nexus
Using CI_HOME: "C:\\eGovCI-4.2.0_64bit\"
Using JAVA_HOME: "C:\\eGovCI-4.2.0_64bit\\bin\\jdk8u242-b08"
Using PATH: "C:\\eGovCI-4.2.0_64bit\\bin\\jdk8u242-b08\\bin;C:\\WINDOWS\\system32;C:\\WINDOWS;C:\\WINDOWS\\System32\\Wbem;C:\\WI
NDOWS\\System32\\WindowsPowerShell\\v1.0\\;C:\\WINDOWS\\System32\\OpenSSH\\;C:\\Program Files\\Bandizip\\;C:\\Program Files\\nodejs\\;
C:\\Program Files\\Docker\\Docker\\resources\\bin;E:\\Devs\\Java\\jdk-11.0.5\\bin;C:\\Program Files\\TortoiseGit\\bin;E:\\PortableGi
t\\cmd;C:\\Program Files\\Git\\cmd;E:\\Devs\\maven\\apache-maven-3.9.6\\bin;E:\\Devs\\Java\\jdk-11.0.5\\bin;C:\\Users\\jung3\\AppData\\L
ocal\\Microsoft\\WindowsApps;C:\\Users\\jung3\\AppData\\Roaming\\npm;C:\\Users\\jung3\\AppData\\Local\\Programs\\Microsoft VS Code\\bi
n;"
Using NEXUS_HOME: "C:\\eGovCI-4.2.0_64bit\\bin\\nexus-2.15.1-02-bundle\\nexus-2.15.1-02"
-----
CI_HOME : "C:\\eGovCI-4.2.0_64bit\"
-----
eGovCI Nexus server start
-----
wrapper | --> Wrapper Started as Console
wrapper | Launching a JVM...
jvm 1 | Wrapper (Version 3.2.3) http://wrapper.tanukisoftware.org
jvm 1 | Copyright 1999-2006 Tanuki Software, Inc. All Rights Reserved.
    
```

포트(Port)변경을 아래와 같은 경로에서 nexus.properties 파일 내에 application-port=8081 을 변경할 수 있다

파일 경로 : <eGovCI_home>\ bin\nexus-2.15.1-02-bundle\nexus-2.15.1-02\conf

```

# Jetty section
application-port=8081
application-host=0.0.0.0
nexus-webapp=${bundleBasedir}/nexus
nexus-webapp-context-path=/nexus
    
```

2.2 Nexus 저장소 설정

Nexus 의 저장소는 기본으로 <eGovCI_home>\home\nexus 로 설정되어 있으며, 아래와 같이 별도 설정이 가능하다.

```

start-nexus - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
> 로컬 디스크 (C:) > eGovCI > home > nexus

@echo off
rem ----- CI Nexus SETTING start -----
set CL_HOME=%~d0%~p0
set JAVA_HOME=%CL_HOME%\bin\jdk1.8.0_162
set JRE_HOME=%CL_HOME%\bin\jdk1.8.0_162\jre
set CATALINA_HOME=%CL_HOME%\bin\apache-tomcat-nexus
set PLEXUS_NEXUS_WORK=%CL_HOME%\home\nexus
echo Using CL_HOME: "%CL_HOME%"
echo Using JAVA_HOME: "%JAVA_HOME%"
echo Using CATALINA_HOME: "%CATALINA_HOME%"
rem ----- CI Nexus SETTING end -----
echo -----
echo CL_HOME : "%CL_HOME%"
echo -----
echo eGovCI Nexus server start
echo -----
set JAVA_OPTS=-Xmx768m
%CATALINA_HOME%\bin\catalina.bat start
echo done.
    
```

2.3 Nexus 저장소에 라이브러리 추가.

Nexus 저장소에 라이브러리 추가 위치

<eGovCI_home>\home\nexus\storage

- Storage 는 Nexus 에 등록된 central repository 들을 포함하고 있으며, 각 repository 는 등록된 URL 기반으로 웹 상의 해당 repository 에 접속하여 라이브러리를 내려 받아 관리함

또한, Nexus storage 에 필요한 라이브러리를 아래와 같이 한 번에 추가 가능하다.

- Target 저장소 : <eGovCI_home>/home/nexus/storage/releases
- 직접 또는 FTP 환경을 설정하여 Target 저장소에 위치시킨다.

```

> 로컬 디스크 (C:) > eGovCI > home > nexus > storage > releases

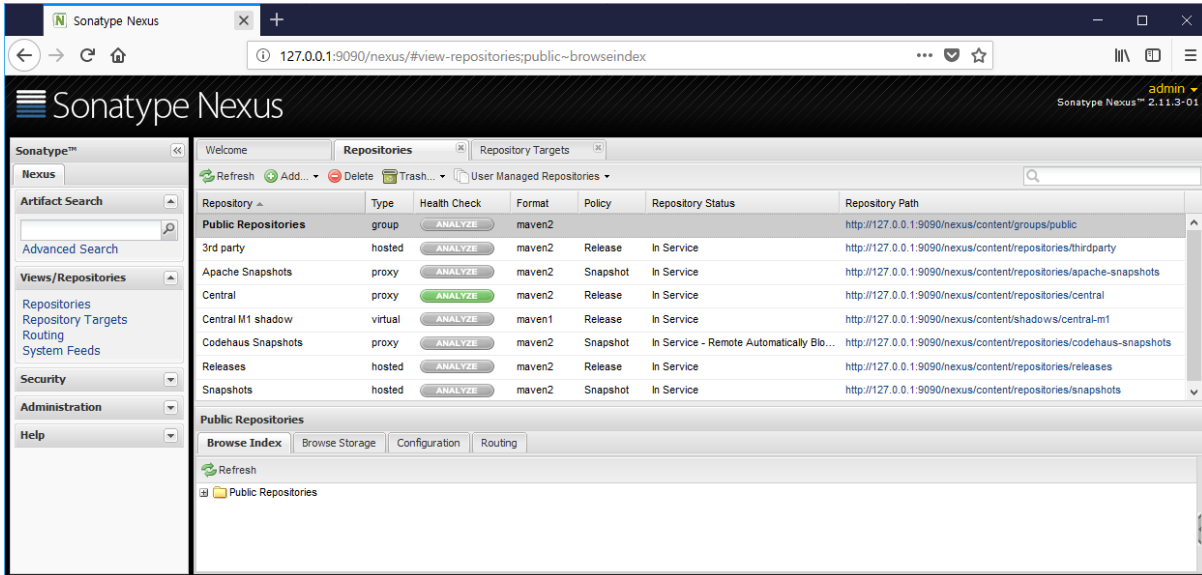
이름
. meta
archetype-catalog
    
```

2.4 Nexus 실행 및 추가된 라이브러리 확인.

"start-nexus.bat" 실행하여, Nexus 를 실행가능하며, 브라우저에서 아래 주소로 Nexus 을 확인 가능하다.

http://localhost:8081/nexus

Nexus 를 접속 후 "repositories" 선택하고 "public repositories" 또는 release 저장소에서 추가한 라이브러리 목록을 확인할 수 있다.



2.5 Nexus 저장소를 프로젝트에 반영.

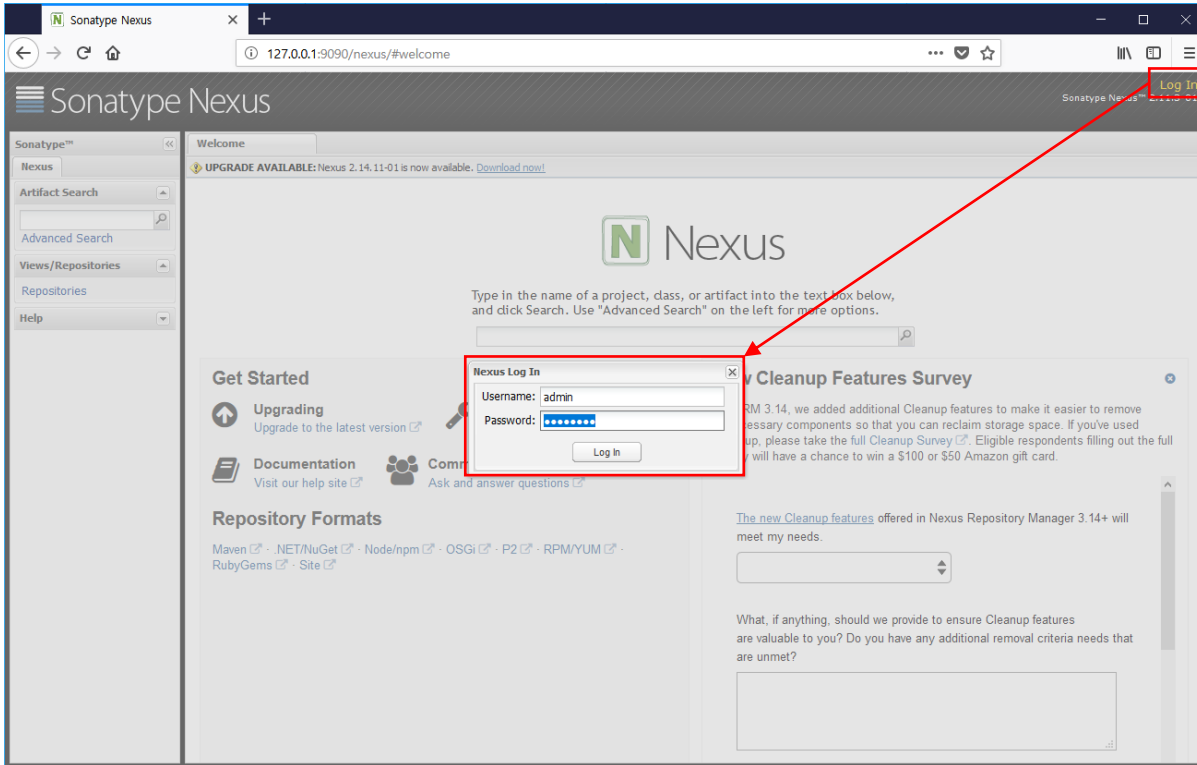
프로젝트의 pom.xml file 을 열고 아래와 같이 <repositories> 변경하여 반영할 수 있다.

```
<repositories>
  <repository>
    <id>public</id>
    <url>http://localhost:8081/nexus/content/groups/public</url>
    <releases>
      <enabled>true</enabled>
    </releases>
    <snapshots>
      <enabled>true</enabled>
    </snapshots>
  </repository>
  <repository>
    <id>projectRepository</id>
    <name>Project Repository</name>
    <url>file://${project.basedir}/src/main/webapp/WEB-INF/lib</url>
  </repository>
</repositories>
```


2.6 Nexus 저장소에 3rd Party 라이브러리 추가.

Nexus 저장소에 별도로 3rd Party 라이브러리를 별개로 등록 가능하다.

1. 관리자로 로그인 : admin / admin123



2. 저장소 목록에서 “3rd party” 저장소를 선택 한다.

3. 이후, Artifact Upload -> GAV Definition -> GAV Parameters 를 차례대로 선택한다.

4. 추가할 라이브러리 Group, Artifact, Version 정보를 입력한다.

5. 정보 입력후 "Select Artifact(s) to Upload" 통해 라이브러리를 추가 한다.

6. 추가한 정보를 확인 후 "Add artifact" 통해 적용한다.

7. “Upload Artifact(s)” 를 통해 업로드한다.

The screenshot shows the Sonatype Nexus web interface. The browser address bar indicates the URL: `127.0.0.1:9090/nexus/#view-repositories;thirdparty~uploadPanel`. The page title is "Sonatype Nexus" and the user is logged in as "admin".

The main content area displays a table of repositories under the "3rd party" view. The table has columns for Repository, Type, Health Check, Format, Policy, Repository Status, and Repository Path. The "3rd party" repository is highlighted with a red box.

Repository	Type	Health Check	Format	Policy	Repository Status	Repository Path
Public Repositories	group	ANALYZE	maven2			<code>http://127.0.0.1:9090/nexus/content/groups/public</code>
3rd party	hosted	ANALYZE	maven2	Release	In Service	<code>http://127.0.0.1:9090/nexus/content/repositories/thirdparty</code>
Apache Snapshots	proxy	ANALYZE	maven2	Snapshot	In Service	<code>http://127.0.0.1:9090/nexus/content/repositories/apache-snapshots</code>
Central	proxy	ANALYZE	maven2	Release	In Service	<code>http://127.0.0.1:9090/nexus/content/repositories/central</code>
Central M1 shadow	virtual	ANALYZE	maven1	Release	In Service	<code>http://127.0.0.1:9090/nexus/content/shadows/central-m1</code>
Codehaus Snapshots	proxy	ANALYZE	maven2	Snapshot	In Service - Remote Automatically Blo...	<code>http://127.0.0.1:9090/nexus/content/repositories/codehaus-snapshots</code>
Releases	hosted	ANALYZE	maven2	Release	In Service	<code>http://127.0.0.1:9090/nexus/content/repositories/releases</code>
Snapshots	hosted	ANALYZE	maven2	Snapshot	In Service	<code>http://127.0.0.1:9090/nexus/content/repositories/snapshots</code>

Below the table, the "Artifact Upload" panel is visible. It includes a "Select GAV Definition Source" dropdown menu with "GAV Parameter" selected. There are input fields for "Group", "Artifact", "Version", and "Packaging". Below these, there is a section for "Select Artifact(s) for Upload" with a "Select Artifact(s) to Upload" button and input fields for "Filename", "Classifier", and "Extension". An "Add Artifact" button is also present. At the bottom, there are "Upload Artifact(s)" and "Reset" buttons.

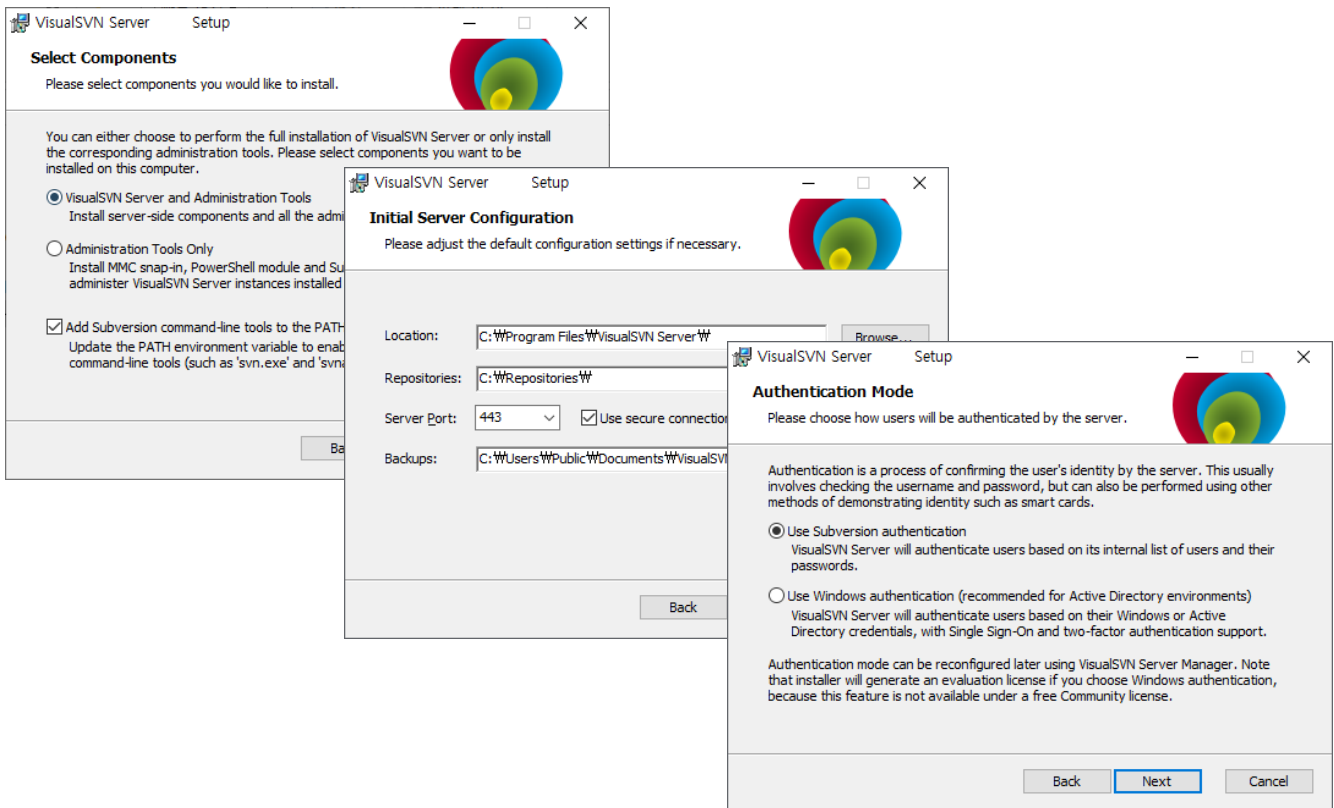
3. CI (SVN & JENKINS) 설정

3.1 Subversion 설치 (선택사항)

CI 설정에 앞서 소스 저장소(Subversion)을 설치한다. eGovCI에는 VisualSVN의 설치파일을 별도로 제공하고 있으며, 필요 시 설치하여 사용한다. (별도로 SVN을 구성하여 사용하여도 무방하다.)

eGovCI의 VisualSVN-server-XXX.msi를 설치한다.

- 📁 apache-ant-1.10.14
- 📁 apache-maven-3.9.6
- 📁 apache-tomcat-9.0.85-jenkins
- 📁 jdk8u242-b08
- 📁 jdk-17.0.10+7
- 📁 nexus-2.15.1-02-bundle
- 📁 VisualSVN-Server-5.3.2-x64



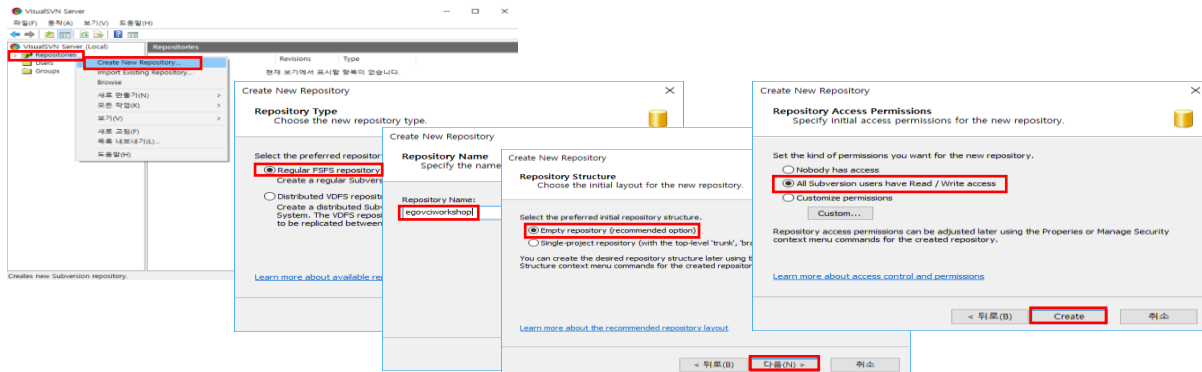
3.2 SVN Repositories 설정 및 사용자 등록.

SVN 저장소 설정 및 사용자 등록 관련 사항을 설정한다.

3.2.1 저장소 설정 및 등록

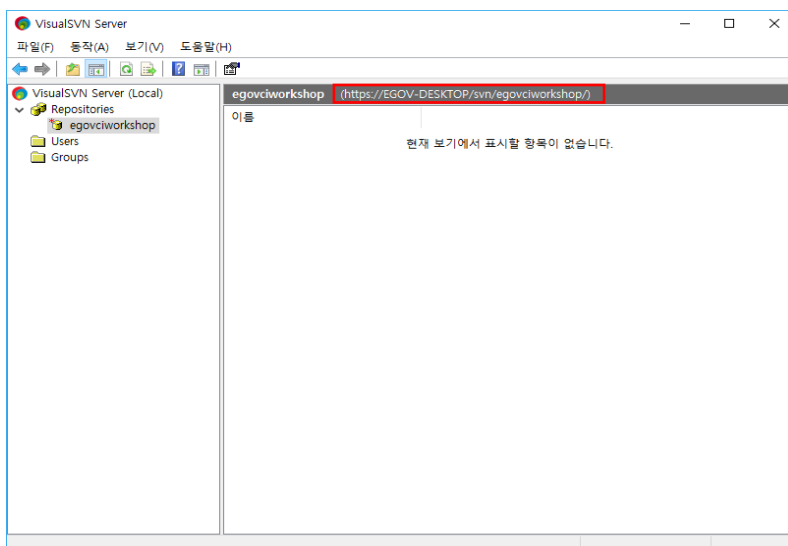
VisualSVN 서버의 구동을 확인하고, "Repositories"의 오른쪽을 클릭하여 "Create New Repository..." 선택, 새로운 저장소를 만든다.

1. 임의 저장소 설정. (ex: egovciworkshop)
2. 필요한 사항을 설정하고 Create 를 한다.

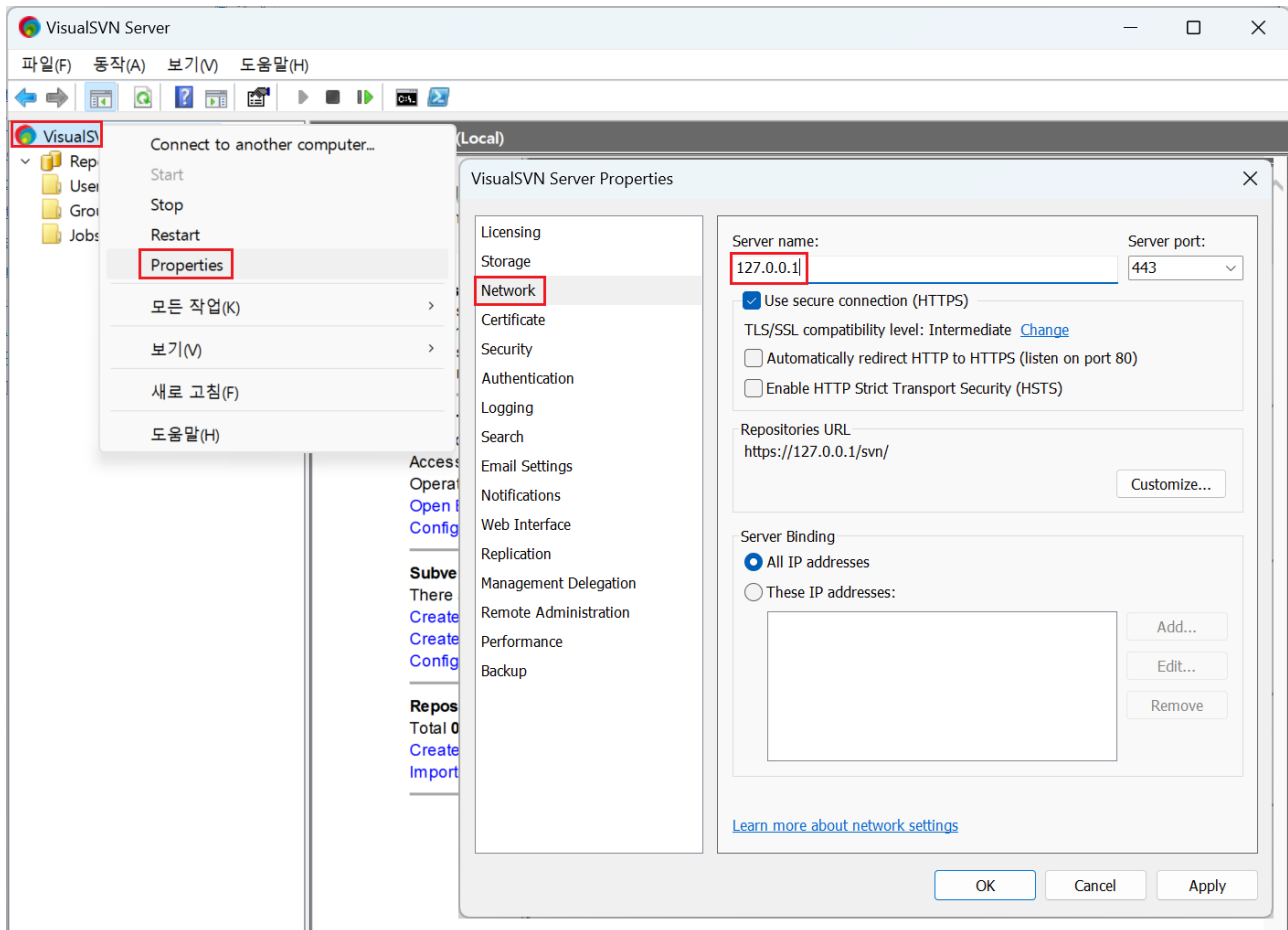


설정을 완료 후, 아래와 같이 저장소 위치를 확인 할 수 있다.

Repository location : "http://<localhost>/svn/egovciworkshop/"



** 참고로 <localhost> 주소를 아래와 같이 실제 IP 주소를 대치하여 사용 가능하다.

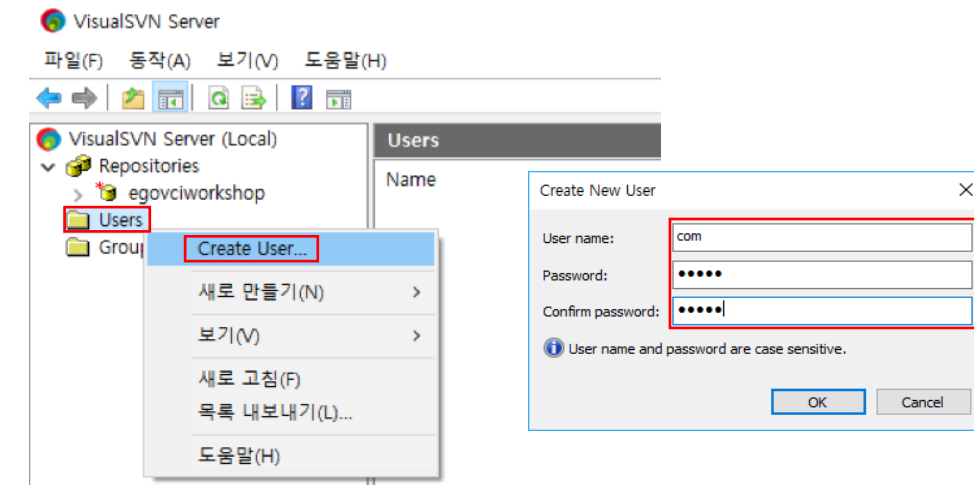


3.2.2 사용자 등록

아래와 같이 "User" 메뉴에서 새로운 사용자를 설정 가능하다.

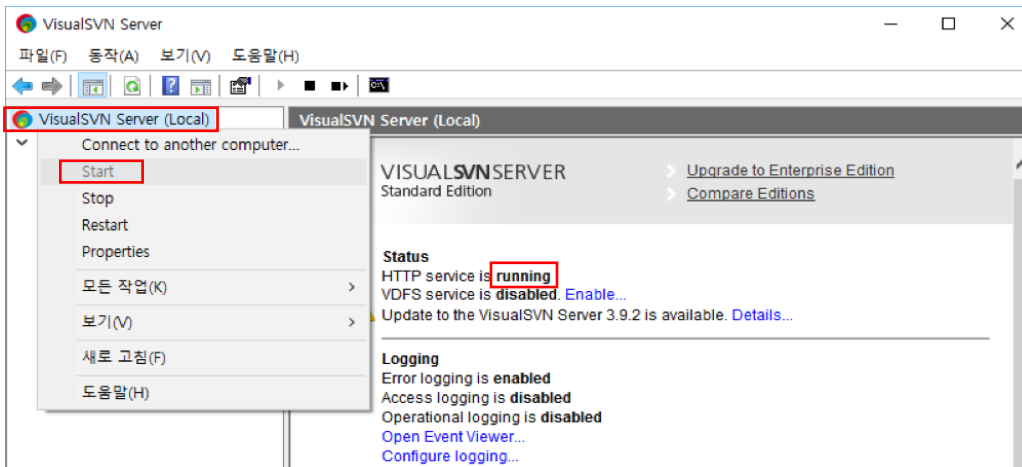
Ex) 사용자 이름: "com"

비밀번호 : "com01"



3.2.3 SVN 서버 시작/종료.

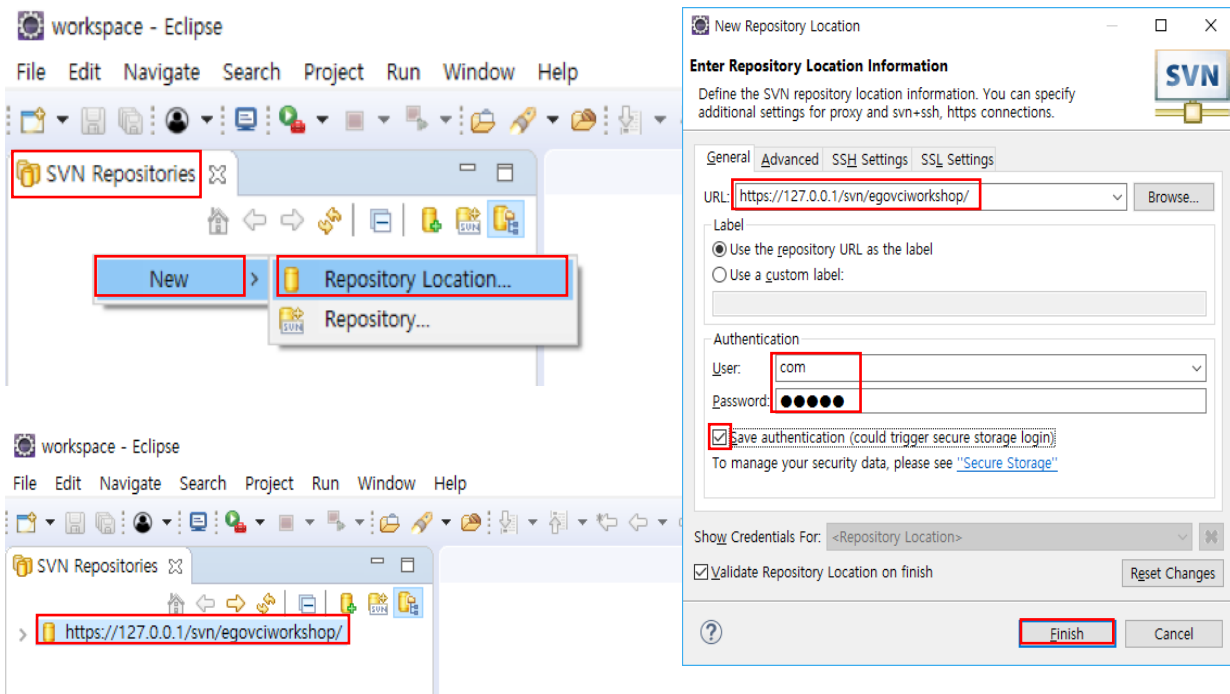
SVN 왼쪽 메뉴에서 아래와 같이 서버를 시작/종료를 할 수 있으며, 확인이 가능하다.



3.2.4 Eclipse 프로젝트를 SVN 서버에 등록.

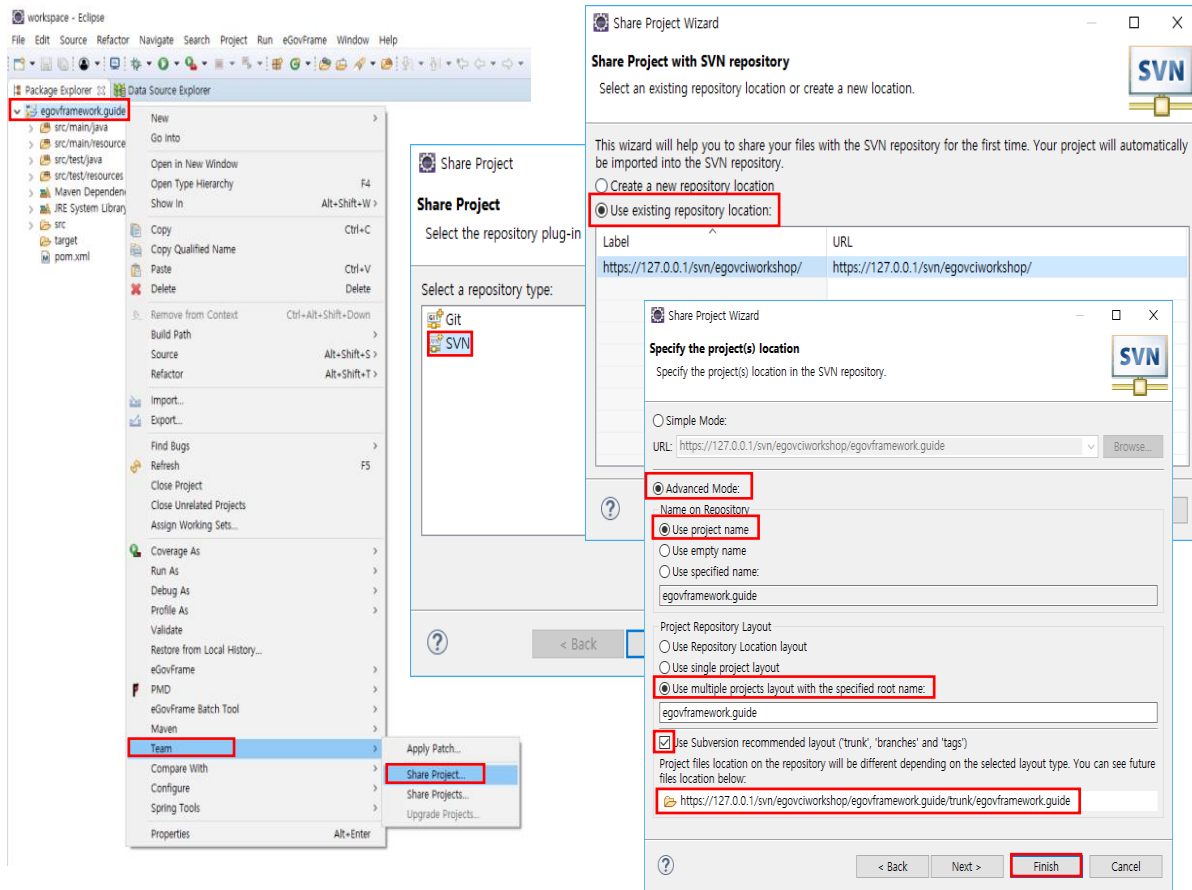
SVN 서버의 정보를 활용하여 Eclipse 내의 프로젝트를 등록하고 소스를 반영한다.

1. Eclipse 에서 SVN Repositories 를 연다.
2. 새로운 저장소 정보를 생성 및 SVN 정보를 등록한다.

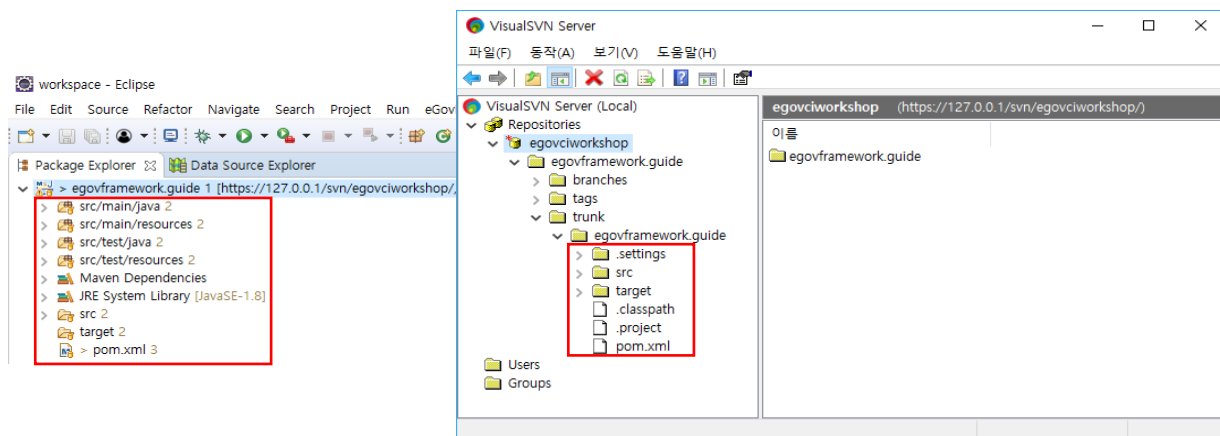


SVN 저장소 정보 등록 후, 프로젝트를 SVN 에 등록 하여야 한다.

1. 프로젝트 선택 후, 오른쪽 마우스를 클릭하여 “Team -> share Project” 로 프로젝트를 SVN 에 공유한다.



등록된 프로젝트는 SVN 서버 Manager 에서 다시 확인 가능하다.

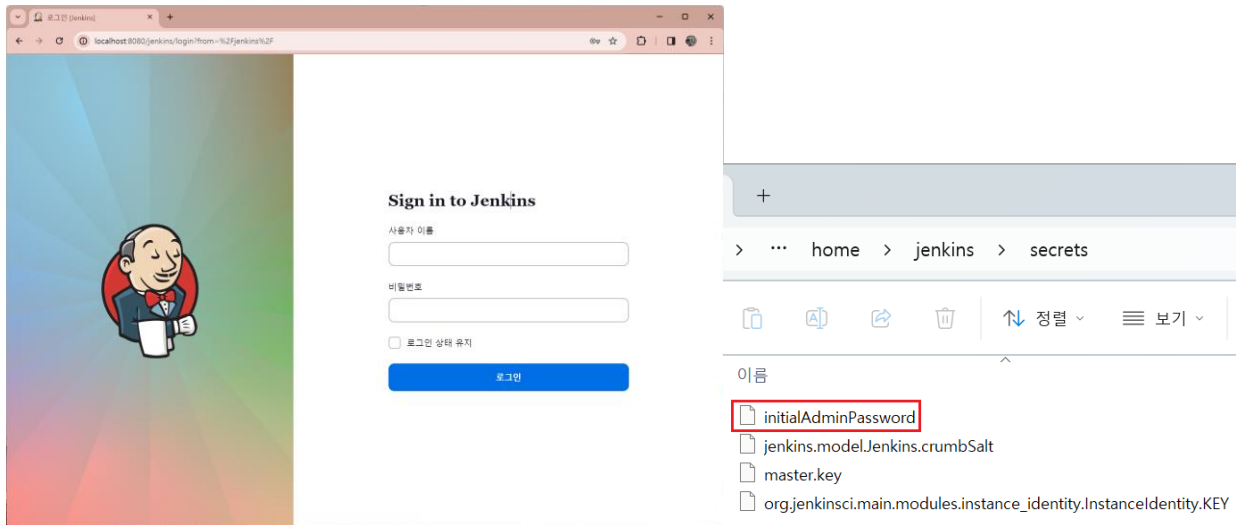


3.3 Jenkins 설정.

다음은 Jenkins 설정으로 아래와 같은 주소를 통해 접속이 가능하다.

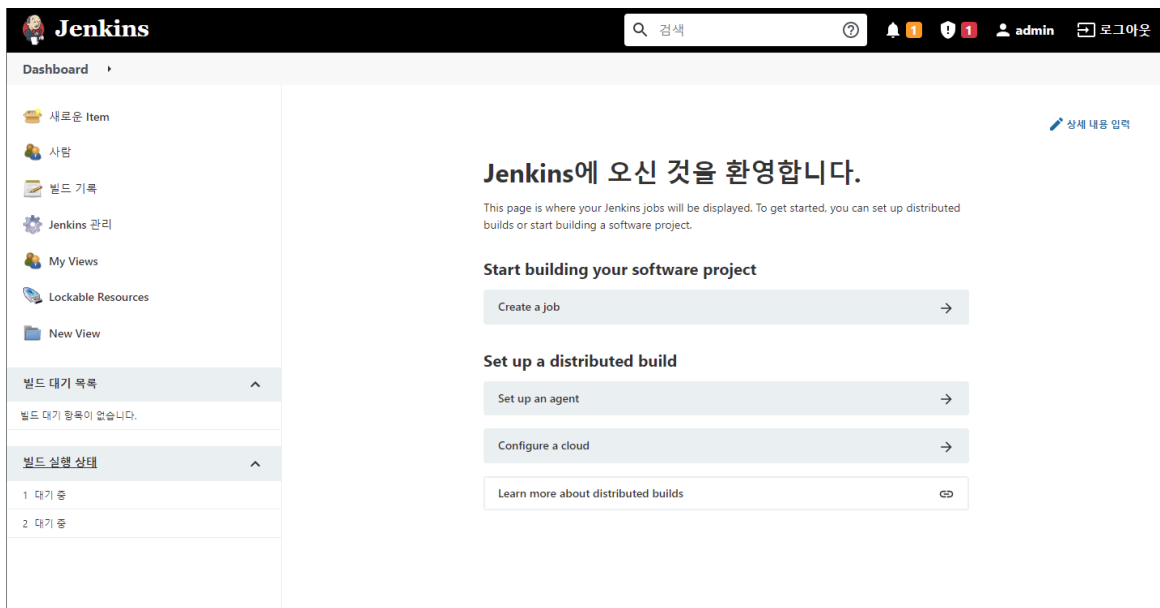
<http://<localhost>:8080/jenkins/>

** 접속이 되지 않은 경우, "start-jenkins.bat"을 통해 Tomcat 를 실행한다.



Ex) 사용자 이름: "admin"

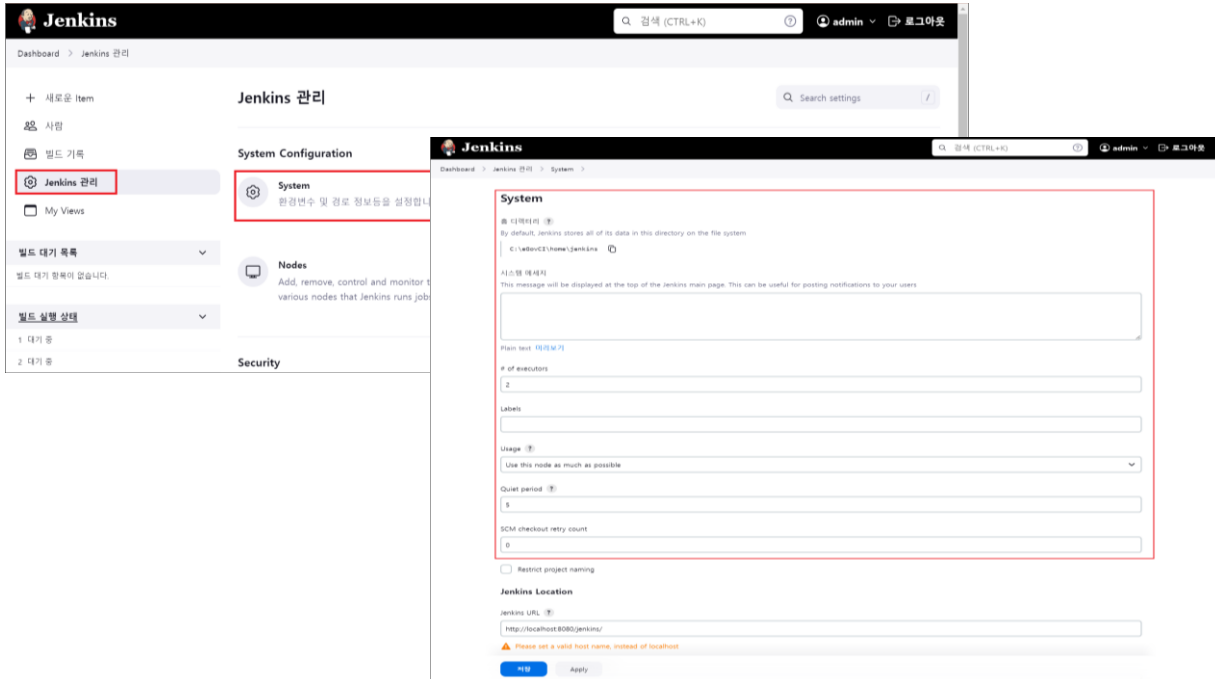
비밀번호 : <initialAdminPassword> 파일 확인



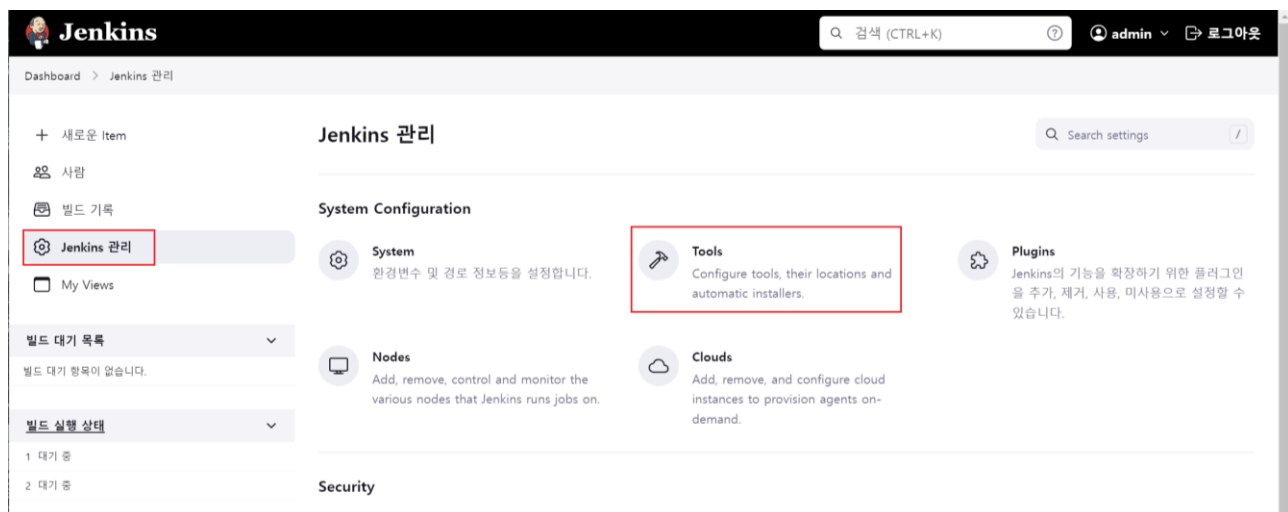
3.3.1 Jenkins 기본 시스템 설정

Jenkins 의 새 작업 등록 전, Jenkins 의 기본 시스템 설정한다. 설정을 통하여 JDK, ANT, Maven, SVN 서버 정보를 등록하여 프로젝트의 컴파일, Deploy 을 할 수 있다.

시스템 설정은 Jenkins 관리 -> Configure System 메뉴를 통해서 할 수 있다..



아래와 같이 JDK, ANT, MAVEN home 디렉토리를 설정한다.



- apache-ant-1.10.14
- apache-maven-3.9.6
- apache-tomcat-9.0.85-jenkins
- jdk8u242-b08
- jdk-17.0.10+7
- nexus-2.15.1-02-bundle
- VisualSVN-Server-5.3.2-x64

Maven Configuration

Default settings provider

Settings file in filesystem

File path ?

C:\WeGovCI\Wbin\Wapache-maven-3.9.6\conf\settings.xml

Path to settings.xml file, relative to project workspace or absolute (variables are supported).

Default global settings provider

Global settings file on filesystem

File path ?

C:\WeGovCI\Wbin\Wapache-maven-3.9.6\conf\settings.xml

JDK installations

Add JDK

JDK

Name
jdk-11.0.22+7

JAVA_HOME
C:\WeGovCI-4.2.0_64bit\Wbin\jdk-11.0.22+7

Install automatically ?

Add JDK

Ant installations

Add Ant

Ant

Name
apache-ant-1.10.14

ANT_HOME
C:\WeGovCI\Wbin\apache-ant-1.10.14

Install automatically ?

Add Ant

Maven installations

Add Maven

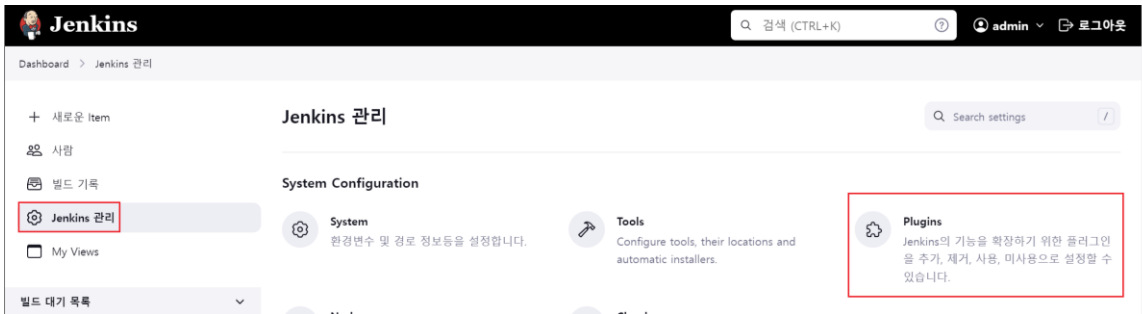
Maven

Name
C:\WeGovCI\Wbin\Wapache-maven-3.9.6

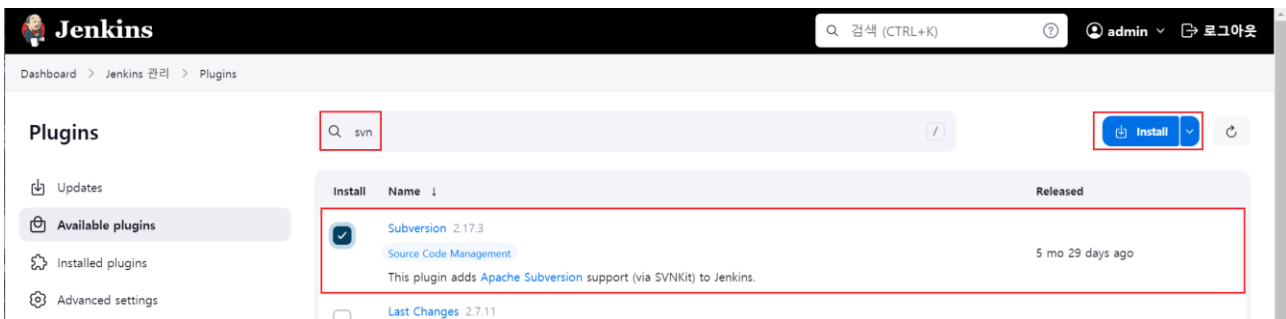
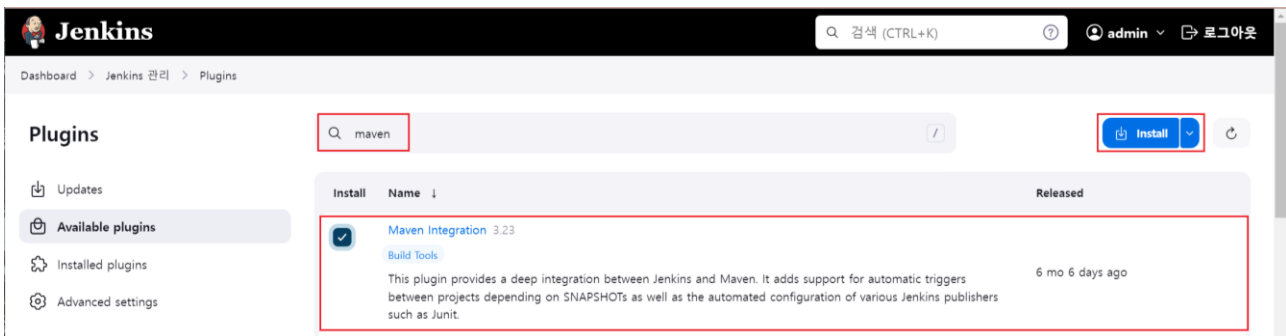
MAVEN_HOME
C:\WeGovCI\Wbin\Wapache-maven-3.9.6

Install automatically ?

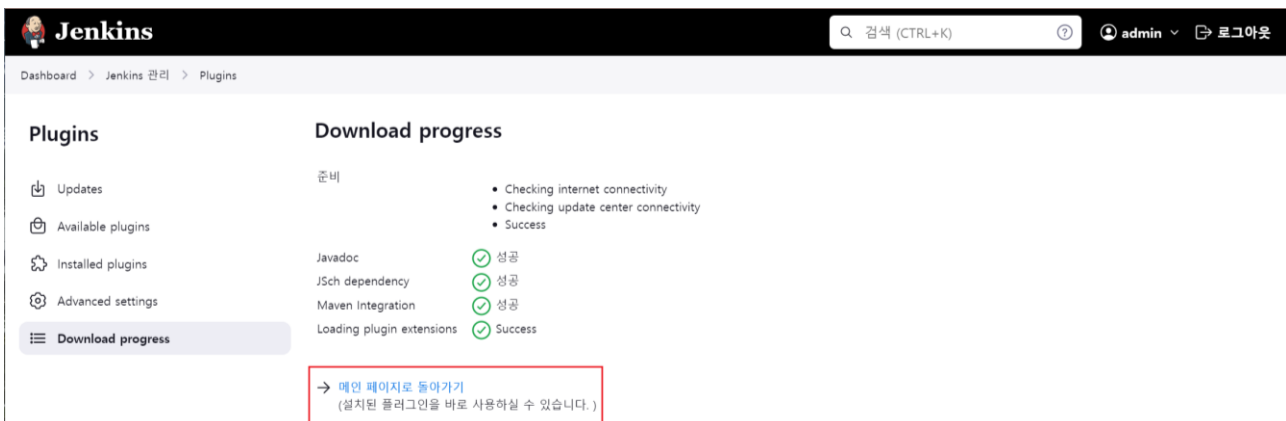
아래와 같이 플러그인을 다운로드 받을 수 있다.



maven 과 svn 플러그인을 설치한다.

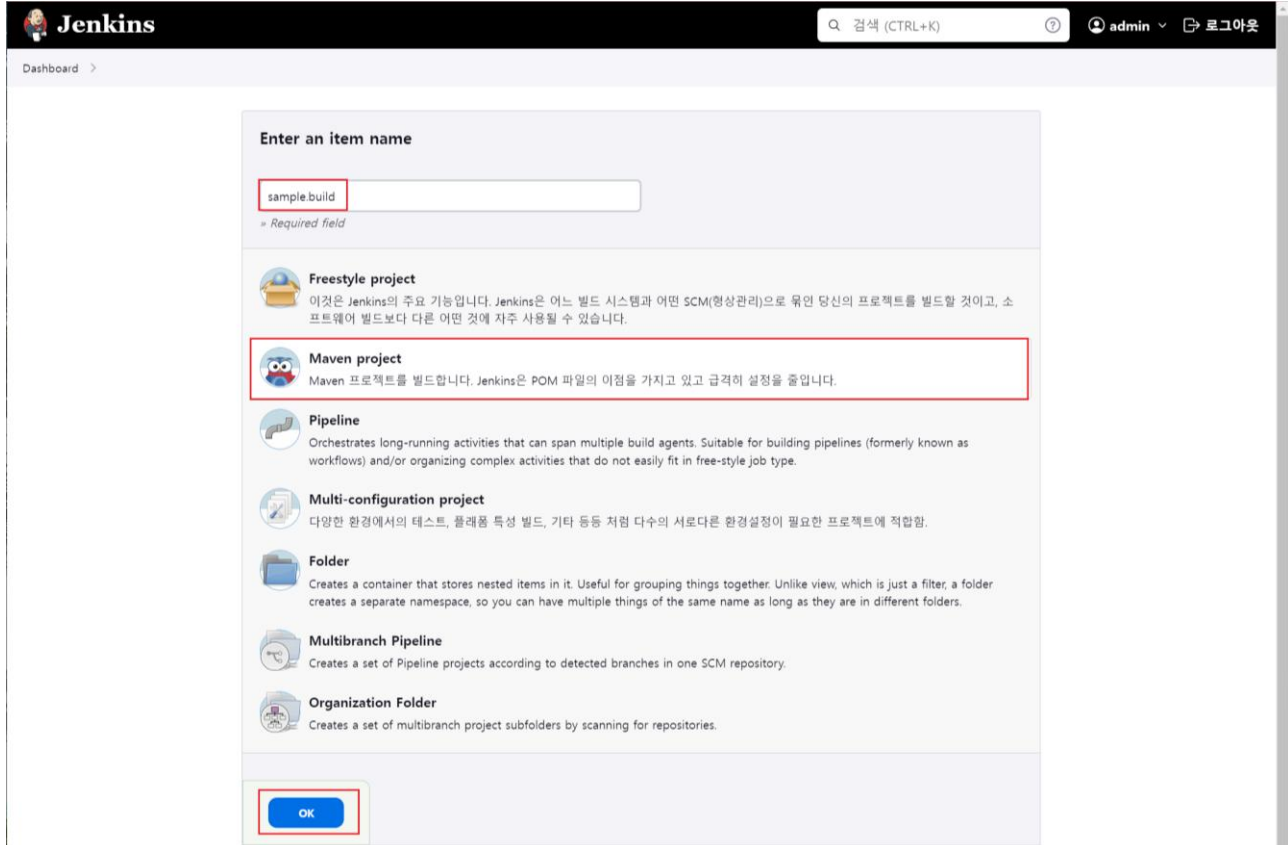
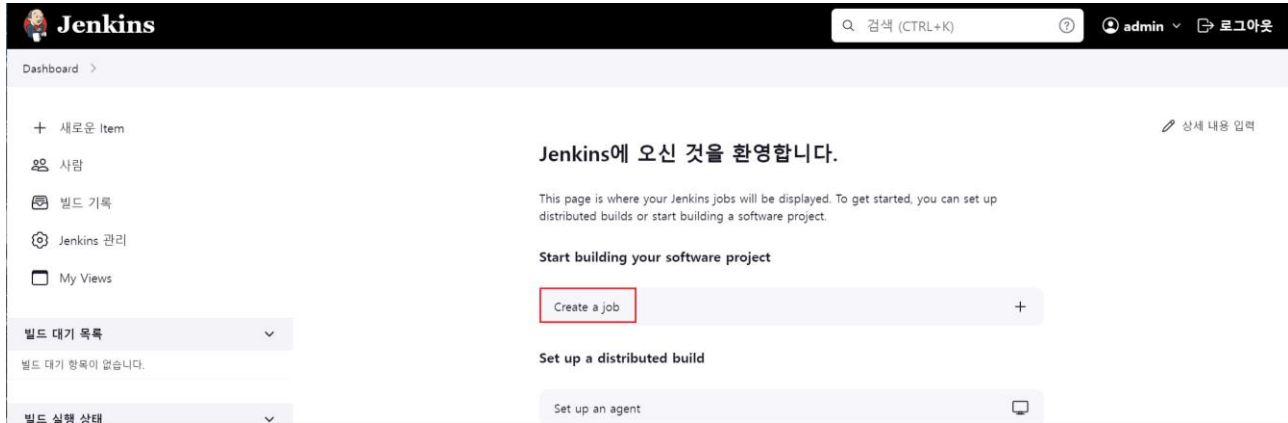


설치완료 후 메인 페이지로 돌아간다. (설치가 반영이 안되었을 경우, jenkins 를 재시작한다.)

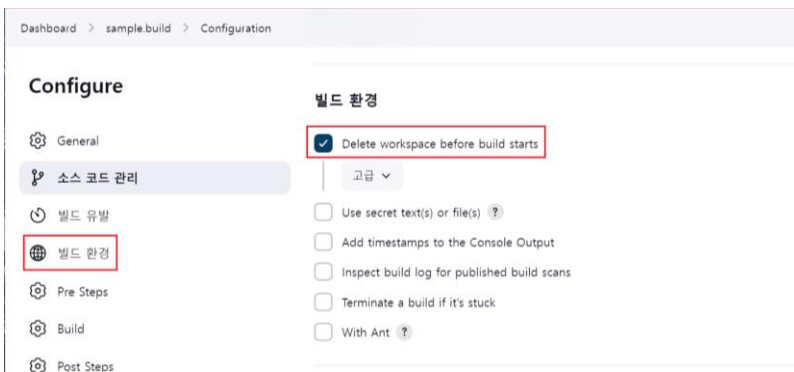
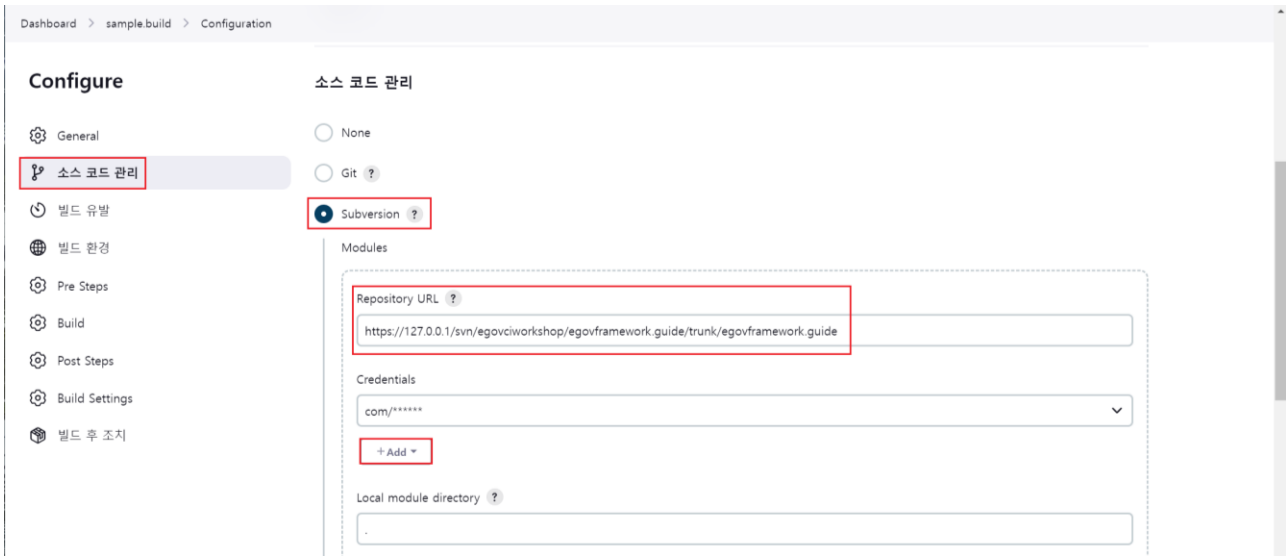


3.3.2 새 작업 생성

통합 환경에서 자동으로 프로젝트 **complie** 을 위해 아래와 같이 새 작업을 생성한다.



Source Code Management 에서 SVN 에 등록된 프로젝트 정보 및 사용자 정보를 아래와 같이 설정한다.



Goals: type "clean install" (which means of MVN clean and install command)

Dashboard > sample.build > Configuration

Configure

- General
- 소스 코드 관리
- 빌드 유발
- 빌드 환경
- Pre Steps
- Build**
- Post Steps
- Build Settings
- 빌드 후 조치

Build

Root POM ?
pom.xml

Goals and options ?
clean install

고급 ▼ Edited

Post Steps

Run only if build succeeds

Dashboard > sample.build > Configuration

Configure

- General
- 소스 코드 관리
- 빌드 유발
- 빌드 환경
- Pre Steps
- Build**
- Post Steps**
- Build Settings
- 빌드 후 조치

Goals and options ?
clean install

고급 ▼ Edited

Post Steps

Run only if build succeeds

Run only if build succeeds or is unstable

Run regardless of build result

Should the post-build steps run only for successful builds, etc.

Add post-build step ▼


Build Settings

E-mail Notification

빌드 후 조치

빌드 후 조치 추가 ▼

저장 Apply

 **Jenkins**

admin
로그아웃

Dashboard >

- + 새로운 Item
- 사람
- 빌드 기록
- Jenkins 관리
- My Views

[상세 내용 입력](#)

All +

S	W	Name ↓	최근 성공	최근 실패	최근 소요 시간
☺	☀	sample.build	—	—	—

아이콘: S M L
Icon legend
Atom feed for all
Atom feed for failures
Atom feed for just latest builds

빌드 대기 목록

빌드 대기 항목이 없습니다.

빌드 실행 상태

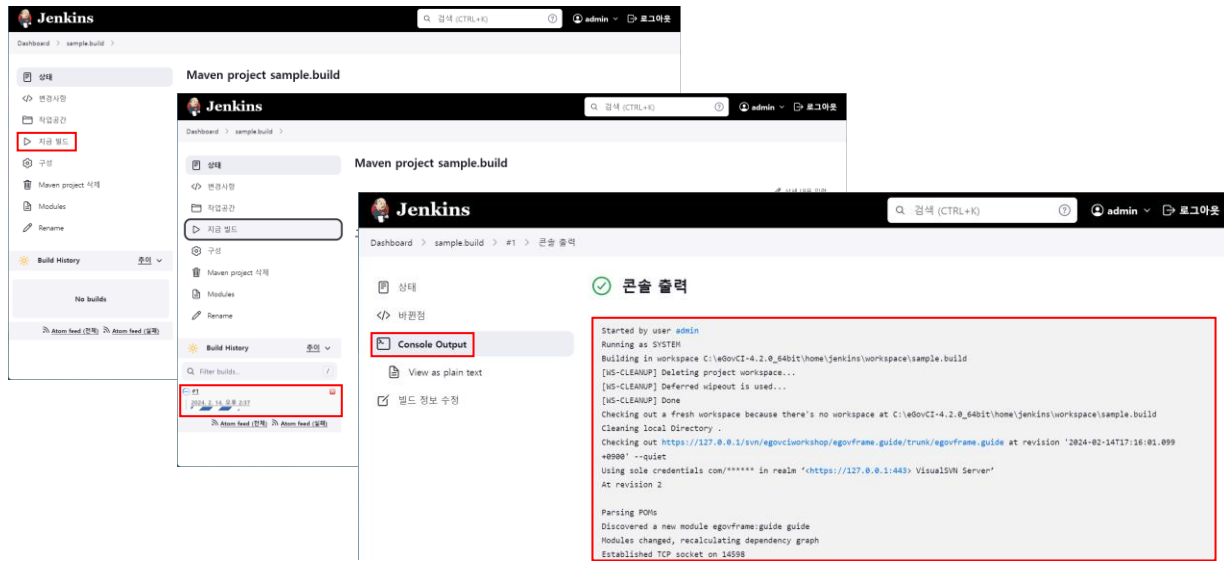
- 1 대기 중
- 2 대기 중

[REST API](#) Jenkins 2.426.3

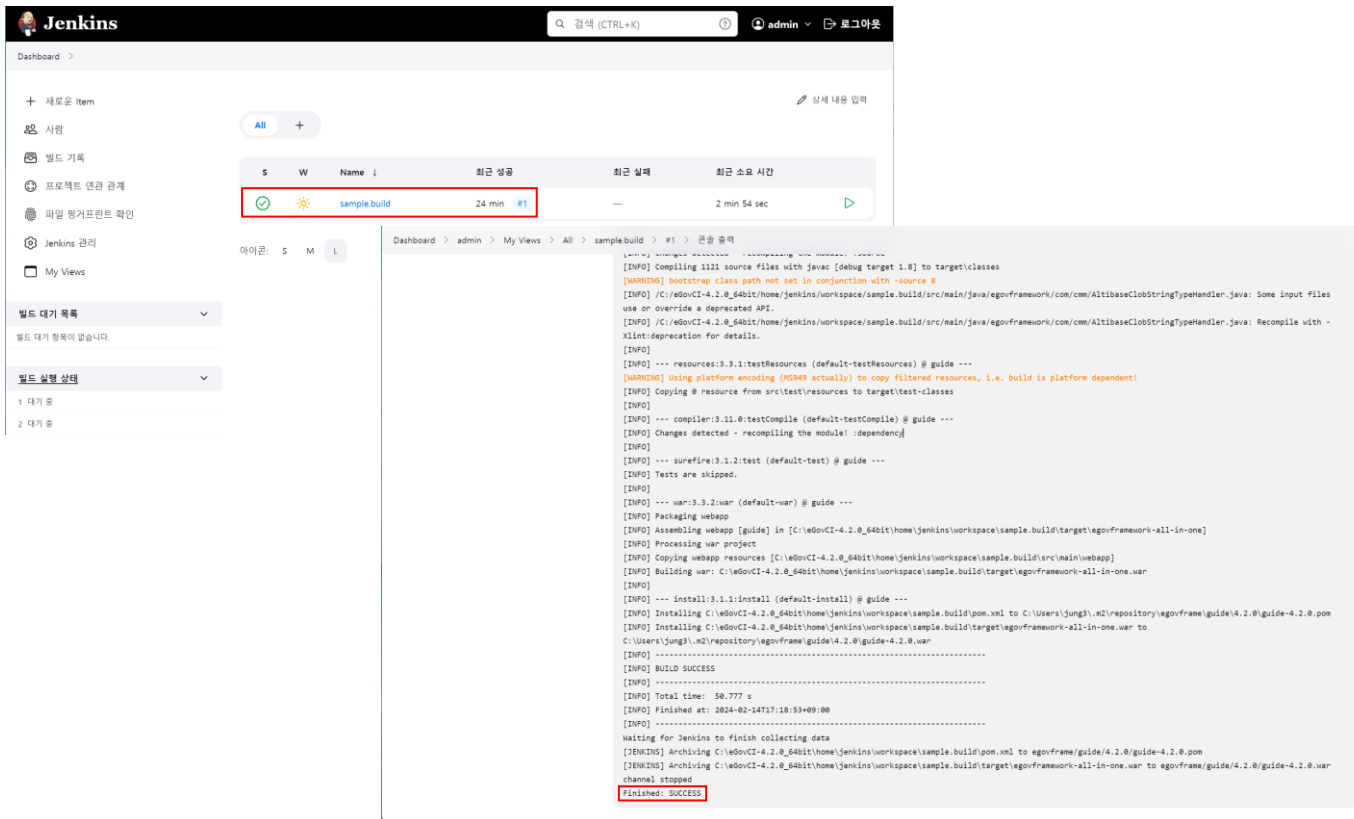
3.3.3 작업 Bulid

모든 설정 이후, 최초 수동으로 Bulid 하여 설정을 확인할 수 있다.

아래와 같이 Build Now”메뉴를 통하여, Maven build 를 실행한다. 빌드 실행 시 콘솔을 통하여 빌드 프로세스를 확인할 수 있다.



아래와 같이 작업 아이콘 또는 콘솔의 정보로 빌드를 확인한다.



정상적으로 빌드가 끝나면, Workspace 에서 빌드를 통하여 만들어진 war(servlet 웹프로젝트의 경우) / jar(자바프로젝트의 경우) 파일을 확인할 수 있다.

