

표준프레임워크
Nexus 및 **CI** 환경 구축 가이드
Version 3.8

Index

1. 표준프레임워크 EGOVCI 패키지 설치	3
1.1 개요.....	3
1.2 EGOVCI 압축풀기.....	3
1.3 EGOVCI 시스템 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 NEXUS 저장소를 프로젝트에 반영.....	8
2.6 NEXUS 저장소에 3 RD PARTY 라이브러리 추가.....	9
3. CI (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 SVN 서버에 등록.....	14
3.3 JENKINS 설정.....	16
3.3.1 Jenkins 기본 시스템 설정	17
3.3.2 새 작업 생성.....	18
3.3.3 작업 Bulid.....	22

1. 표준프레임워크 eGovCI 패키지 설치

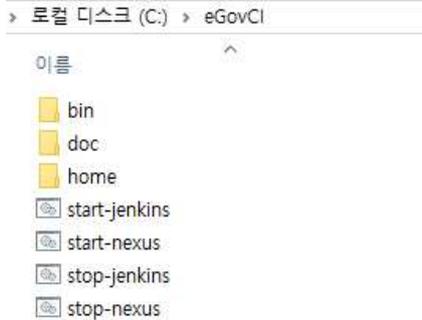
1.1 개요

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

- eGovCI 패키지 (32bit, 64bit 별도 제공)
- Nexus repository 구성 정보.
- CI 관련 구성 정보 (SVN 포함).

1.2 eGovCI 압축풀기

전자정부 표준프레임워크에서 제공하는 eGovCI ("eGovCI-X.X.X_XXbit.exe") 파일을 "C:\\"에 압축을 푼다. 압축을 푼 구조는 아래와 같다.



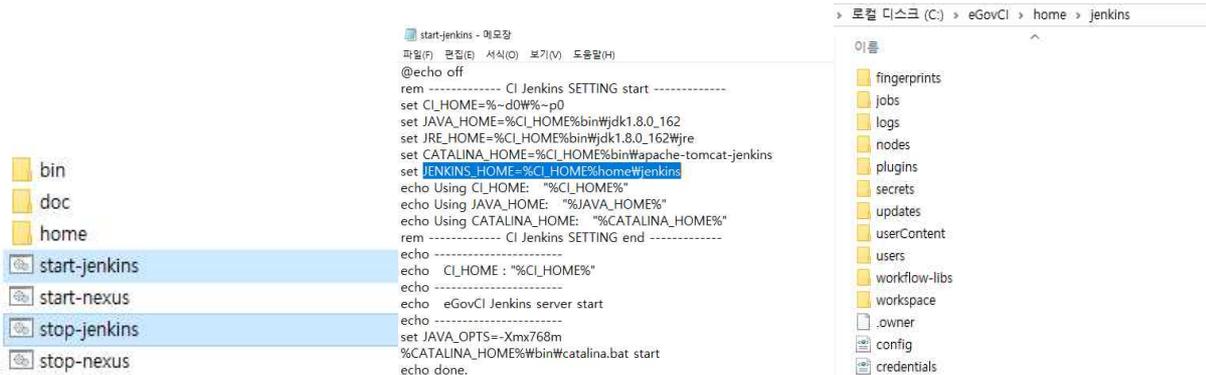
1.3 eGovCI 시스템 구성

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

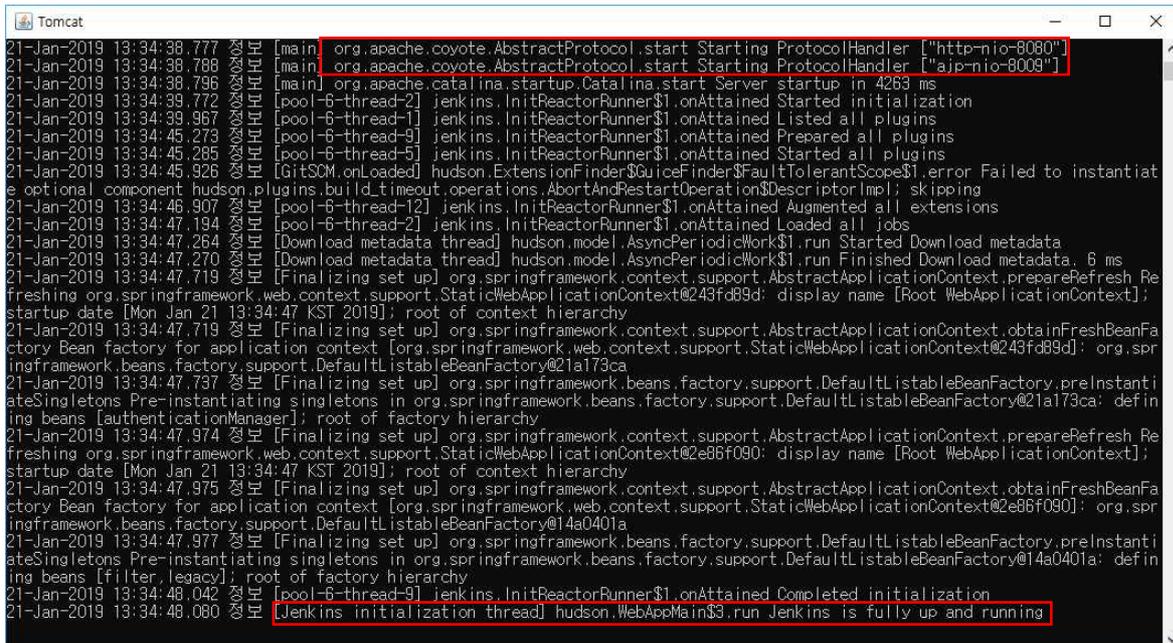
- Ant-1.9.5 (under /bin dictory)
- Maven-3.3.3 (under /bin dictory)
- Tomcat-8.0.47 (under /bin dictory)
- JDK.1.8 (under /bin dictory)
- VisualSVN-Server-3.8.4.msi(Instatallation file - 32.bit/64bit 별도 제공) (under /bin dictory)

1.4 CI 시스템 구동 (Start/Stop)

CI 시스템 구성 이후, “start-jenkins.bat”, “stop-jenkins.bat” 파일로 Jenkins 서버 구동 및 Jenkins 서버를 멈출 수 있다.



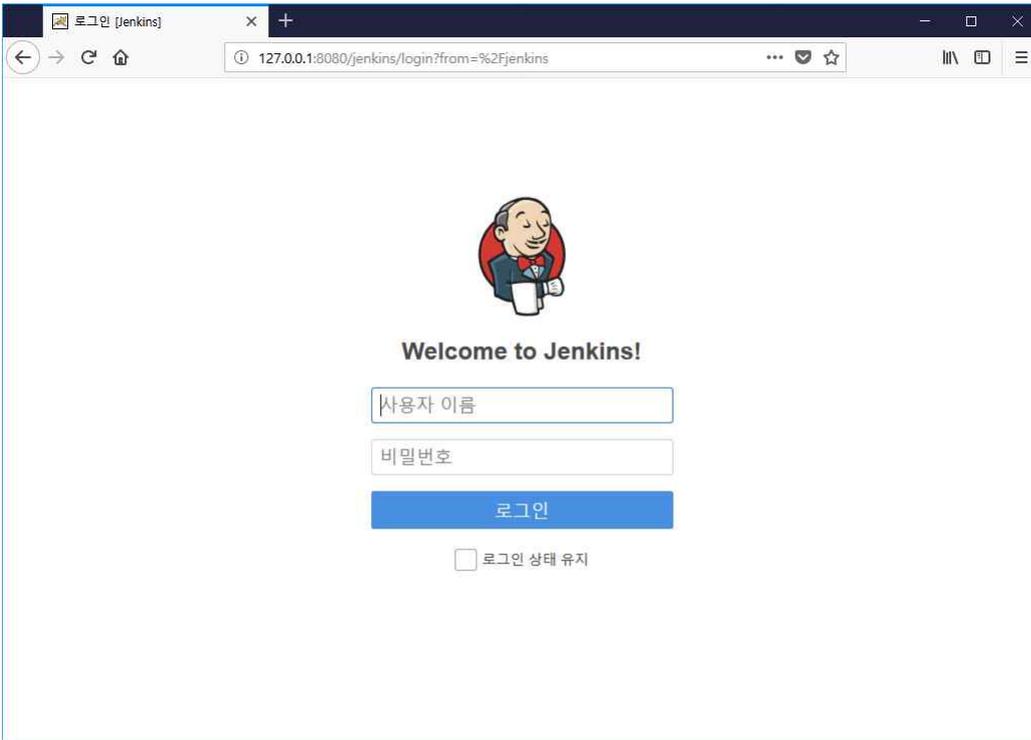
구동 이후, 콘솔 창에서 아래와 같은 내용을 참조할 수 있으며, 해당 포트 넘버를 확인 할 수 있다.



** 서버 성능에 따라 구동 시간이 다소 소요될 수 있다.

구동후, 아래와 같은 주소로 CI 웹 페이지를 확인 할 수 있다.

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



** CI 서버 기본 포트는 '8080'으로 설정되어 있으며, 아래와 같이 별도 포트로 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”, “stop-nexus.bat” 파일로 Nexus 서버 구동 및 Nexus 서버를 멈출 수 있다.



구동 이후, 콘솔 창에서 아래와 같은 내용을 참조할 수 있으며, 해당 포트 넘버를 확인 할 수 있다.

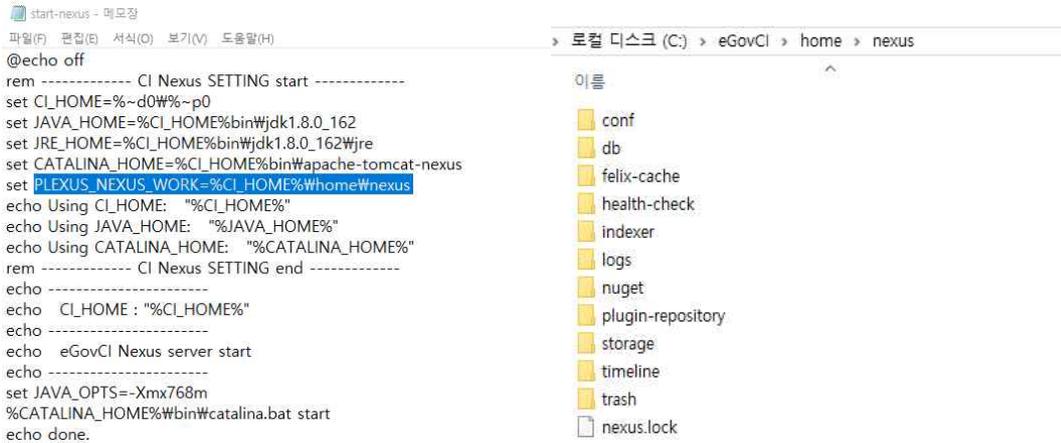
```

Tomcat
-egovframe - Next attempt to auto-unblock the "egovframe" (id=egovframe) repository by checking its remote peer health will occur in 5 minutes.
2019-01-21 14:51:24.316+0900 INFO [proxy-3-thread-1] *UNKNOWN org.sonatype.nexus.proxy.storage.remote.httpclient.HttpClientRemoteStorage - Initializing remote transport for proxy repository "egovframe" [id=egovframe]...
2019-01-21 14:51:24.414+0900 WARN [proxy-3-thread-1] *UNKNOWN org.sonatype.nexus.proxy.maven.maven2.M2Repository - Remote peer of proxy repository "egovframe" [id=egovframe] detected as healthy, un-blocking the proxy repository (it was AutoBlocked by Nexus).
2019-01-21 14:51:25.854+0900 INFO [localhost-startStop-1] org.sonatype.nexus.webresources.internal.WebResourceServiceImpl - Discovered 1967 resources
2019-01-21 14:51:25.856+0900 INFO [localhost-startStop-1] org.sonatype.nexus.webresources.internal.WebResourceServlet - Max-age: 30 days (2592000 seconds)
21-Jan-2019 14:51:25.862 정보 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory C:\eGovCI\bin\apache-tomcat-nexus\webapps\nexus has finished in 118,005 ms
21-Jan-2019 14:51:25.863 정보 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory C:\eGovCI\bin\apache-tomcat-nexus\webapps\ROOT
21-Jan-2019 14:51:25.880 정보 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory C:\eGovCI\bin\apache-tomcat-nexus\webapps\ROOT has finished in 17 ms
21-Jan-2019 14:51:25.885 정보 [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-nio-9090"]
21-Jan-2019 14:51:25.890 정보 [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["ajp-nio-9009"]
21-Jan-2019 14:51:25.892 정보 [main] org.apache.catalina.startup.Catalina.start Server startup in 132541 ms
2019-01-21 14:52:04.164+0900 INFO [RepositoryStatusChecker-codehaus-snapshots] org.sonatype.nexus.proxy.maven.maven2.M2Repository-codehaus-snapshots - Next attempt to auto-unblock the "Codehaus Snapshots" (id=codehaus-snapshots) repository by checking its remote peer health will occur in 1 minute 20 seconds.
2019-01-21 14:53:24.166+0900 INFO [RepositoryStatusChecker-codehaus-snapshots] org.sonatype.nexus.proxy.maven.maven2.M2Repository-codehaus-snapshots - Next attempt to auto-unblock the "Codehaus Snapshots" (id=codehaus-snapshots) repository by checking its remote peer health will occur in 2 minutes.
2019-01-21 14:55:24.167+0900 INFO [RepositoryStatusChecker-codehaus-snapshots] org.sonatype.nexus.proxy.maven.maven2.M2Repository-codehaus-snapshots - Next attempt to auto-unblock the "Codehaus Snapshots" (id=codehaus-snapshots) repository by checking its remote peer health will occur in 3 minutes 20 seconds.
    
```

2.2 Nexus 저장소 설정

Nexus 의 저장소는 기본으로 <eGovCI_home>\home\nexus 로 설정되어 있으며,

아래와 같이 별도 설정이 가능하다.



2.3 Nexus 저장소에 라이브러리 추가.

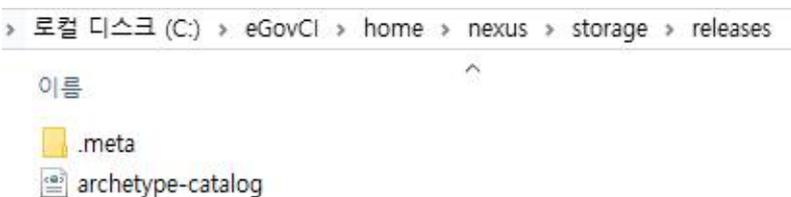
Nexus 저장소에 라이브러리 추가 위치

<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 추가가능하다.

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

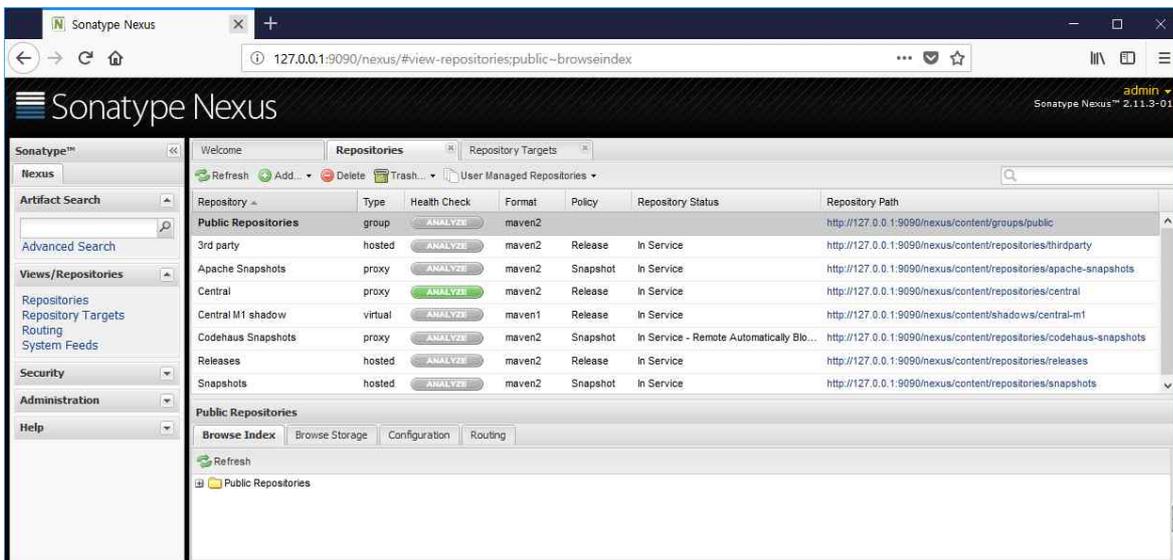


2.4 Nexus 실행 및 추가된 라이브러리 확인.

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

http://127.0.0.1:9090/nexus

Nexus 를 접속 후 “repositories” 선택하고 “public repositories” 또는 release 저장소에서 추가한 라이브러리 목록을 확인 할 수 있다.



2.5 Nexus 저장소를 프로젝트에 반영.

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

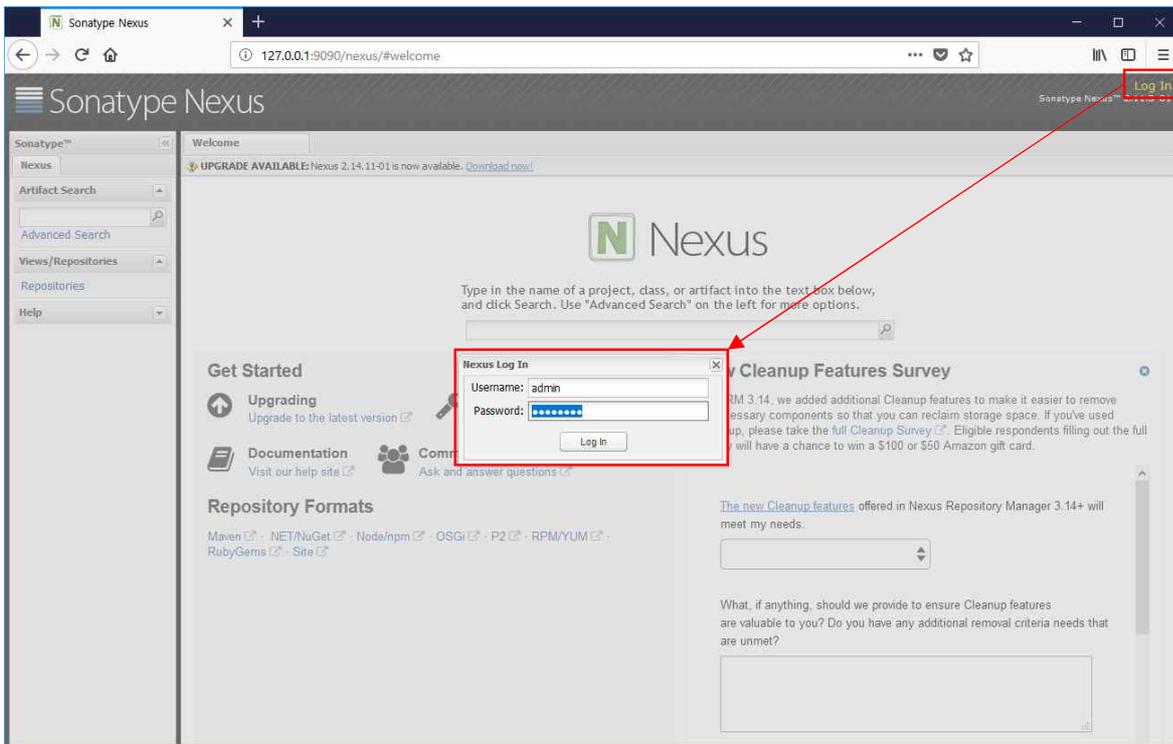
```
<repositories>
  <repository>
    <id>public</id>
    <url>http://127.0.0.1:9090/nexus/content/groups/public/</url>
    <releases>
      <enabled>true</enabled>
    </releases>
    <snapshots>
      <enabled>true</enabled>
    </snapshots>
  </repository>

  <repository>
    <id>egovframe</id>
    <url>http://www.egovframe.go.kr/maven/</url>
    <releases>
      <enabled>true</enabled>
    </releases>
    <snapshots>
      <enabled>false</enabled>
    </snapshots>
  </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 -> GA 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.

The screenshot shows the Sonatype Nexus web interface. The main content area displays a table of repositories. The '3rd party' repository is selected, and the 'Artifact Upload' tab is active. The 'GAV Parameters' dropdown is selected, and the 'Upload Artifact(s)' button is highlighted.

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

3rd party

Browse Index | Browse Storage | Configuration | Routing | Summary | **Artifact Upload**

Select GAV Definition Source

GAV Definition: **GAV Parameters**

Auto Guess:

Group:

Artifact:

Version:

Packaging:

Select Artifact(s) for Upload

Select Artifact(s) to Upload...

Filename:

Classifier:

Extension:

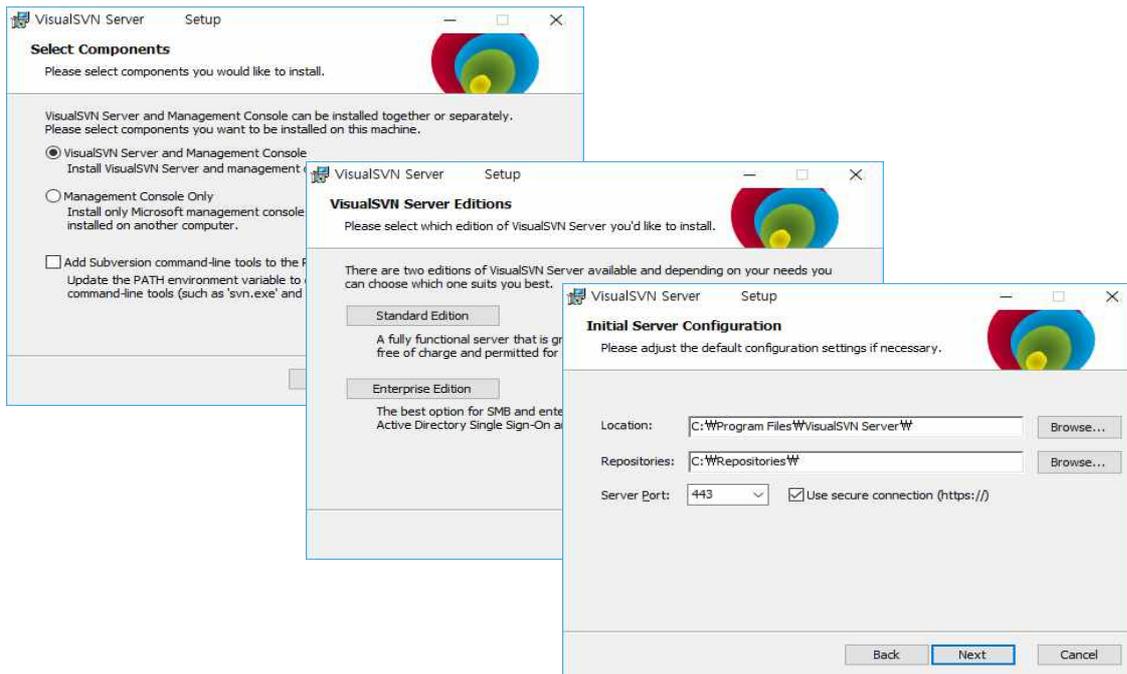
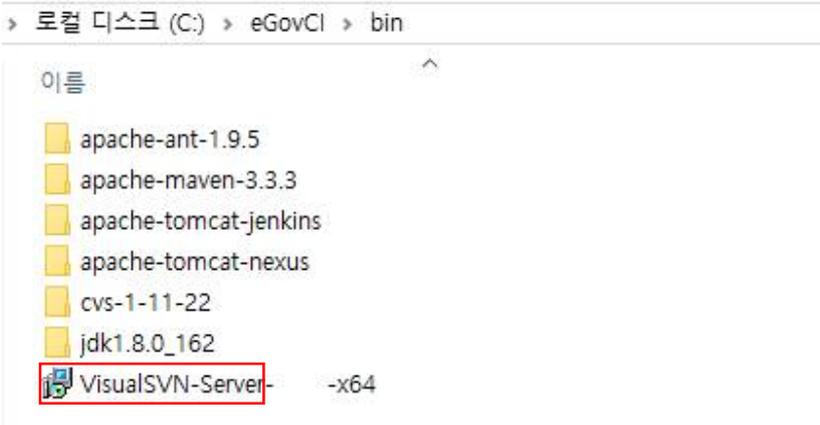
Artifacts

3. CI (Jenkins) 설정

3.1 Subversion 설치 (선택사항)

CI 설정에 앞서 소스 저장소(Subversion)을 설치 한다. eGovCI 에는 VisualSVN 의 설치파일을 별도로 제공하고 있으며, 필요 시 설치하여 사용한다. (별도로 SVN 을 구성하여 사용하여도 무방하다.)

eGovCI 의 VisualSVN-server-XXX.msi 를 설치한다.



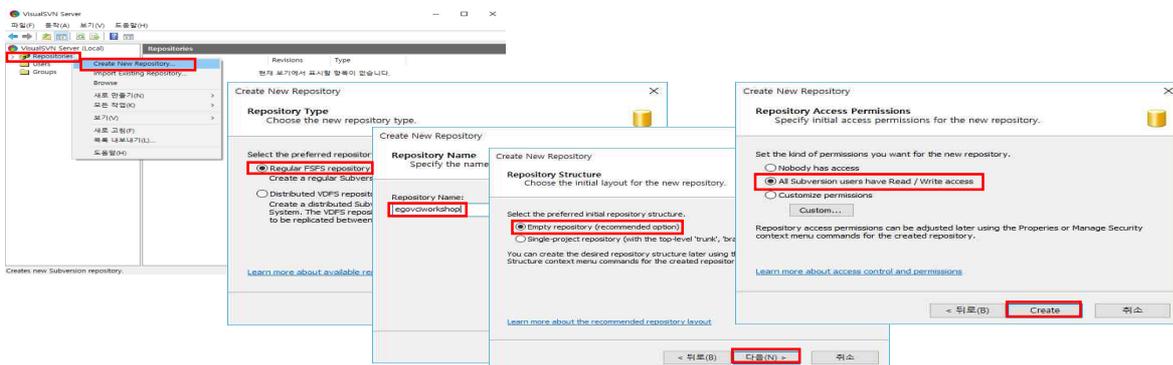
3.2 SVN Repositories 설정 및 사용자 등록.

SVN 저장소 설정 및 사용자 등록 관련 사항을 설정한다.

3.2.1 저장소 설정 및 등록

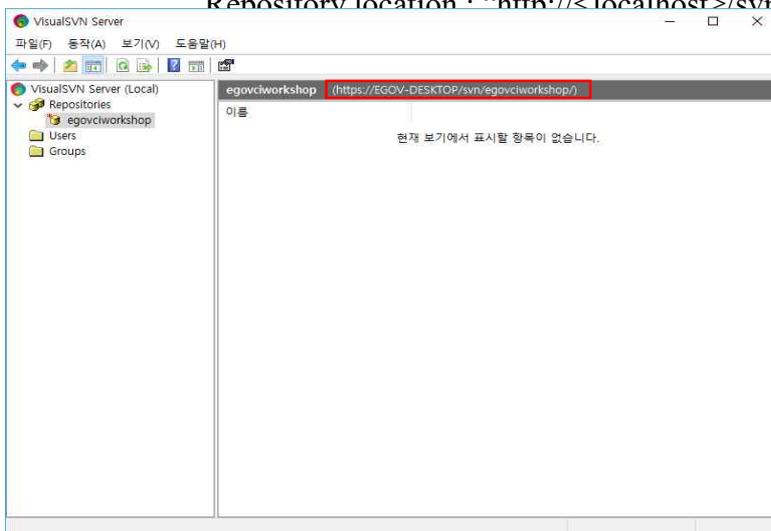
VisualSVN 서버의 구동을 확인 하고, "Repositories"의 오른쪽을 클릭하여 “Create New Repository...” 선택, 새로운 저장소를 만든다.

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

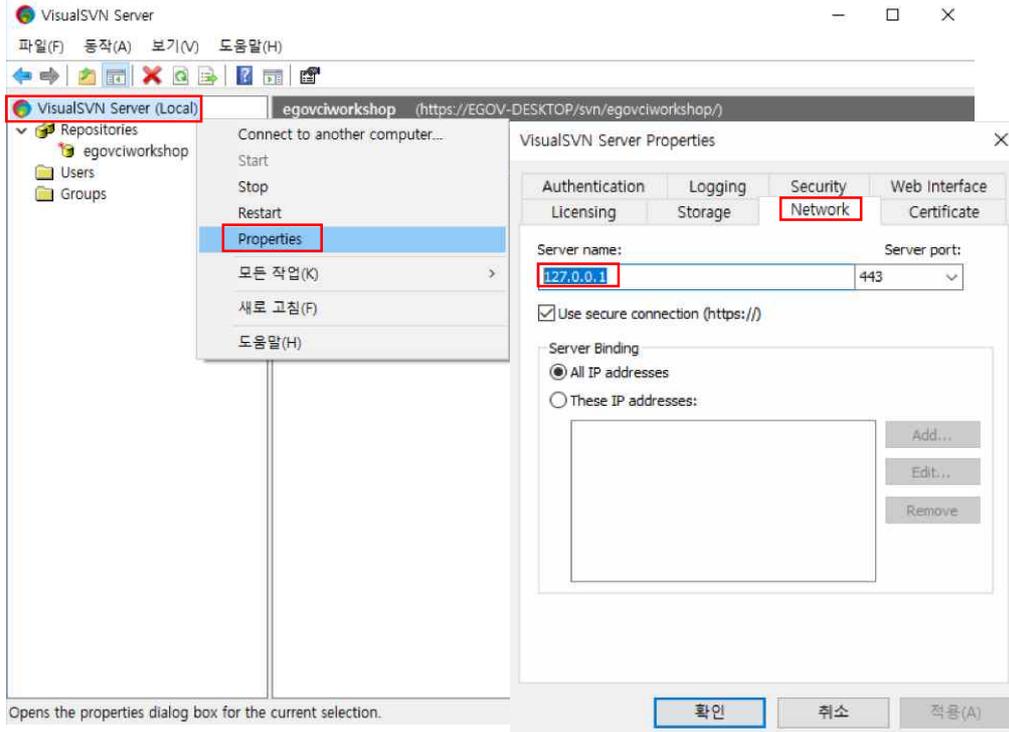


설정을 완료 후, 아래와 같이 저장소 위치를 확인 할 수 있다.

Repository location : “http://<localhost>/svn/egovciworkshop/”



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

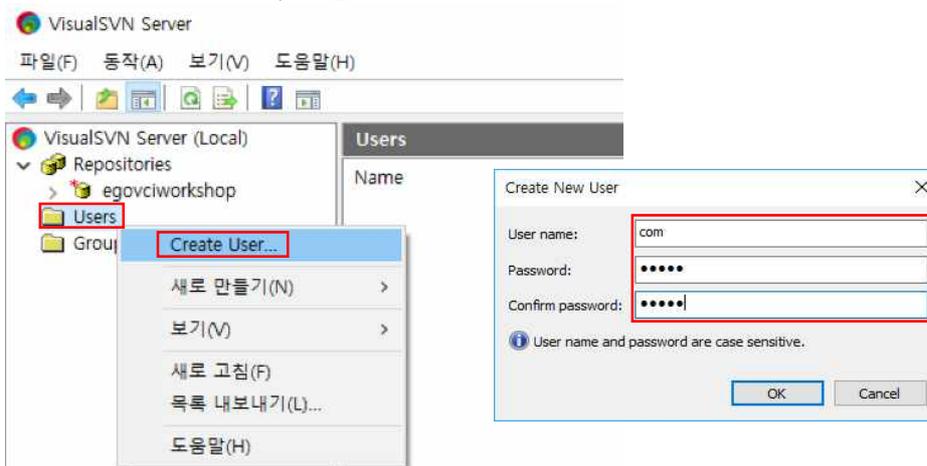


3.2.2 사용자 등록

아래와 같이 "User" 메뉴에서 새로운 사용자를 설정 가능하다.

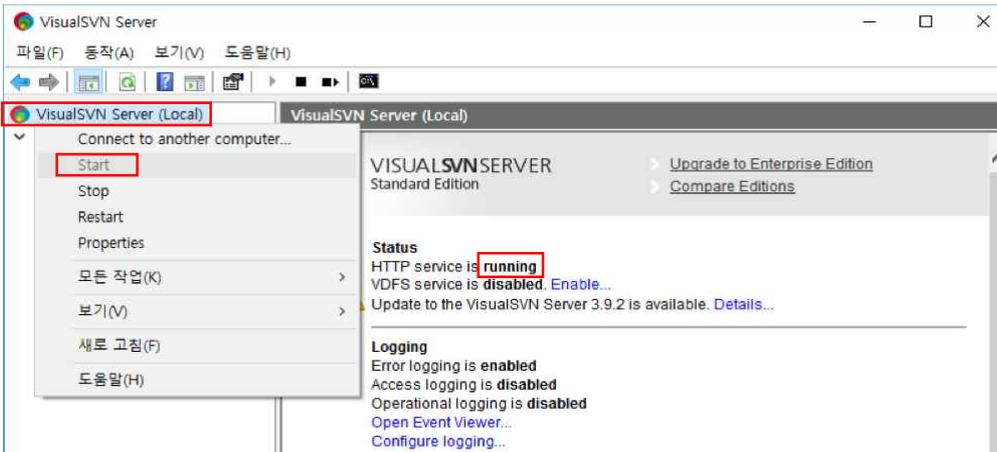
Ex) 사용자 이름: "com"

비밀번호 : "com01"



3.2.3 SVN 서버 시작/종료.

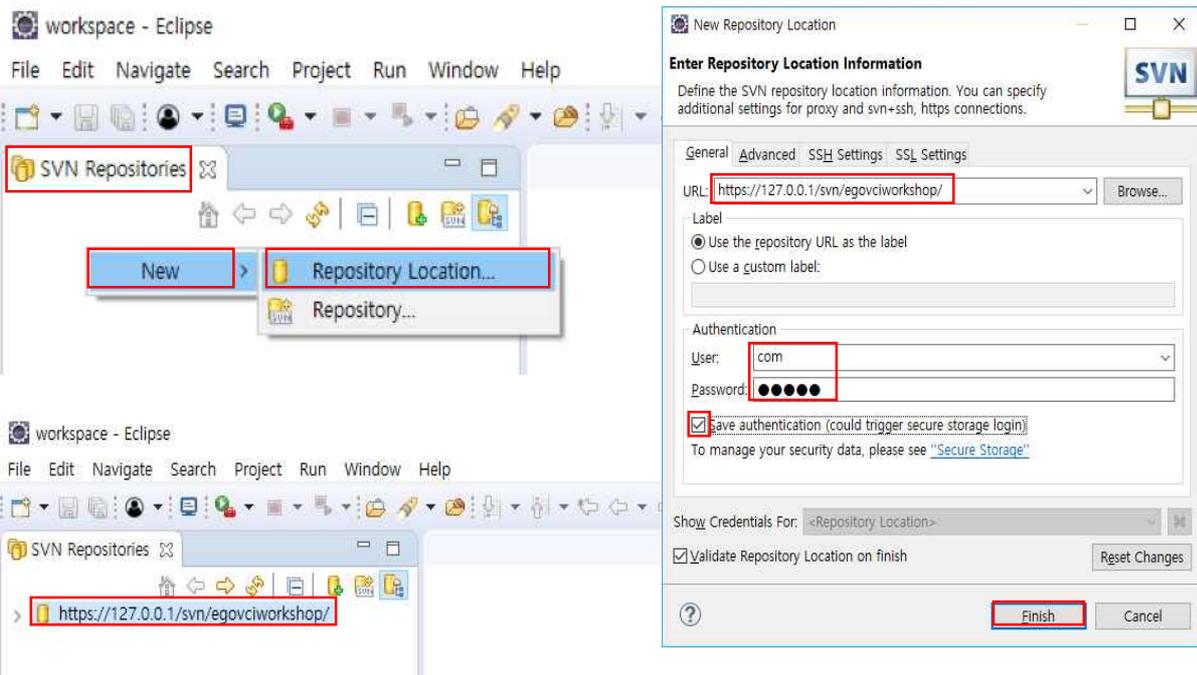
SVN 왼쪽 메뉴에서 아래와 같이 서버를 시작/종료를 할 수 있으며, 확인이 가능하다.



3.2.4 Eclipse 프로젝트를 SVN 서버에 등록.

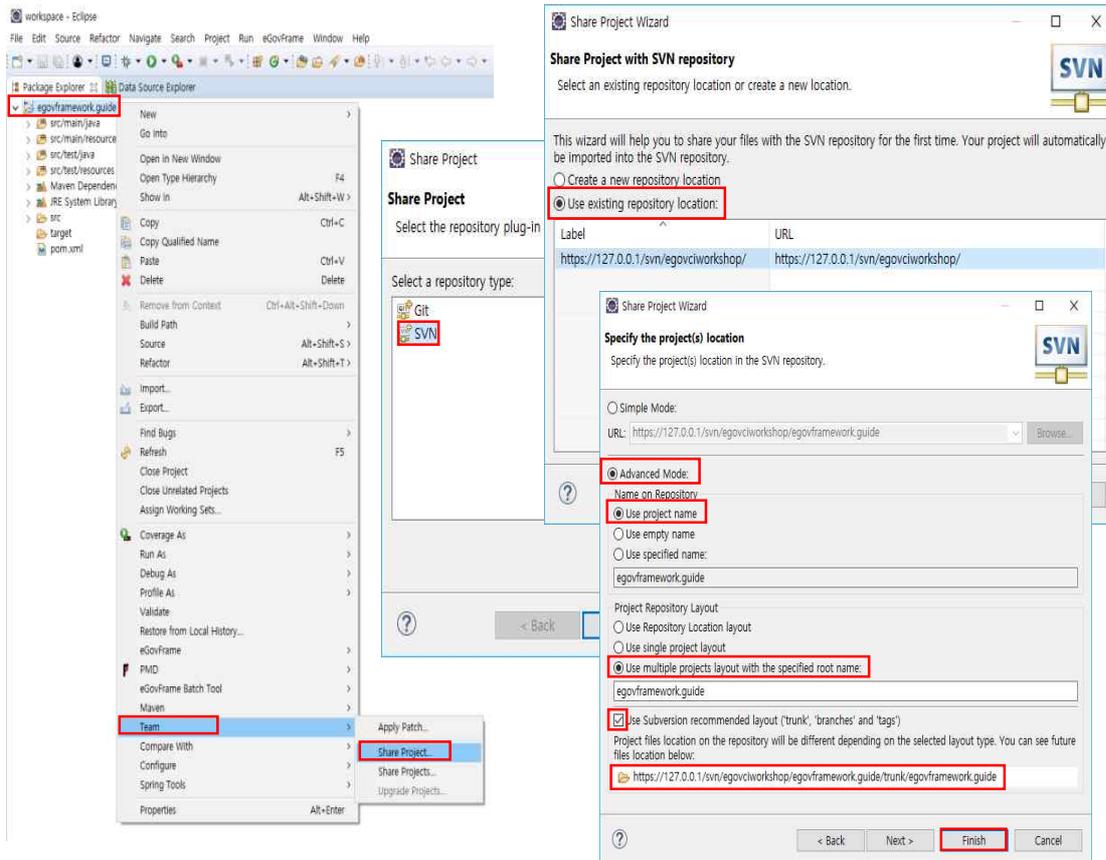
SVN 서버의 정보를 활용하여 Eclipse 내의 프로젝트를 등록하고 소스를 반영한다.

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

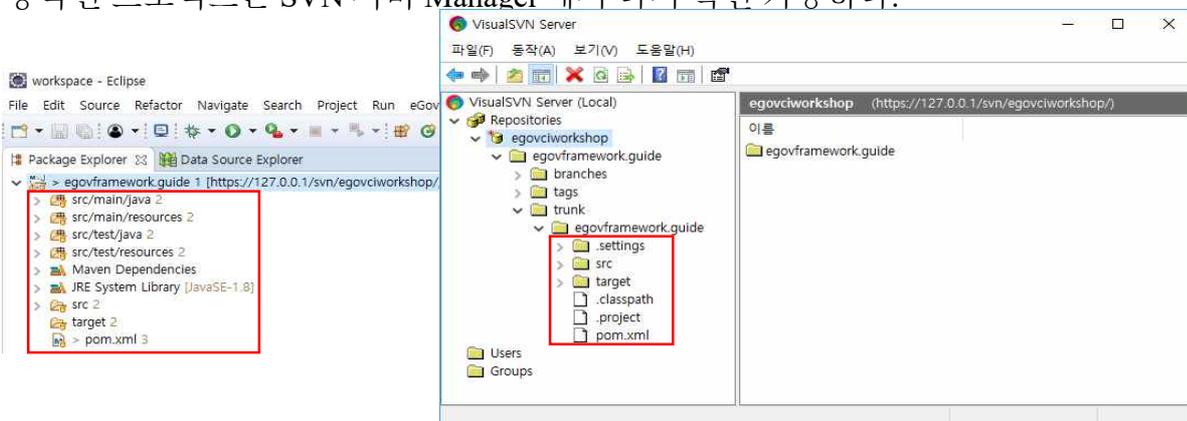


SVN 저장소 정보 등록 후, 프로젝트를 SVN에 등록 하여야 한다.

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



등록된 프로젝트는 SVN 서버 Manager에서 다시 확인 가능하다.

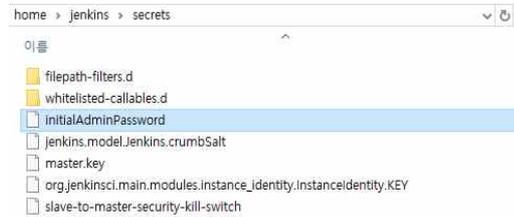
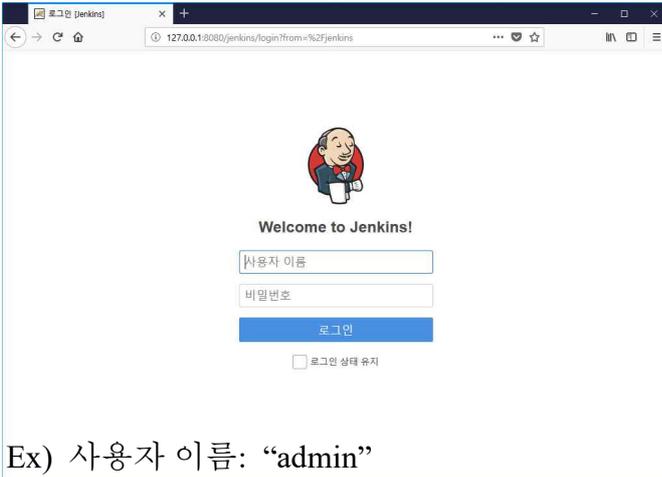


3.3 Jenkins 설정.

다음은 Jenkins 설정으로 아래와 같은 주소를 통해 접속이 가능하다.

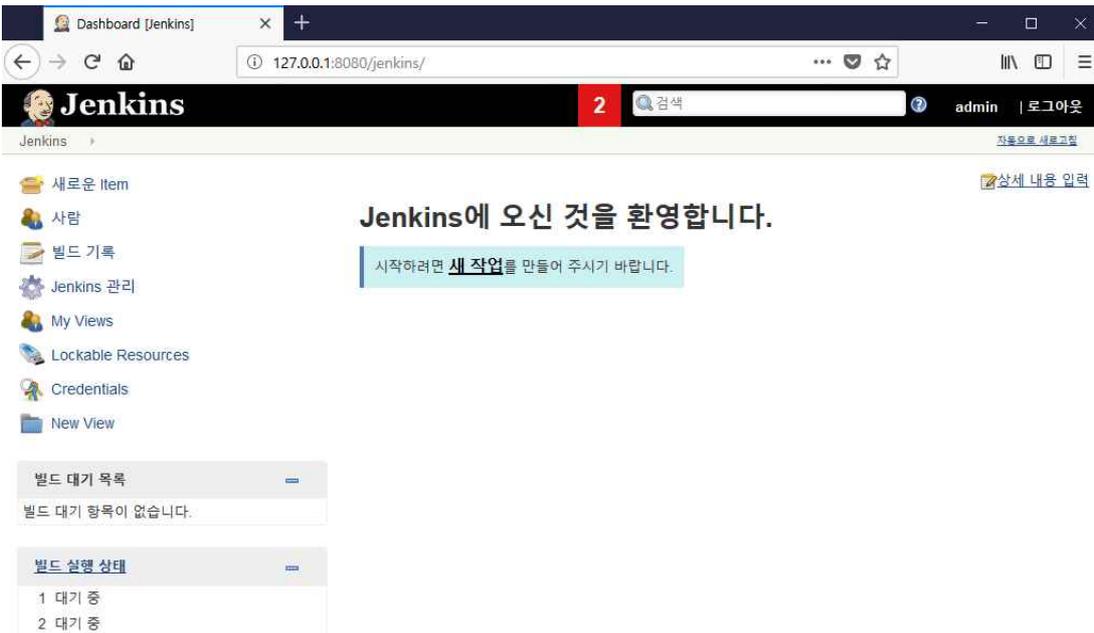
<http://<localhost>:8080:/jenkins/>

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



Ex) 사용자 이름: "admin"

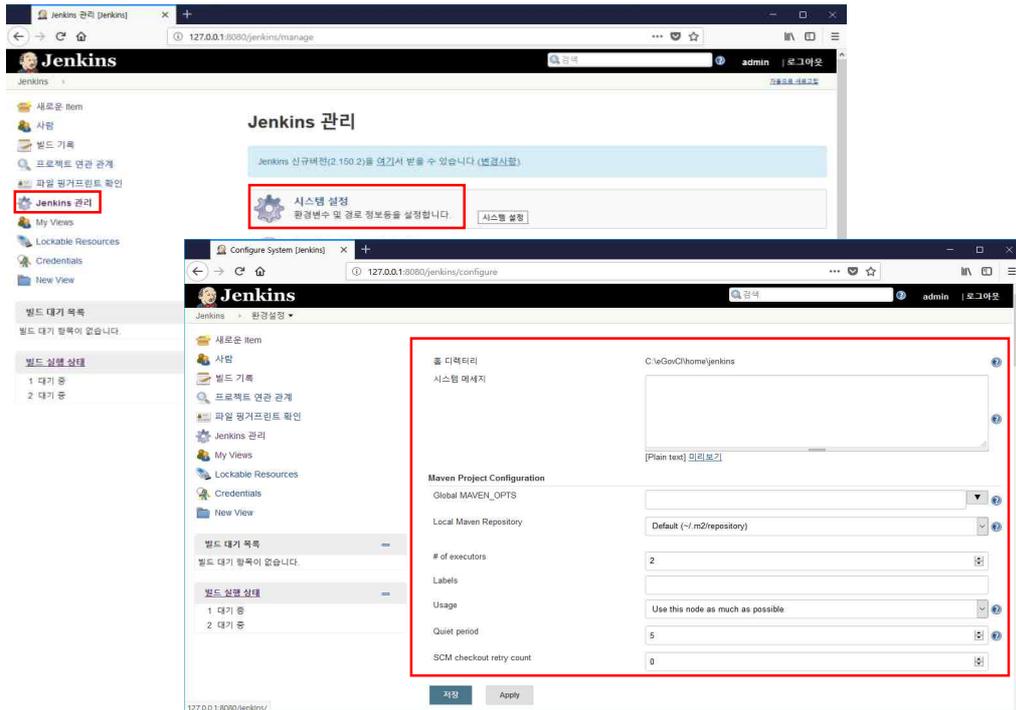
비밀번호 : <initialAdminPassword> 파일 확인



3.3.1 Jenkins 기본 시스템 설정

Jenkins의 새 작업 등록 전, Jenkins의 기본 시스템 설정한다. 설정을 통하여 JDK, ANT, Maven, SVN 서버 정보를 등록하여 프로젝트의 컴파일, Deploy을 할 수 있다.

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



아래와 같이 JDK, ANT, MAVEN home 디렉토리를 설정한다.



The screenshot shows the Jenkins configuration interface. On the left, a file explorer displays the local disk path 'C:\eGovCI\bin' with several folders. Three folders are highlighted with red boxes: 'jdk1.8.0_162', 'apache-ant-1.9.5', and 'apache-maven-3.3.3'. The main configuration area is divided into three sections: 'JDK installations', 'Ant installations', and 'Maven installations'. Each section has an 'Add' button and a form. In the 'JDK' section, the 'Name' is 'jdk1.8.0_162' and 'JAVA_HOME' is 'C:\eGovCI\bin\jdk1.8.0_162'. In the 'Ant' section, the 'Name' is 'apache-ant-1.9.5' and 'ANT_HOME' is 'C:\eGovCI\bin\apache-ant-1.9.5'. In the 'Maven' section, the 'Name' is 'apache-maven-3.3.3' and 'MAVEN_HOME' is 'C:\eGovCI\bin\apache-maven-3.3.3'. Each section also has an 'Install automatically' checkbox and a 'Delete' button.

3.3.2 새 작업 생성

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

The screenshot shows the Jenkins 'New Item' page in a web browser. The browser address bar shows '127.0.0.1:8080/jenkins/view/all/newJob'. The page title is 'New Item [Jenkins]'. The main content area has a header 'Enter an item name' with a text input field containing 'sample.buid'. Below this, there are four options: 'Freestyle project', 'Maven project', 'Pipeline', and 'Multi-configuration project'. The 'Maven project' option is selected and highlighted with a red box. At the bottom left, there is an 'OK' button, also highlighted with a red box.

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

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

소스 코드 관리

None
 Git
 Subversion

Modules

Repository URL:

Credentials:

Local module directory:

Repository depth:

Ignore externals:

Cancel process on externals fail:

Additional Credentials:

Check-out Strategy:

Use 'svn update' whenever possible, making the build faster. But this causes the artifacts from the previous build to remain when a new build starts.

Quiet check-out:

Repository browser:

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

빌드 유발

Build whenever a SNAPSHOT dependency is built
 Schedule build when some upstream has no successful builds

빌드를 원격으로 유발 (예: 스크립트 사용)
 Build after other projects are built
 Build periodically
 GitHub hook trigger for GITScm polling
 Poll SCM

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

빌드 환경

Delete workspace before build starts

고급...

Use secret text(s) or file(s) ?

Abort the build if it's stuck

Add timestamps to the Console Output

With Ant ?

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

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

Pre Steps

Add pre-build step ▾

Build

Root POM ?

Goals and options ?

MAVEN_OPTS ▾ ?

Incremental build - only build changed modules ?

Disable automatic artifact archiving ?

Disable automatic site documentation artifact archiving ?

Disable automatic fingerprinting of consumed and produced artifacts ?

Enable triggering of downstream projects ?

Block downstream trigger when building ?

Build modules in parallel ?

Use private Maven repository ?

Resolve Dependencies during Pom parsing ?

Run Headless ?

Process Plugins during Pom parsing ?

사용자 빌드 경로 사용 ?

Maven Validation Level ▾

Settings file ▾ ?

Global Settings file ▾ ?

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

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

Dashboard [Jenkins] x +

127.0.0.1:8080/jenkins/

Jenkins 1 검색 admin | 로그인

새로운 Item
사람
빌드 기록
프로젝트 연관 관계
파일 핑거프린트 확인
Jenkins 관리
My Views
Lockable Resources
Credentials
New View

빌드 대기 목록
빌드 대기 항목이 없습니다.

빌드 실행 상태
1 대기 중
2 대기 중

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

아이콘: S M L

Legend RSS 모두 RSS 실패 RSS 최근 빌드

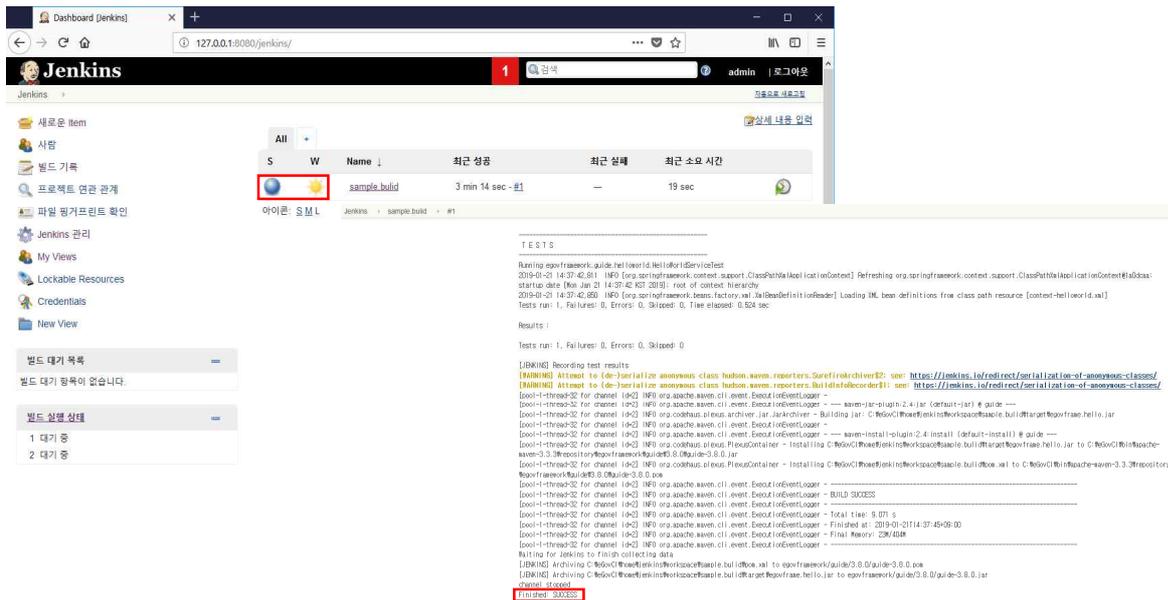
3.3.3 작업 Build

모든 설정 이후, 최초 수동으로 Build 하여 설정을 확인 할 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.

