



젠킨스

1. Jenkins란?



1) CI

지속적 통합(Continuous Integration) - 지속적으로 품질 관리를 적용하는 프로세스를 실행하는 것

2) Jenkins

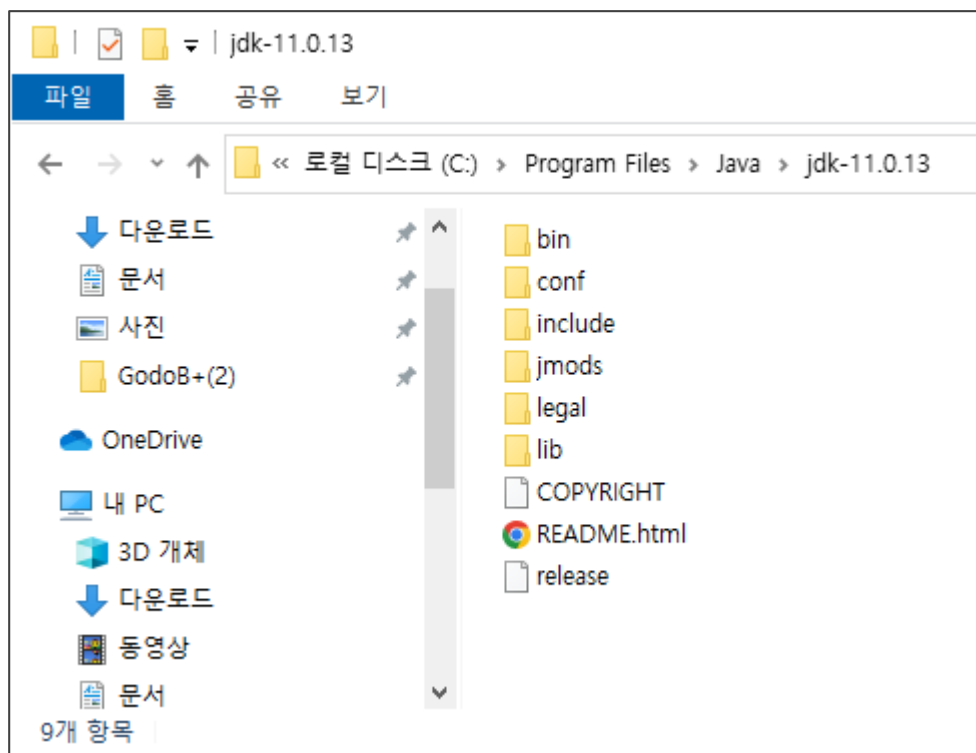
지속적 통합 서비스를 제공하는 툴(tool)로 허드슨(hudson) 프로젝트에서 시작

다수의 개발자들이 하나의 애플리케이션을 개발할 때
버전 충돌을 방지하기 위해서 github을 활용하는데
젠킨스는 github에 있는 애플리케이션을
자동으로 빌드(build)하고 톰캣 서버에 배포(deploy)해 줌

2. Java

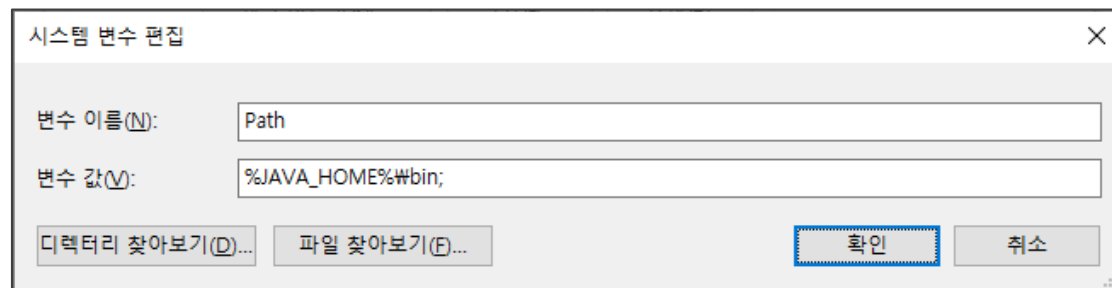
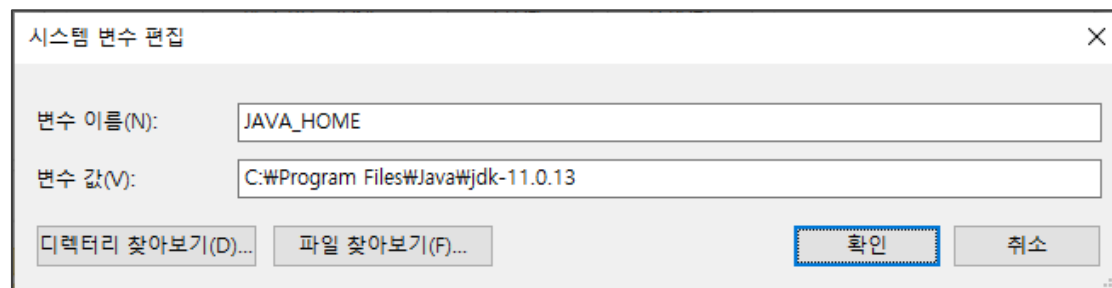
1) 자바 설치 경로 확인

C:\Program Files\Java\jdk-11.0.13



2) 자바 환경변수

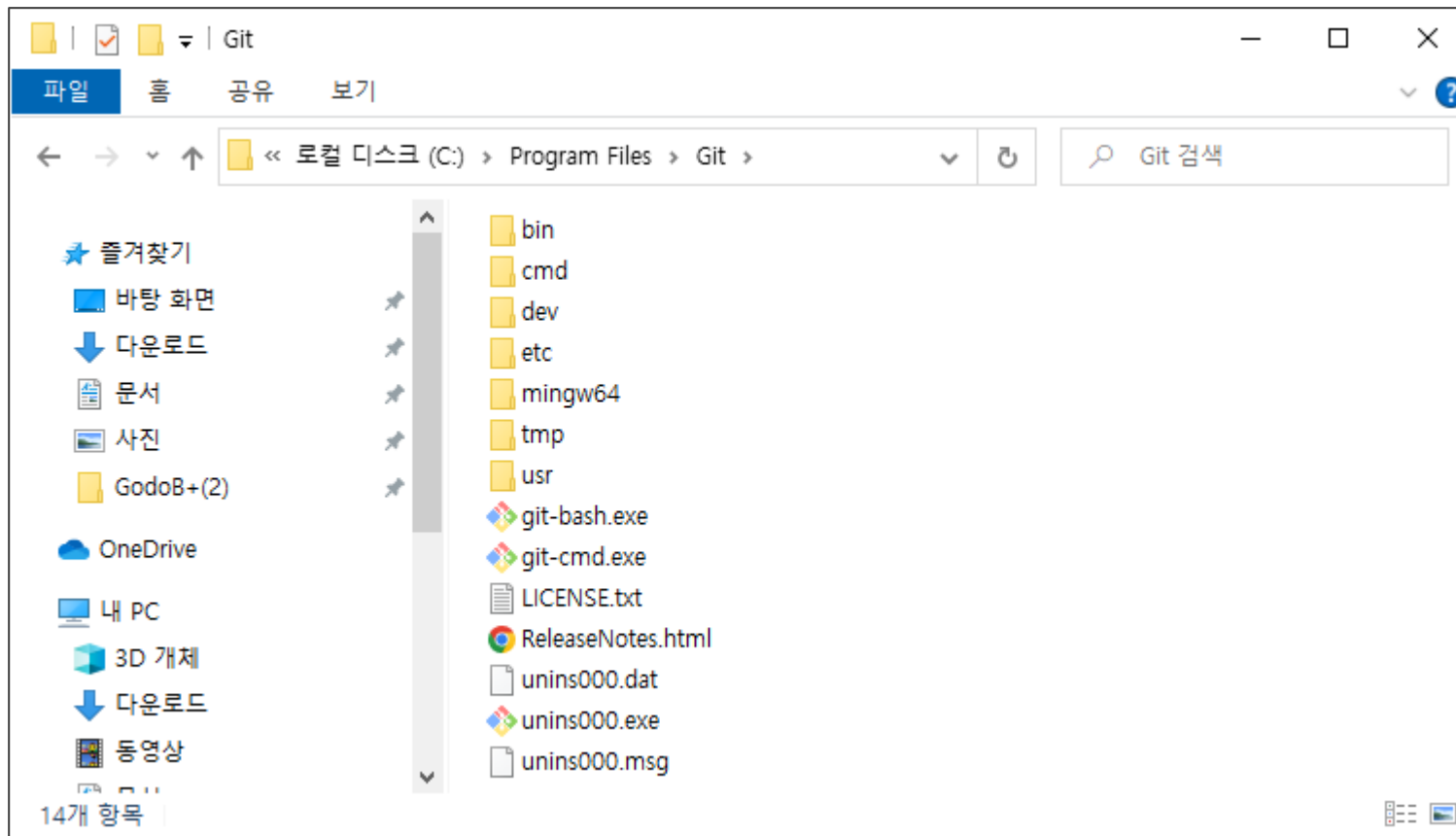
제어판 - 시스템



3. Git

1) Git 설치 경로 확인

C:\Program Files\Git



3. Git

2) Personal Access Token

github 로그인 - [계정 아이콘] - [Settings] - [Developer settings] - [Personal access token]

The screenshot illustrates the GitHub interface for generating a Personal Access Token. The navigation path is highlighted with red boxes and arrows:

- Profile icon**: Located in the top right corner of the GitHub header.
- Settings**: Located in the left sidebar under the 'Your profile' section.
- Developer settings**: Located in the left sidebar under the 'Settings' section.
- Personal access tokens**: Located in the 'Developer settings' section.
- Generate new token**: A button located next to the 'Personal access tokens' section.
- Expiration**: A dropdown menu set to '30 days'.
- Select scopes**: A section where you can choose the permissions for the token. The 'repo' scope is selected.
- Generate token**: A green button located at the bottom right of the 'Select scopes' section.

Additional details visible in the interface:

- Repositories**: A list of repositories on the left sidebar.
- Security**: A section with 'Code security and analysis'.
- Integrations**: A section with 'Applications' and 'Scheduled reminders'.
- Archives**: A section with 'Security log' and 'Sponsorship log'.
- Full control of private repositories**: A list of scopes including 'repo:status', 'repo_deployment', 'public_repo', 'repo:invite', and 'security_events'.
- Access commit status**: A label for the 'repo:status' scope.
- Access deployment status**: A label for the 'repo_deployment' scope.
- Access public repositories**: A label for the 'public_repo' scope.
- Access repository invitations**: A label for the 'repo:invite' scope.
- Read and write security events**: A label for the 'security_events' scope.

※ 발급 후 백업해야 함!

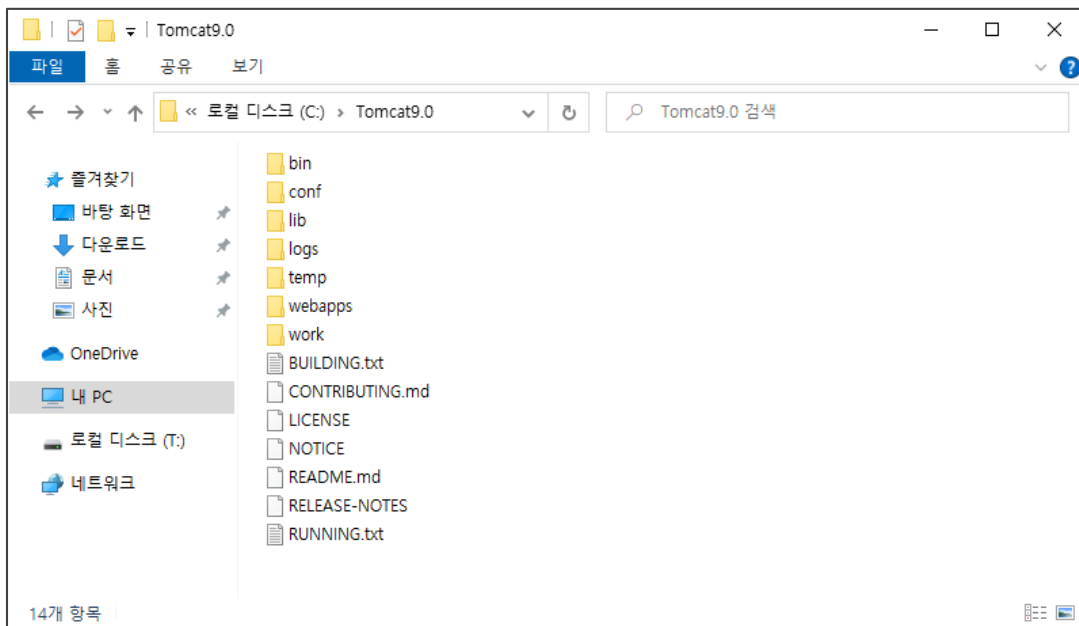
4. Tomcat

1) 톰캣 다운로드 (버전 9.0.x)

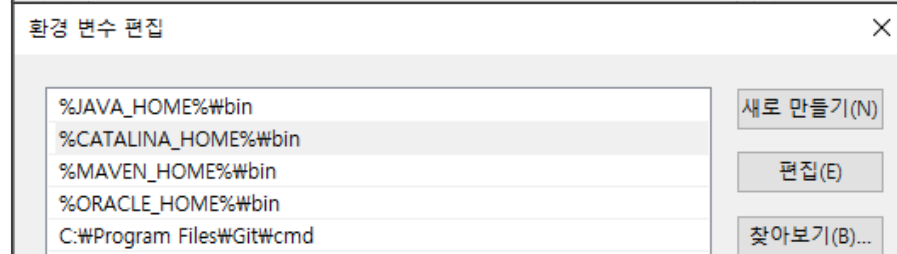
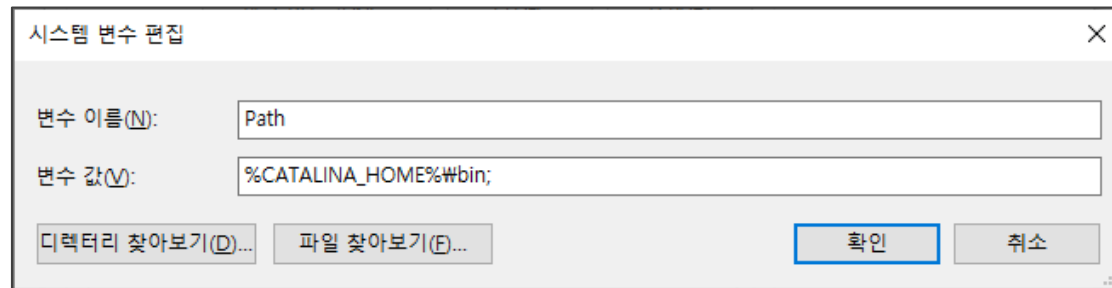
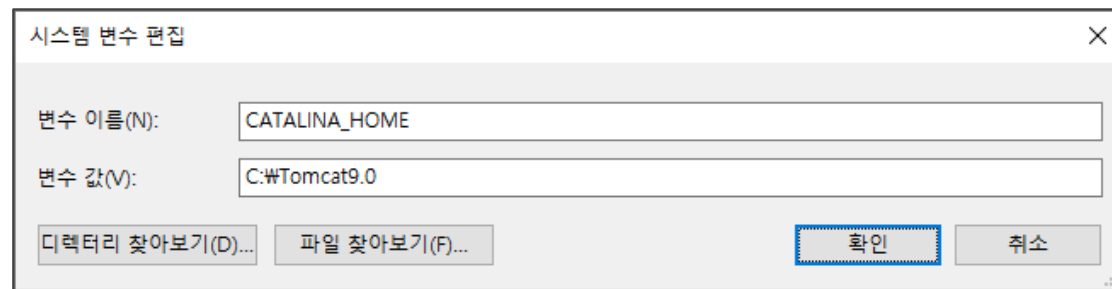
<https://tomcat.apache.org/>

2) 톰캣 설치

C:\Tomcat9.0



3) 톰캣 환경변수 제어판 - 시스템



4. Tomcat

4) 포트번호 수정 (Oracle DB인 경우)

C:\Tomcat\conf\server.xml



```
server.xml - Windows 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
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="9090" 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" />
-->
```

태그 내용 대신 주석을 수정하지 않도록 주의!

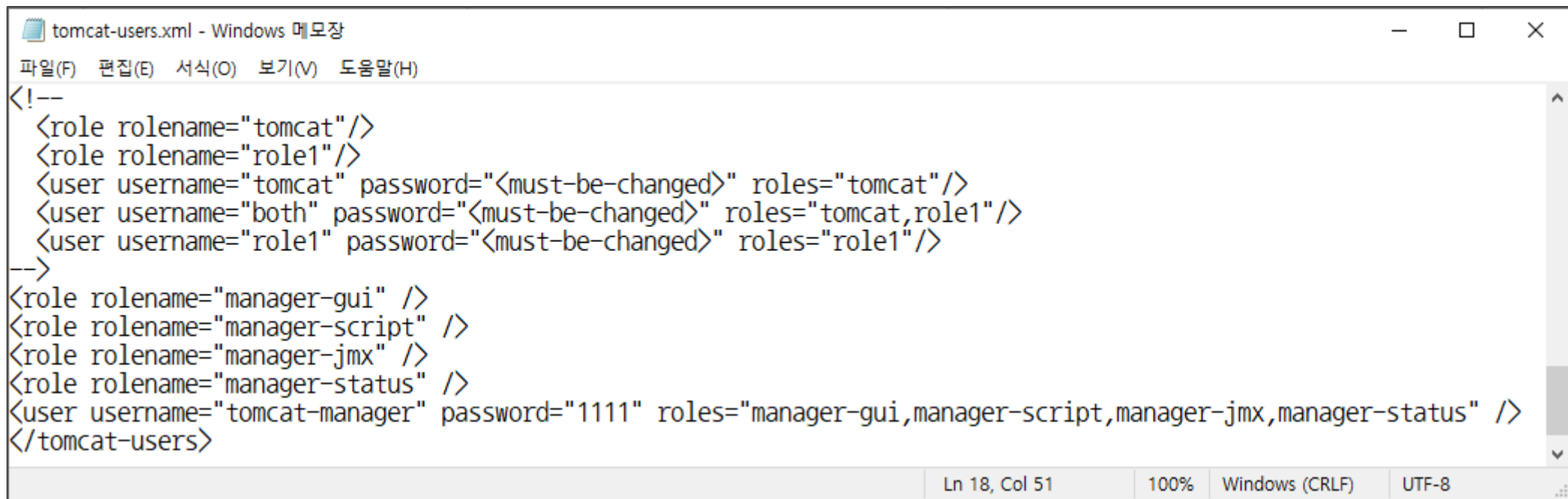
4. Tomcat

5) 톰캣 관리자 등록

C:\WTomcat\conf\tomcat-users.xml

관리자명 : tomcat-manager

비밀번호 : 1111



```
tomcat-users.xml - Windows 메모장
파일(F)  편집(E)  서식(O)  보기(V)  도움말(H)
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
  <user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
  <user username="role1" password="<must-be-changed>" roles="role1"/>
-->
<role rolename="manager-gui" />
<role rolename="manager-script" />
<role rolename="manager-jmx" />
<role rolename="manager-status" />
<user username="tomcat-manager" password="1111" roles="manager-gui,manager-script,manager-jmx,manager-status" />
</tomcat-users>
```

Ln 18, Col 51 100% Windows (CRLF) UTF-8

5. Maven

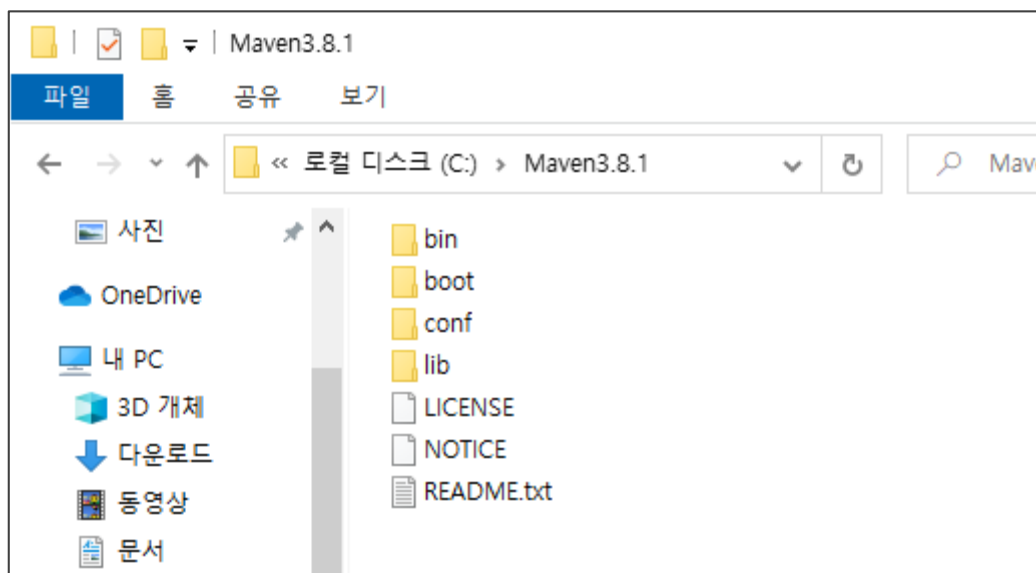
1) 메이븐 다운로드 (버전 3.8.1)

<https://maven.apache.org/>

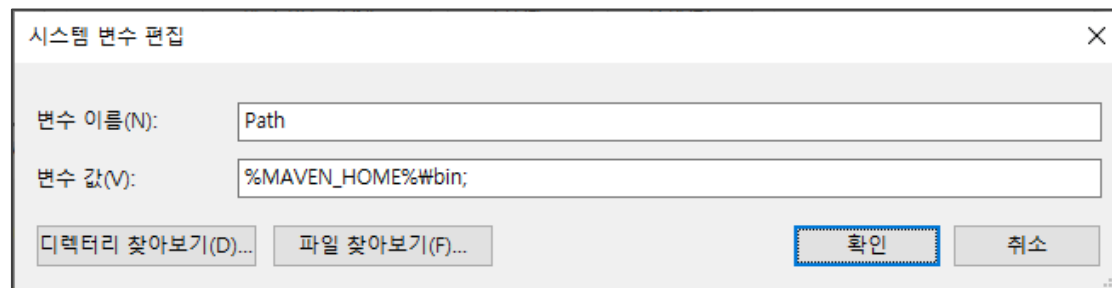
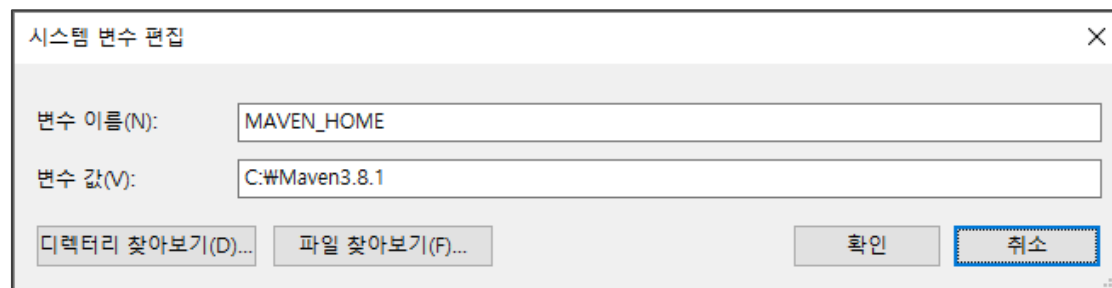
* mvnrepository.com에 있는 버전을 사용해야 함

2) 메이븐 설치

C:\Maven3.8.1



3) 메이븐 환경변수 제어판 - 시스템



6. Application

pom.xml

```
<groupId>com.company</groupId>
<artifactId>app</artifactId>
<name>project</name>
<packaging>war</packaging>
<version>1.0.0</version>
```

```
<properties>
  <java-version>11</java-version>
  <org.springframework-version>5.3.3</org.springframework-version>
  <org.aspectj-version>1.9.6</org.aspectj-version>
  <org.slf4j-version>1.7.30</org.slf4j-version>
  <maven-version>3.8.1</maven-version>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
</properties>
```

설치한 Maven 버전과 젠킨스 빌드에서 사용할 Encoding

빌드 결과로 app-1.0.0.war 파일이 생성됨.
파일명의 버전을 없애려면 finalName 태그를 추가함.
finalName 태그가 추가되면 app.war 파일이 생성됨.

```
<build>
  <finalName>${artifactId}</finalName>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>${maven-version}</version>
      <configuration>
        <source>${java-version}</source>
        <target>${java-version}</target>
        <compilerArgument>-Xlint:all</compilerArgument>
        <showWarnings>true</showWarnings>
        <showDeprecation>true</showDeprecation>
      </configuration>
    </plugin>
  </plugins>
</build>
```

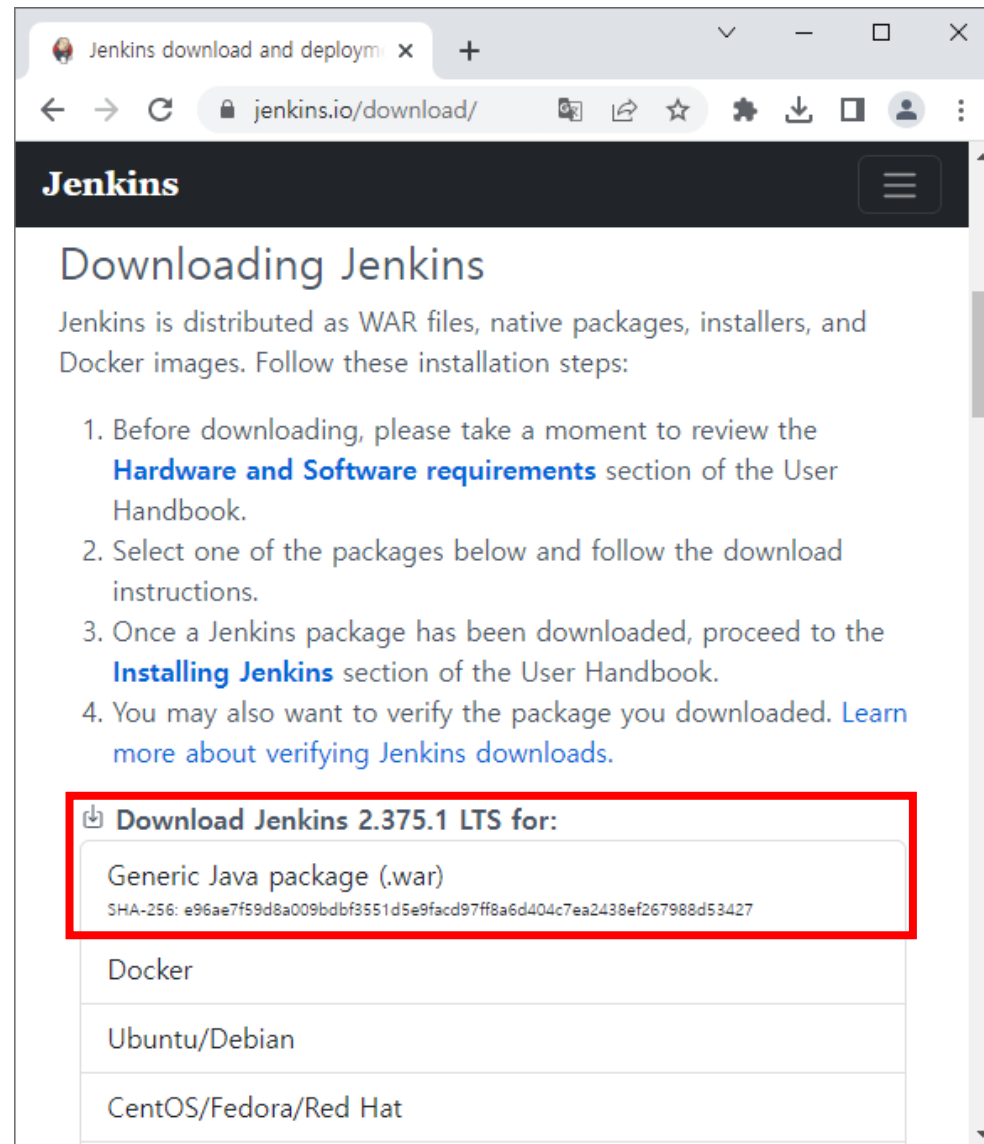
7. Jenkins Start

1) 젠킨스 다운로드

<https://www.jenkins.io/download>

LTS 버전

Generic Java Package(.war)



The screenshot shows the Jenkins download page in a web browser. The browser's address bar displays 'jenkins.io/download/'. The page has a dark header with the 'Jenkins' logo. The main heading is 'Downloading Jenkins', followed by a paragraph stating that Jenkins is distributed as WAR files, native packages, installers, and Docker images, and that users should follow the installation steps. A numbered list of four steps is provided. Below the list, a section titled 'Download Jenkins 2.375.1 LTS for:' is highlighted with a red rectangle. This section contains a table with three rows: 'Generic Java package (.war)' (which includes a SHA-256 hash), 'Docker', and 'Ubuntu/Debian'. Below these, 'CentOS/Fedora/Red Hat' is listed as an option.

Jenkins download and deployment x +

jenkins.io/download/

Jenkins

Downloading Jenkins

Jenkins is distributed as WAR files, native packages, installers, and Docker images. Follow these installation steps:

1. Before downloading, please take a moment to review the [Hardware and Software requirements](#) section of the User Handbook.
2. Select one of the packages below and follow the download instructions.
3. Once a Jenkins package has been downloaded, proceed to the [Installing Jenkins](#) section of the User Handbook.
4. You may also want to verify the package you downloaded. [Learn more about verifying Jenkins downloads.](#)

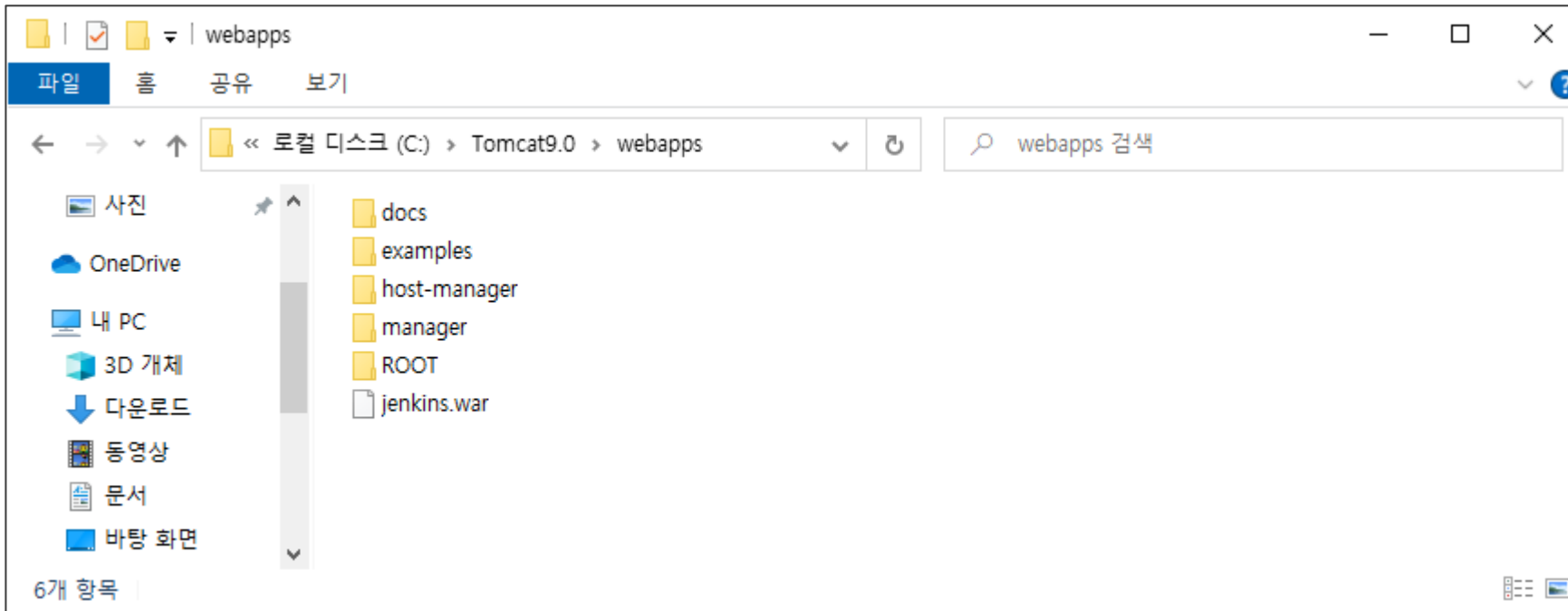
Download Jenkins 2.375.1 LTS for:

Generic Java package (.war)
SHA-256: e96ae7f59d8a009bdbf3551d5e9facd97ff8a6d404c7ea2438ef267988d53427
Docker
Ubuntu/Debian
CentOS/Fedora/Red Hat

7. Jenkins Start

2) 젠킨스 배포

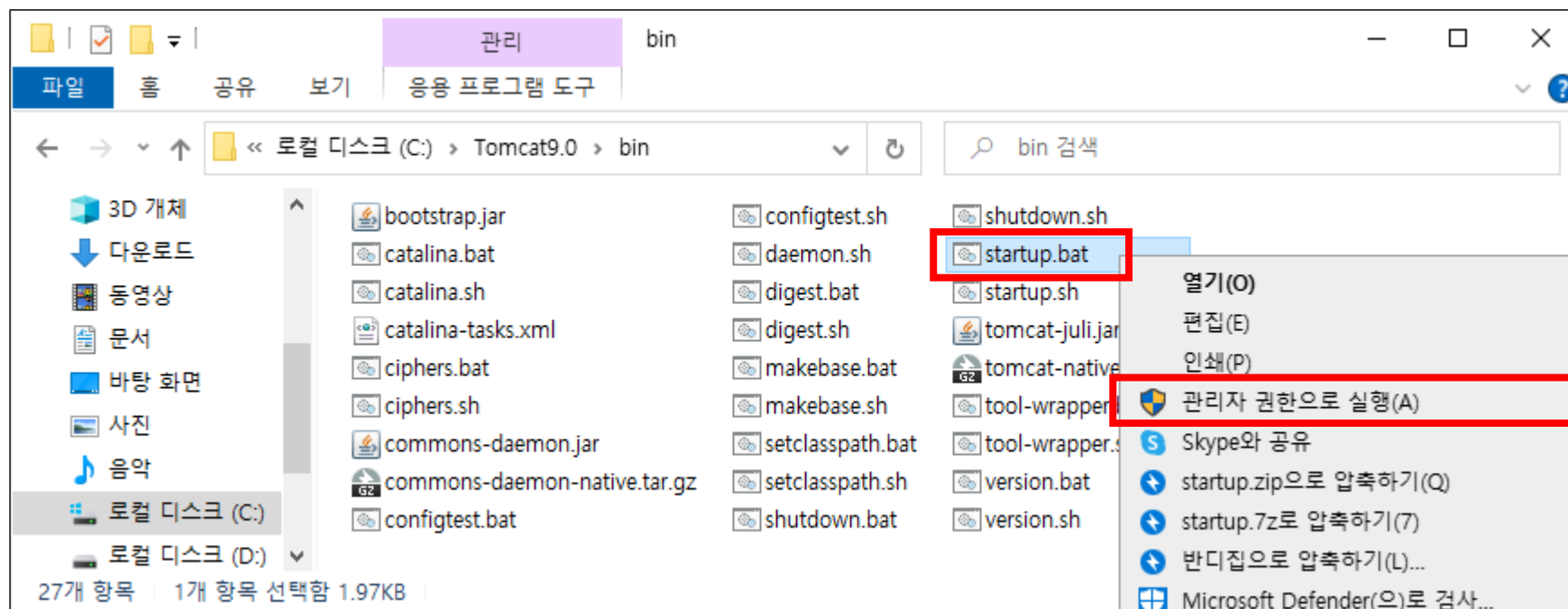
C:\Tomcat9.0\webapps 디렉터리에
다운로드 받은 jenkins.war 파일을 복사해 둬



7. Jenkins Start

3) 톰캣 실행

C:\Tomcat9.0\bin\startup.bat 파일을
[관리자 권한으로 실행]

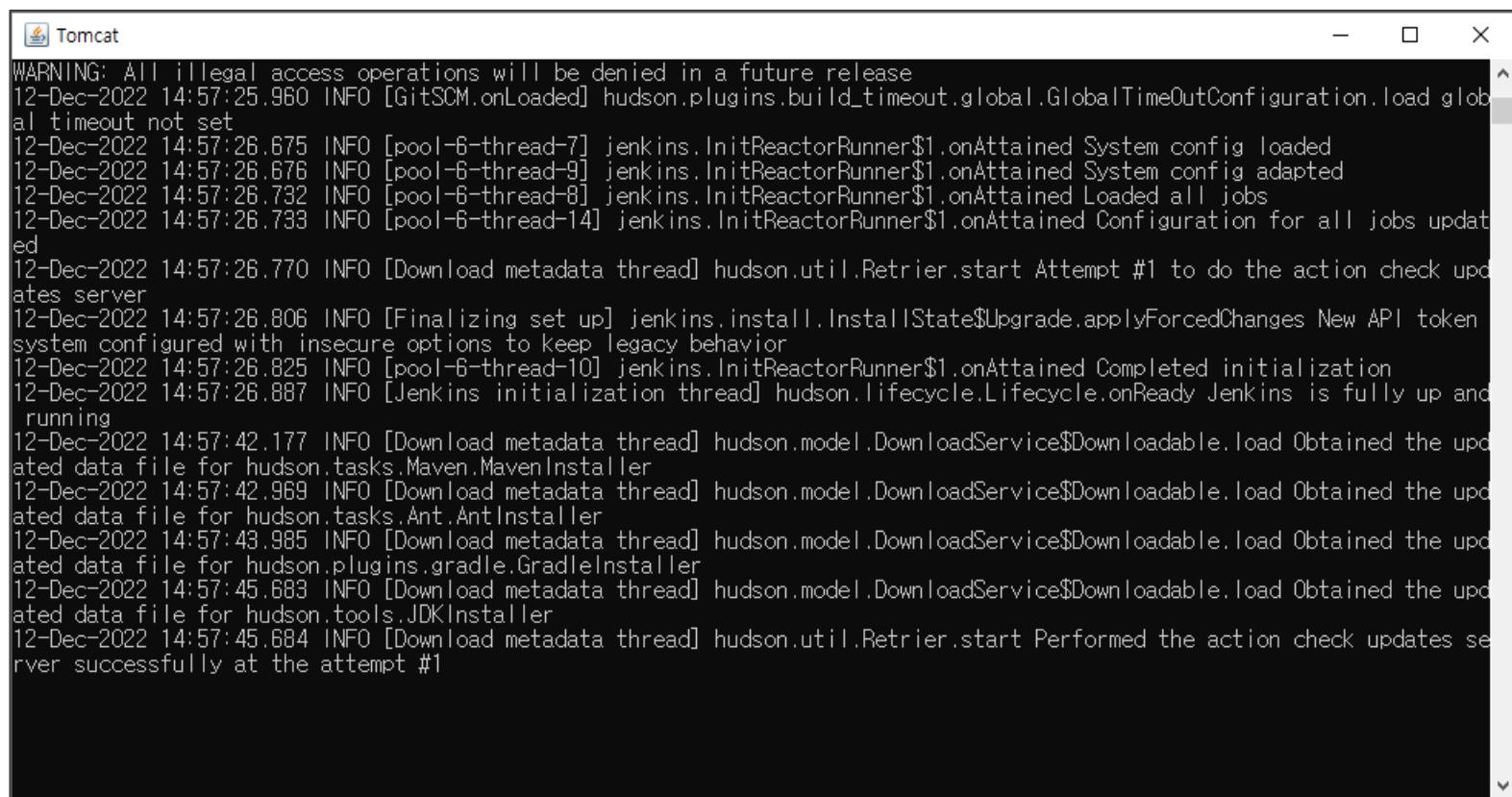
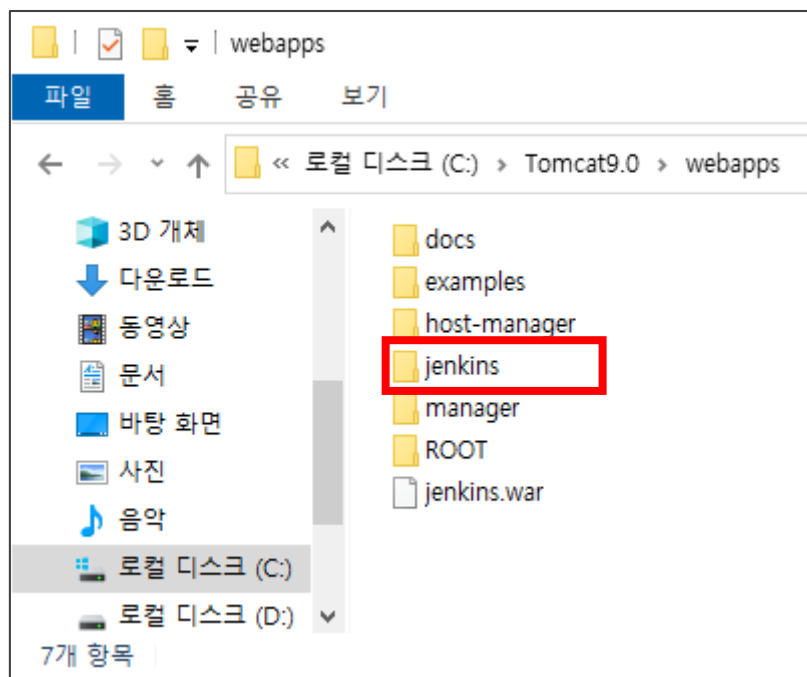


* 리눅스에서는 startup.sh 파일 실행

7. Jenkins Start

4) 톰캣 실행

webapps/jenkins.war -> webapps/jenkins 디렉터리로 압축이 풀림



고지 말 것!

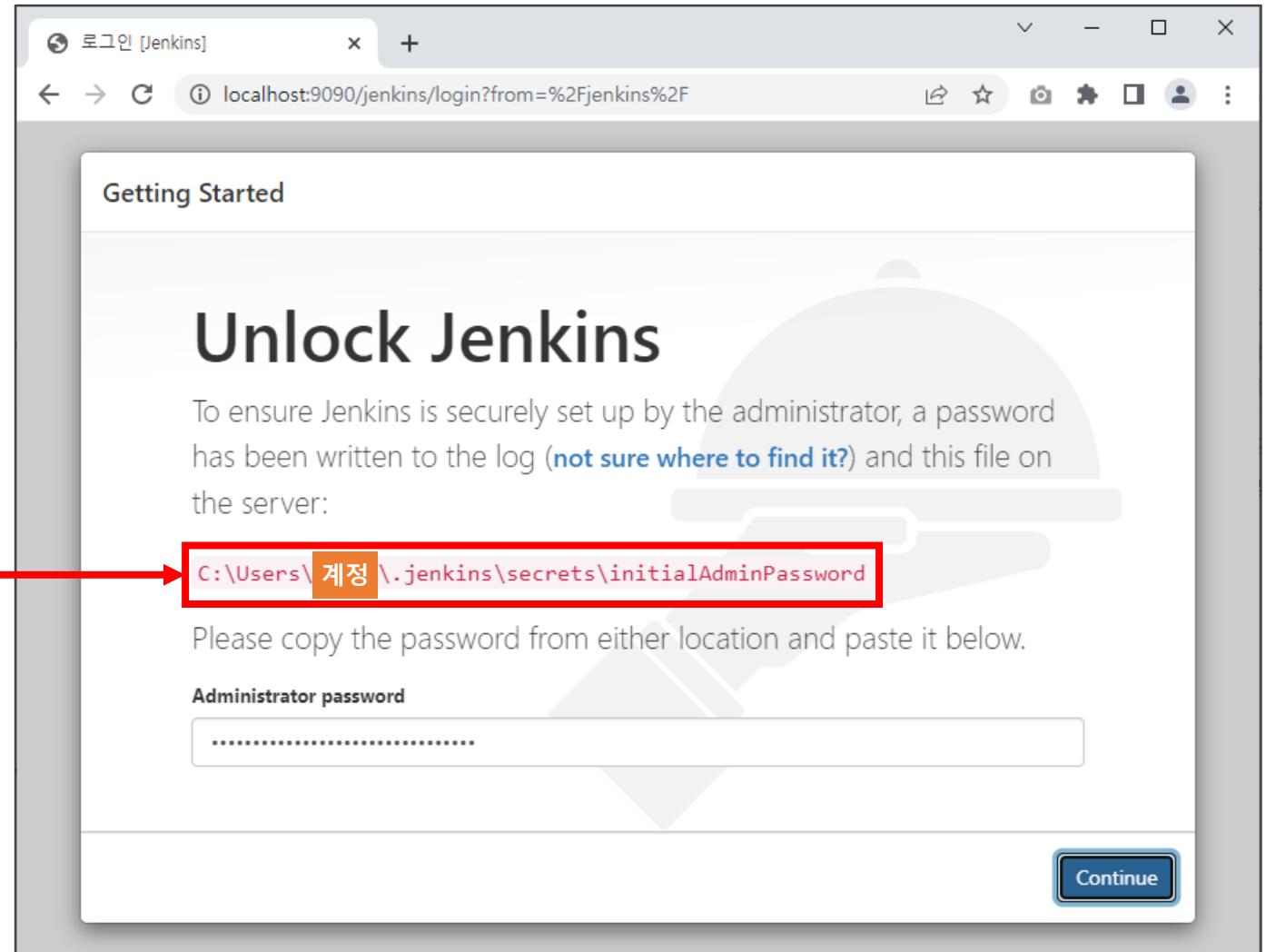


7. Jenkins Start

4) 젠킨스 실행

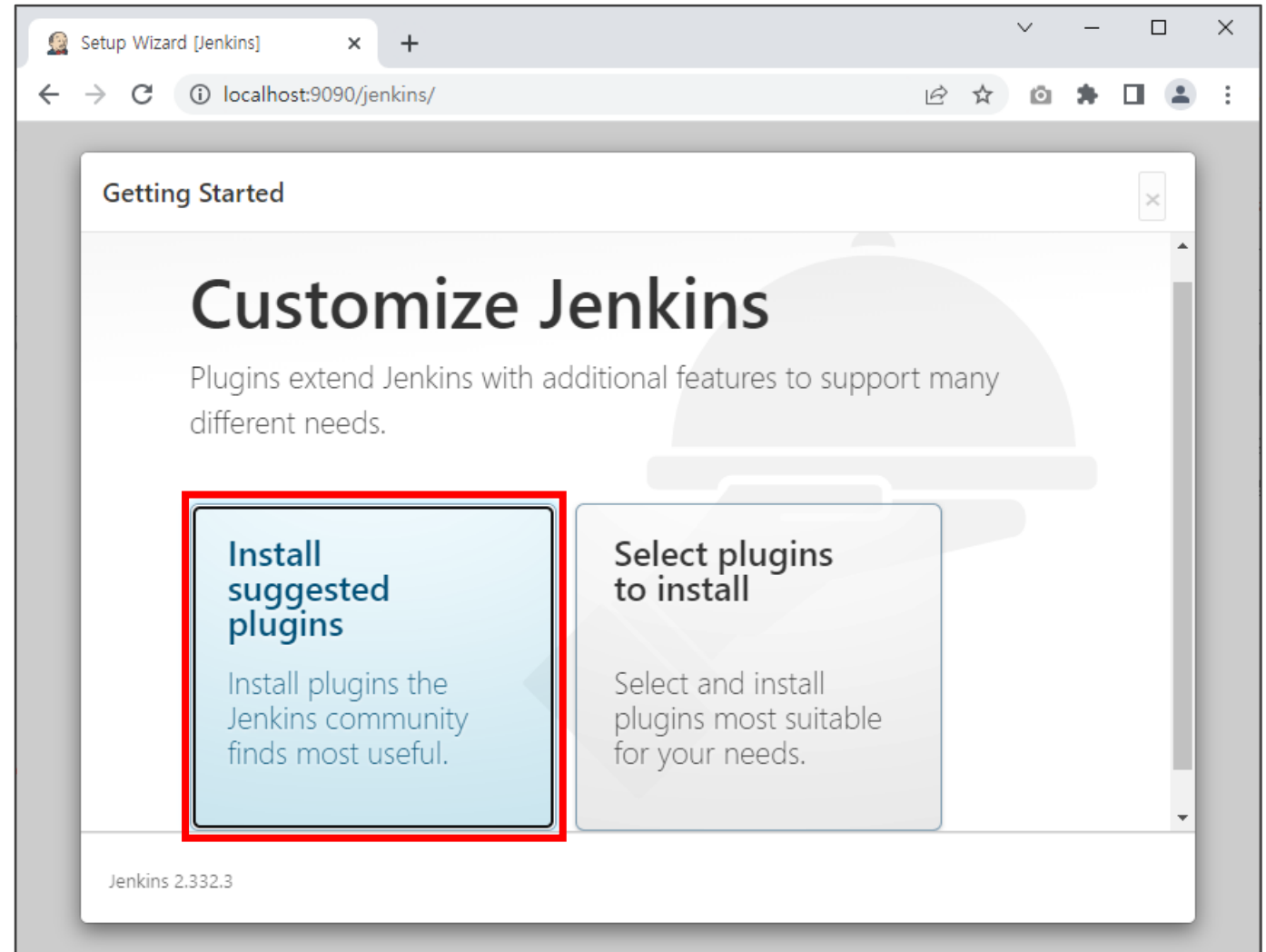
http://localhost:9090/jenkins 접속

초기비밀번호 입력
(초기비밀번호 위치)



7. Jenkins Start

5) 젠킨스 플러그인 설치 Install suggested plugins



7. Jenkins Start

6) 젠킨스 관리자 등록 관리자 등록

* Skip and continue as admin
관리자 등록을 생략하는 경우
admin 계정과 초기비밀번호를
이용해서 계속 로그인 가능

Setup Wizard [Jenkins]

localhost:9090/jenkins/

Getting Started

Create First Admin User

계정명:

암호:

암호 확인:

이름:

이메일 주소:

정보 등록

Jenkins 2.332.3

[Skip and continue as admin](#)

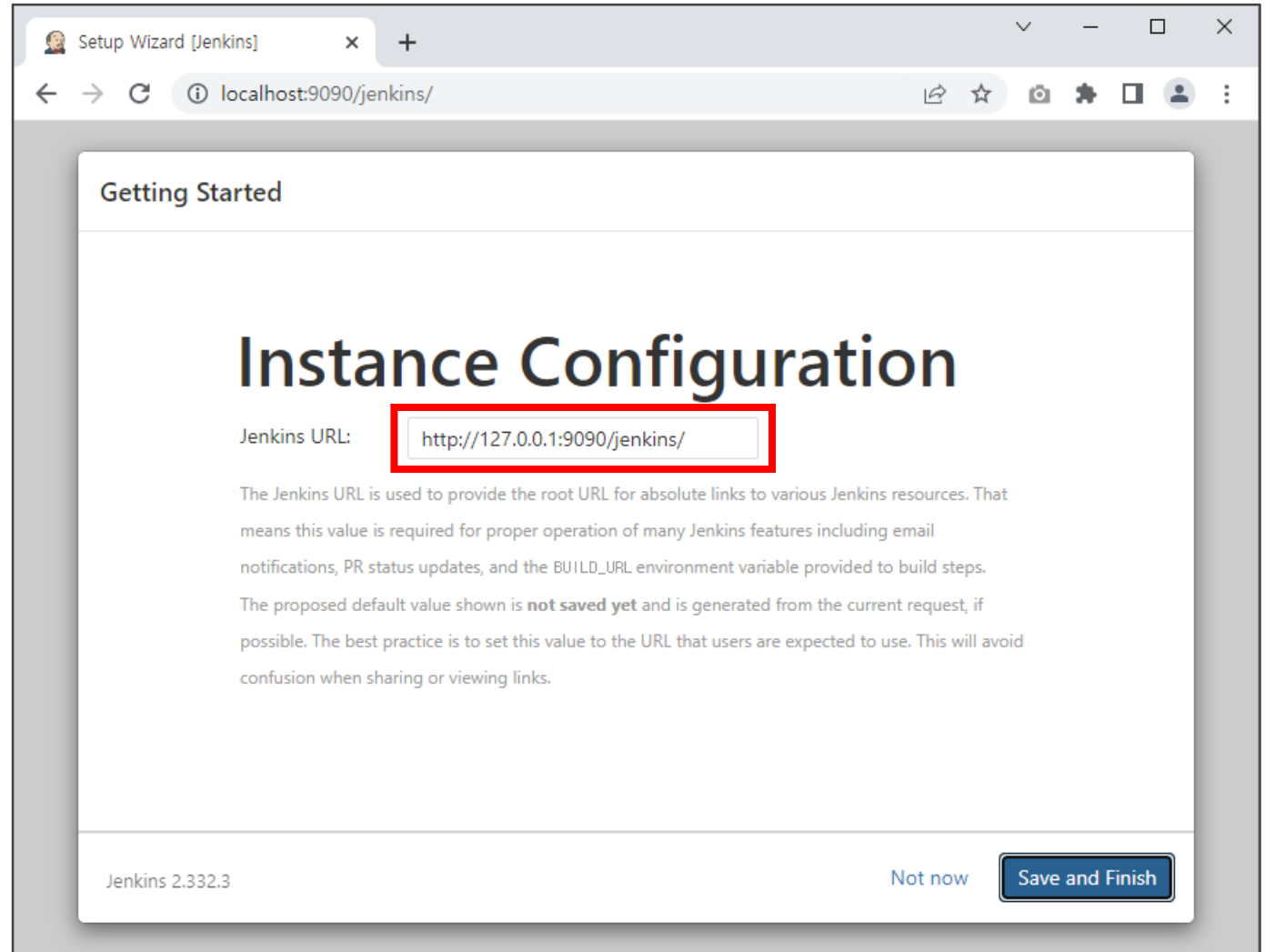
[Save and Continue](#)

7. Jenkins Start

7) 젠킨스 주소 등록

http://127.0.0.1:9090/jenkins/

* 젠킨스 주소는 localhost 사용 불가



Setup Wizard [Jenkins]

localhost:9090/jenkins/

Getting Started

Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD_URL environment variable provided to build steps.

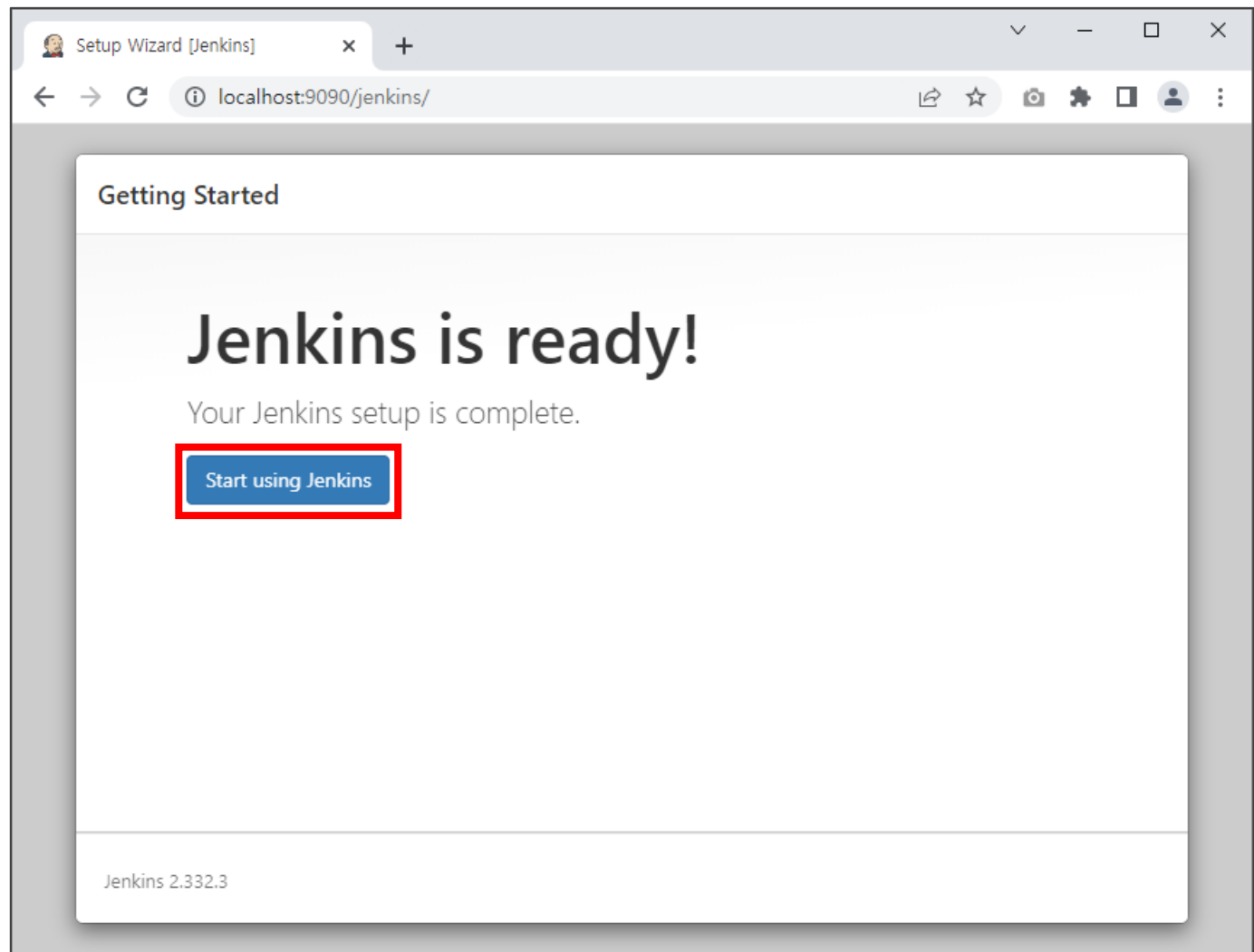
The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Jenkins 2.332.3

Not now **Save and Finish**

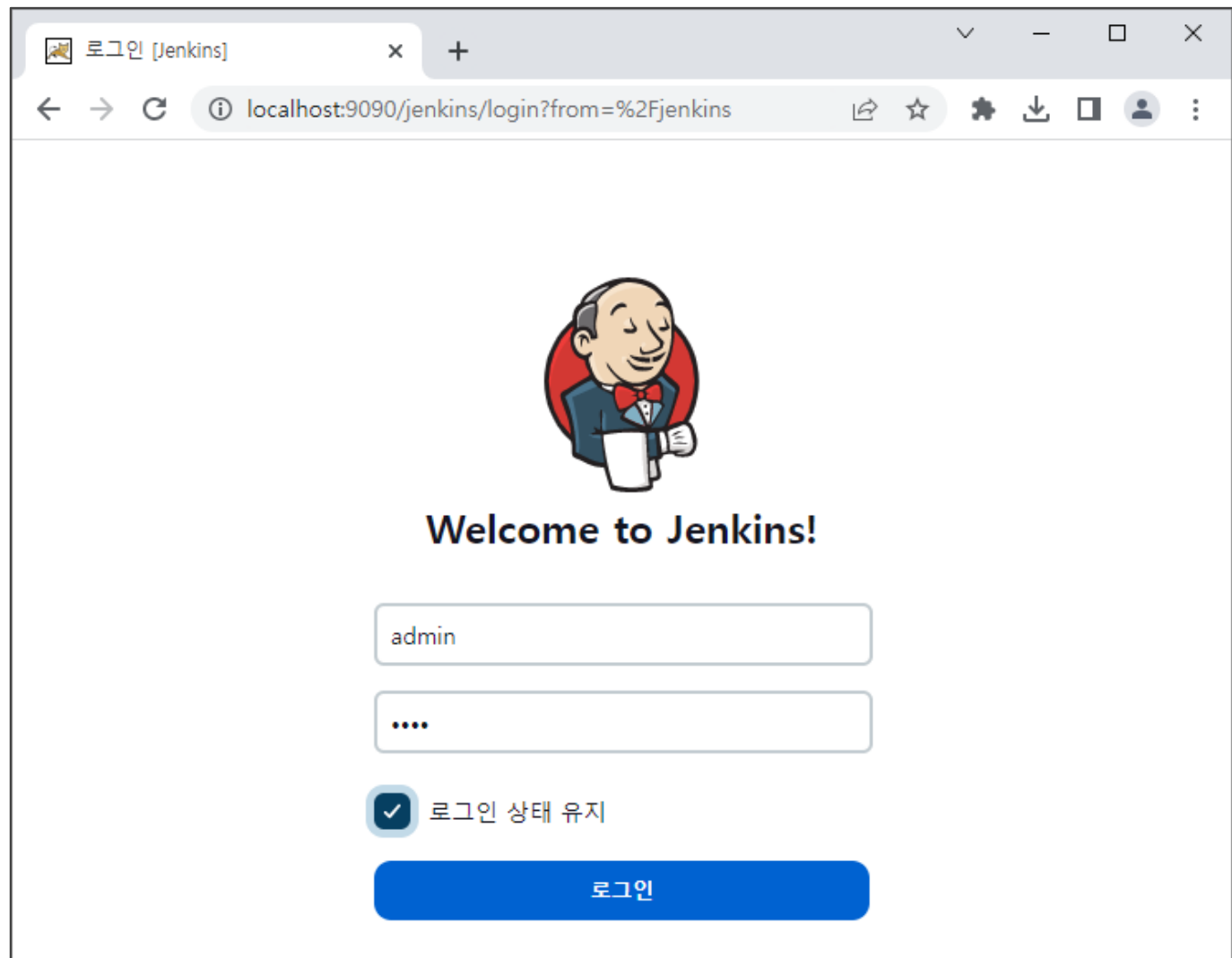
7. Jenkins Start

8) 젠킨스 준비 완료



7. Jenkins Start

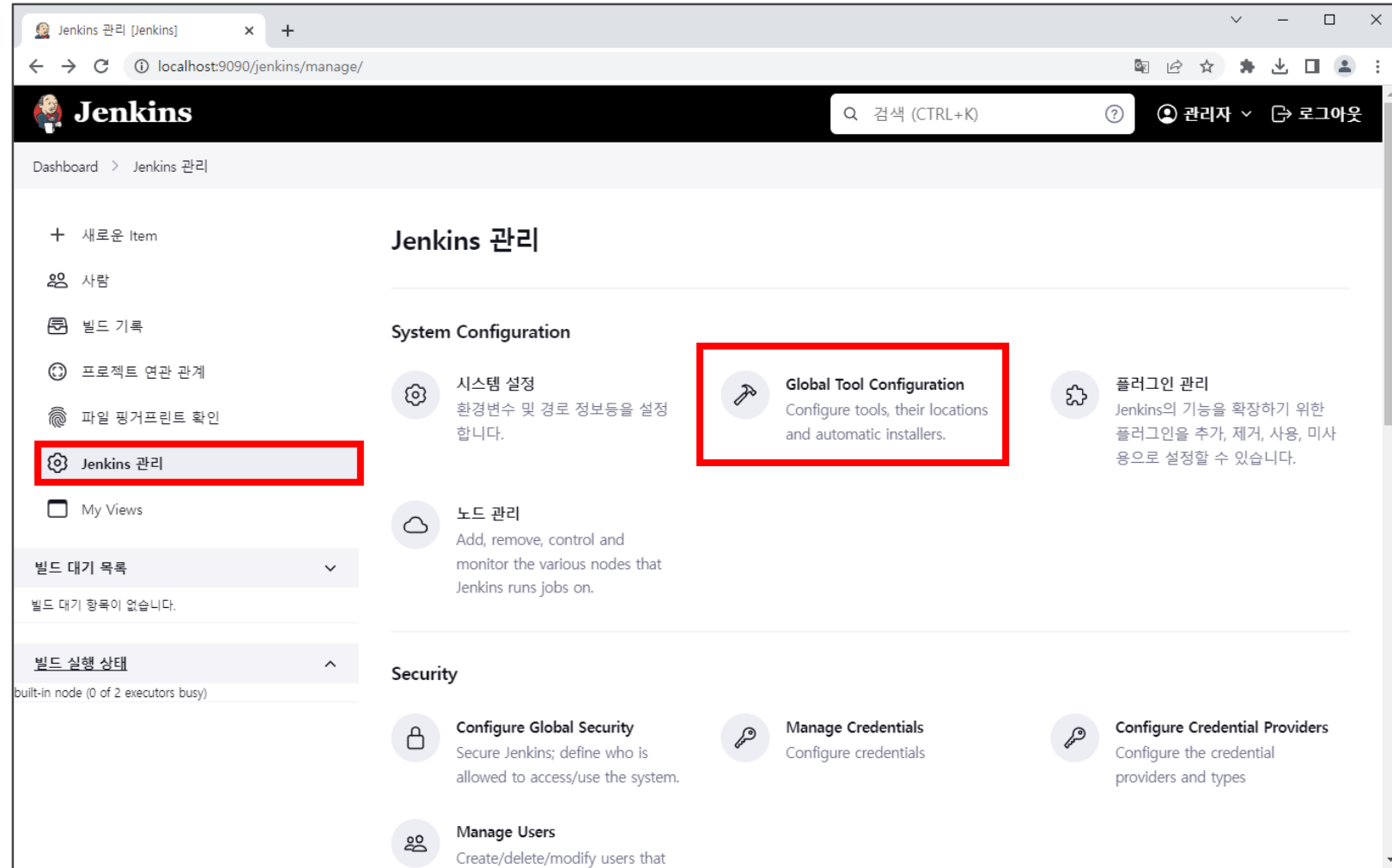
9) 첫 로그인 이후 나타나는 로그인 화면



8. Jenkins Setting

1) [Jenkins 관리]

└ [Global Tool Configuration]



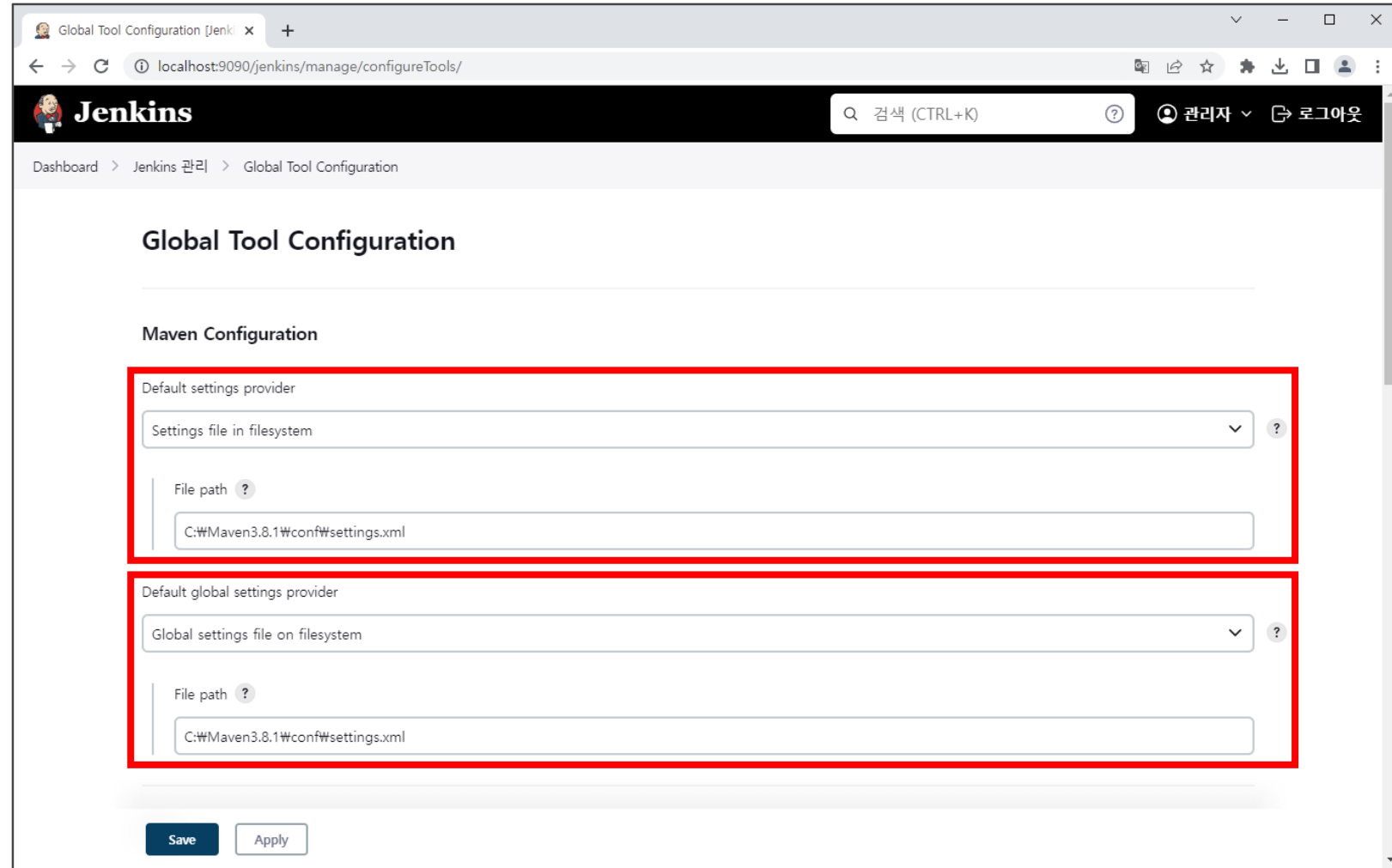
8. Jenkins Setting

1) [Jenkins 관리]

└ [Global Tool Configuration]

└ [Maven Configuration]

- Maven 환경설정파일을 등록하는 작업임



Global Tool Configuration [Jenki x +]

localhost:9090/jenkins/manage/configureTools/

Jenkins

검색 (CTRL+K) 관리자 로그아웃

Dashboard > Jenkins 관리 > Global Tool Configuration

Global Tool Configuration

Maven Configuration

Default settings provider

Settings file in filesystem

File path ?

C:\Maven3.8.1\conf\settings.xml

Default global settings provider

Global settings file on filesystem

File path ?

C:\Maven3.8.1\conf\settings.xml

Save Apply

8. Jenkins Setting

1) [Jenkins 관리]

└ [Global Tool Configuration]

└ [JDK]

- Java가 설치된 경로를 등록하는 작업임

The screenshot shows the Jenkins web interface for configuring global tools. The browser address bar indicates the URL is `localhost:9090/jenkins/manage/configureTools/`. The breadcrumb navigation shows the path: `Dashboard > Jenkins 관리 > Global Tool Configuration`.

The main section is titled **JDK** and contains the following elements:

- JDK installations**: A heading for the configuration section.
- List of JDK installations on this system**: A sub-heading.
- Add JDK**: A button to add a new JDK installation, highlighted with a red box.
- JDK Configuration Form**: A dashed box containing the configuration details for a new JDK:
 - Name**: A text input field containing `MyJDK`.
 - JAVA_HOME**: A text input field containing `C:\Program Files\Java\jdk-11.0.13`.
 - Install automatically**: A checkbox that is currently unchecked, with a red box around it and the Korean text **해제함** (Uncheck) next to it.
- Add JDK**: A button at the bottom of the configuration form.

At the bottom of the page, there are **Save** and **Apply** buttons.

8. Jenkins Setting

1) [Jenkins 관리]

└ [Global Tool Configuration]

└ [Git]

- Git이 설치된 경로를 등록하는 작업임

The screenshot shows the Jenkins web interface for configuring global tools. The browser address bar indicates the URL is `localhost:9090/jenkins/manage/configureTools/`. The breadcrumb navigation shows the path: `Dashboard > Jenkins 관리 > Global Tool Configuration`.

The main section is titled **Git** and contains a sub-section for **Git installations**. A new installation entry is being added, highlighted with a red box. This entry has the following fields:

- Name:** `MyGit`
- Path to Git executable:** `C:\Program Files\Git\bin\git.exe`
- Install automatically:** An unchecked checkbox.

Below the installation entry is a button labeled **Add Git**, also highlighted with a red box. To the right of the 'Install automatically' checkbox, the Korean text **해제함** (Uninstall) is visible.

Below the Git section, the **Gradle** section is partially visible, showing a heading and a sub-section for **Gradle installations**. At the bottom of the page, there are **Save** and **Apply** buttons.

8. Jenkins Setting

1) [Jenkins 관리]

└ [Global Tool Configuration]

└ [Maven]

- Maven이 설치된 경로를 등록하는 작업임

Global Tool Configuration [Jenki x +

localhost:9090/jenkins/manage/configureTools/

Dashboard > Jenkins 관리 > Global Tool Configuration

Maven

Maven installations

List of Maven installations on this system

Add Maven

Maven Name

MyMaven

MAVEN_HOME

C:\Maven3.8.1

☐ Install automatically ? **해제함**

Add Maven

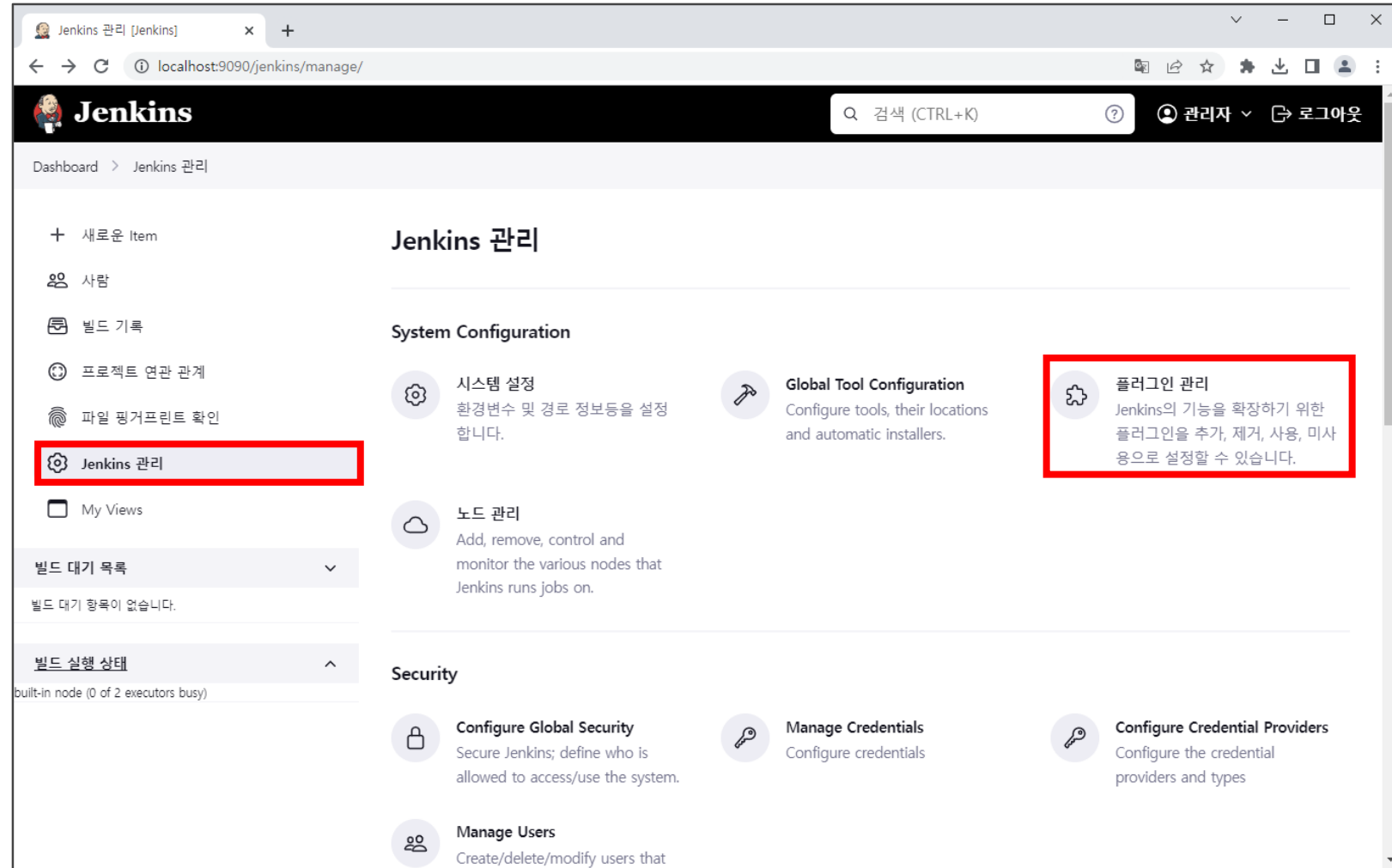
Save **Apply**

Jenkins 2.375.1

8. Jenkins Setting

1) [Jenkins 관리]

↳ [플러그인 관리]



8. Jenkins Setting

1) [Jenkins 관리]

└ [플러그인 관리]

└ 설치 가능

└ deploy 검색

└ Deploy to container 선택

└ Install without restart

The screenshot shows the Jenkins Plugin Manager interface. The left sidebar has a menu with 'Available plugins' highlighted. The main area shows a search bar with 'deploy' entered. Below the search bar, a table lists available plugins. The first plugin, 'Deploy to container 1.16', is selected with a checked checkbox. The table also shows 'Docker Pipeline' and 'Artifactory' as other available plugins. At the bottom, there are buttons for 'Install without restart' and 'Download now and install after restart'.

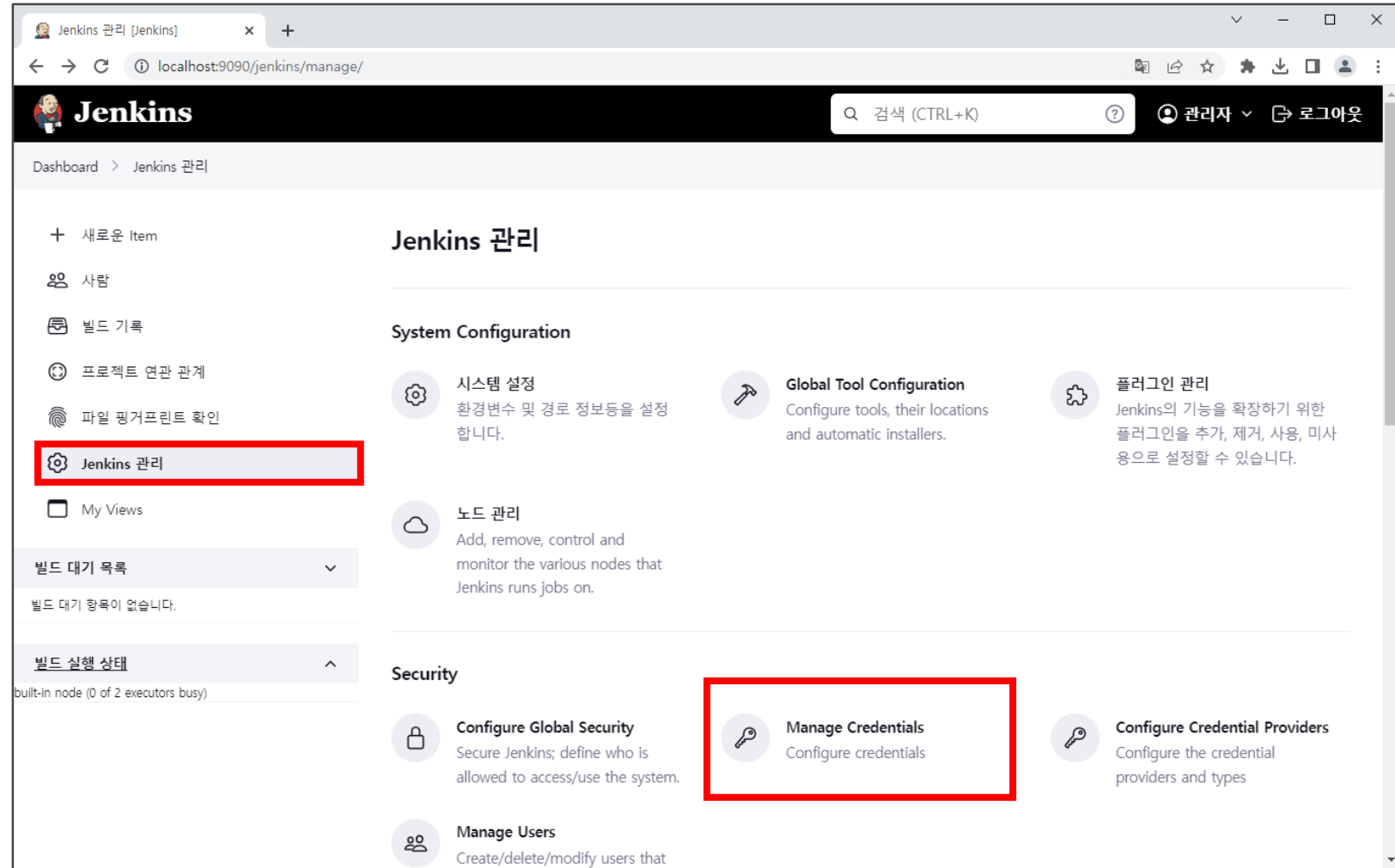
Install	Name ↓	Released
<input checked="" type="checkbox"/>	Deploy to container 1.16 Artifact Uploaders This plugin allows you to deploy a war to a container after a successful build. Glassfish 3.x remote deployment	2 yr 1 mo ago
<input type="checkbox"/>	Docker Pipeline 563.vd5d2e5c4007f pipeline DevOps Deployment docker Build and use Docker containers from pipelines.	10 days ago
<input type="checkbox"/>	Artifactory 3.17.4 pipeline This plugin allows your build jobs to deploy artifacts and resolve dependencies to and from Artifactory, and then have them linked to the build job that created them. The plugin includes a vast collection of features, including a rich pipeline API library and release management for Maven and Gradle builds with Staging and Promotion.	10 days ago

Buttons at the bottom: **Install without restart**, **Download now and install after restart**, Update information obtained: 19 min ago, 지금 확인

8. Jenkins Setting

1) [Jenkins 관리]

└ [Manage Credentials]



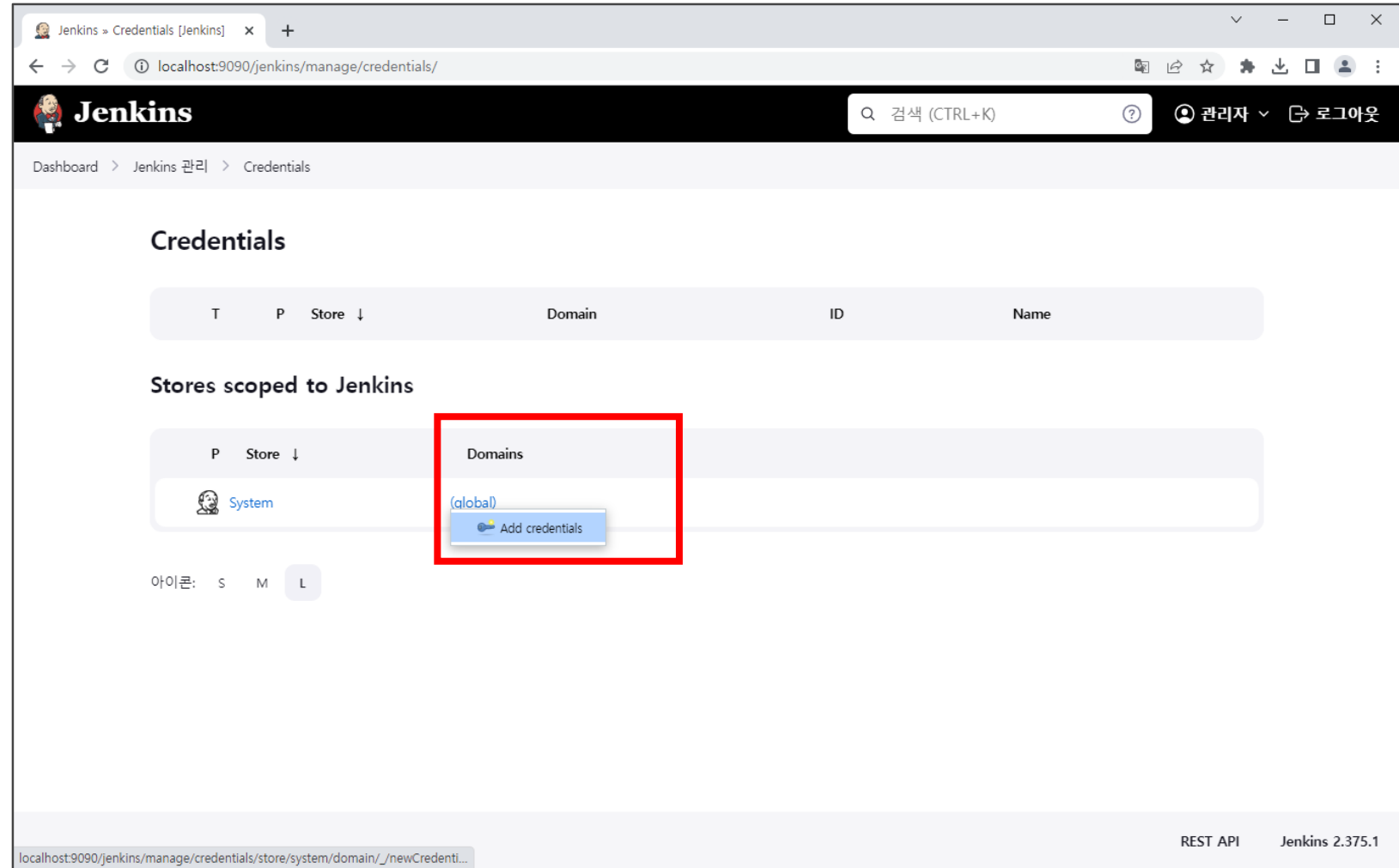
8. Jenkins Setting

1) [Jenkins 관리]

└ [Manage Credentials]

└ [Add credentials]

총 2번 수행해야 함



8. Jenkins Setting

(1) 톰캣 관리자의 Username과 Password 등록

Username : tomcat-manager

Password : 1111

* tomcat-users.xml 에 등록된 정보임

New credentials [Jenkins]

localhost:9090/jenkins/manage/credentials/store/system/domain/_/newCredentials

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

Kind

Username with password

Scope ?

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

Username ?

tomcat-manager

☐ Treat username as secret ?

Password ?

....

ID ?

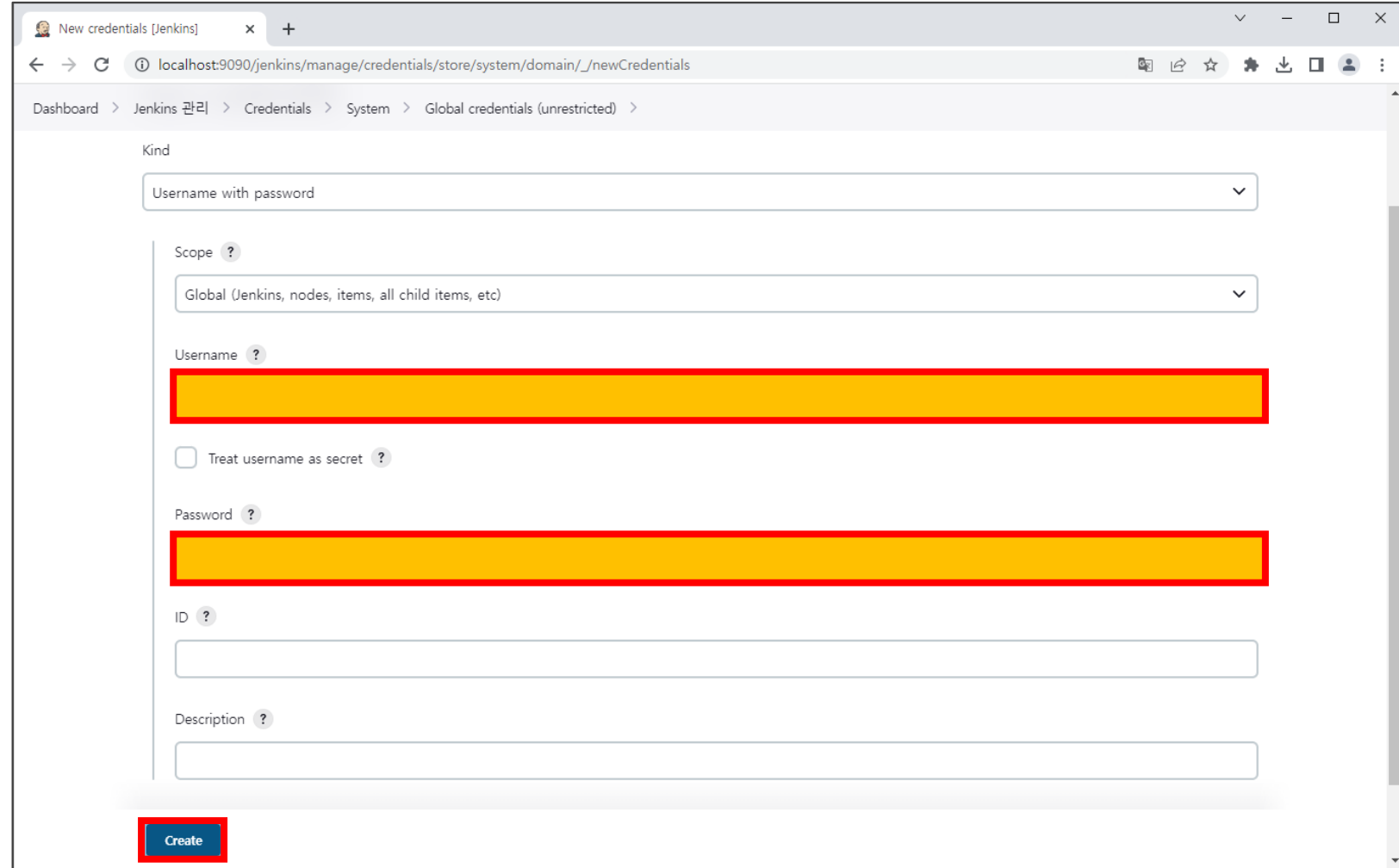
Description ?

Create

8. Jenkins Setting

(2) github 관리자의
Username과 Password 등록

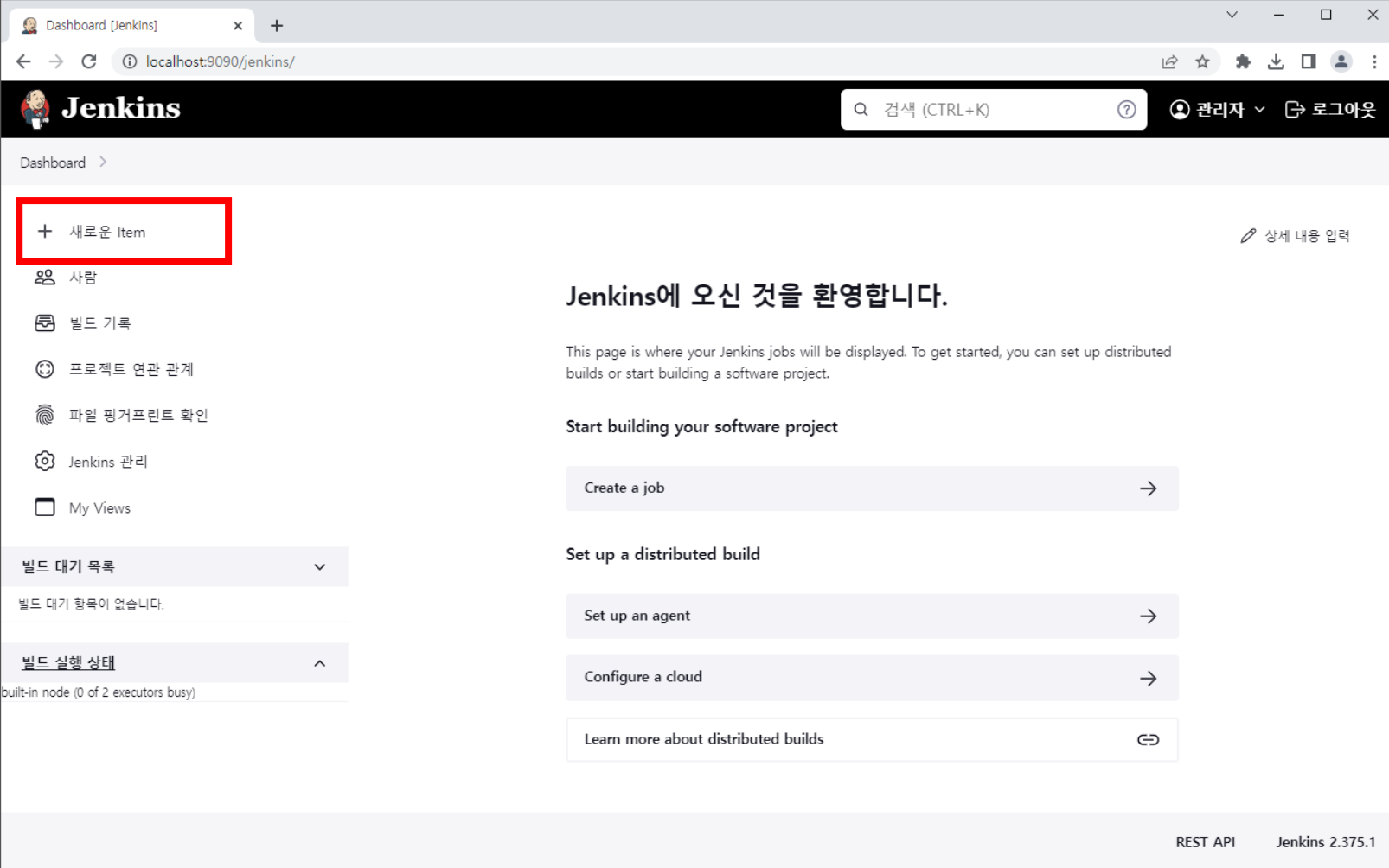
Username : 본인 github 아이디
Password : Personal access token 값



The screenshot shows the Jenkins web interface for creating new credentials. The browser tab is titled 'New credentials [Jenkins]' and the URL is 'localhost:9090/jenkins/manage/credentials/store/system/domain/_/newCredentials'. The breadcrumb navigation shows 'Dashboard > Jenkins 관리 > Credentials > System > Global credentials (unrestricted)'. The form is titled 'Kind' and has a dropdown menu set to 'Username with password'. Below this, the 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc)'. The 'Username' field is highlighted with a red border and contains a yellowed-out text. There is an unchecked checkbox labeled 'Treat username as secret'. The 'Password' field is also highlighted with a red border and contains a yellowed-out text. Below the password field are empty input fields for 'ID' and 'Description'. At the bottom left of the form, there is a blue 'Create' button highlighted with a red border.

9. Project Setting

[새로운 Item]



The screenshot shows the Jenkins Dashboard interface. The browser address bar indicates the URL is `localhost:9090/jenkins/`. The Jenkins logo and name are prominently displayed at the top. A search bar and user management links are visible in the top right. On the left sidebar, the '+ 새로운 Item' (New Item) button is highlighted with a red rectangular box. Below this, other sidebar options like '사람' (People), '빌드 기록' (Build History), and '프로젝트 연관 관계' (Project Relationships) are listed. The main content area features a welcome message in Korean, 'Jenkins에 오신 것을 환영합니다.', followed by instructions on how to start building a software project. Several action buttons are provided, including 'Create a job', 'Set up a distributed build', 'Set up an agent', 'Configure a cloud', and 'Learn more about distributed builds'. The bottom of the page shows the REST API and Jenkins version (2.375.1).

Dashboard [Jenkins] x +

← → ↻ ⓘ localhost:9090/jenkins/

Jenkins

Q 검색 (CTRL+K) ? 관리자 v 로그아웃

Dashboard >

+ 새로운 Item

사람

빌드 기록

프로젝트 연관 관계

파일 핑거프린트 확인

Jenkins 관리

My Views

빌드 대기 목록 ▼

빌드 대기 항목이 없습니다.

빌드 실행 상태 ▲

built-in node (0 of 2 executors busy)

Jenkins에 오신 것을 환영합니다.

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job →

Set up a distributed build

Set up an agent →

Configure a cloud →

Learn more about distributed builds ↗

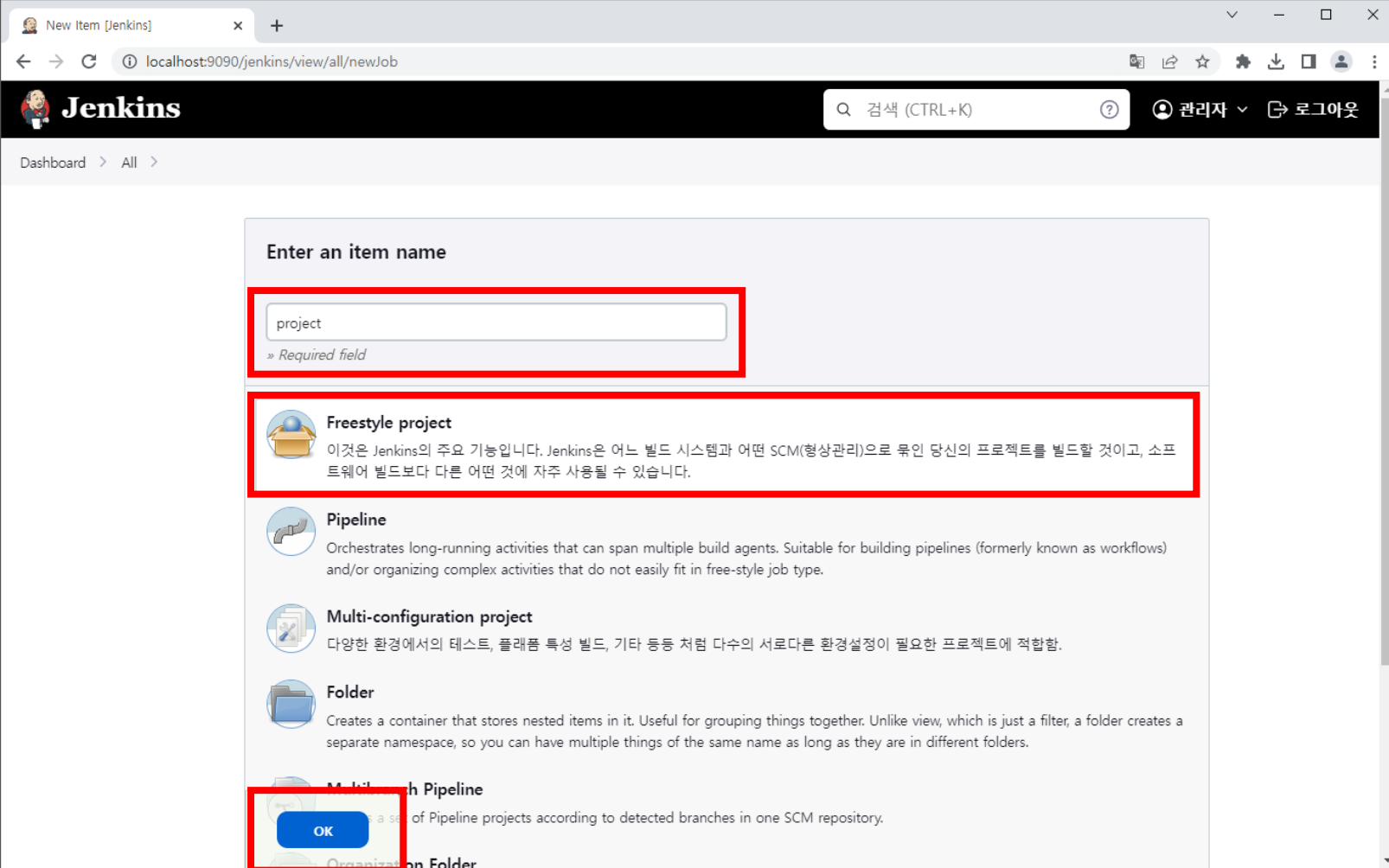
상세 내용 입력

REST API Jenkins 2.375.1

9. Project Setting

[프로젝트명] 입력

Freestyle project 선택



The screenshot shows the Jenkins 'New Item' page. The browser address bar indicates the URL is `localhost:9090/jenkins/view/all/newJob`. The page title is 'New Item [Jenkins]'. The Jenkins logo and a search bar are at the top. Below the navigation bar, the breadcrumb 'Dashboard > All >' is visible. The main content area is titled 'Enter an item name'. A text input field contains the word 'project', and a red box highlights this field with the text '» Required field' below it. Below the input field, there is a list of project types, each with an icon and a description. The 'Freestyle project' option is highlighted with a red box. The other options are 'Pipeline', 'Multi-configuration project', 'Folder', and 'Multi-branch Pipeline'. At the bottom of the list, there is an 'OK' button, which is also highlighted with a red box.

Enter an item name

project

» 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.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
다양한 환경에서의 테스트, 플러폼 특성 빌드, 기타 등등 처럼 다수의 서로다른 환경설정이 필요한 프로젝트에 적합함.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

Multi-branch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.

OK

9. Project Setting

[소스 코드 관리]

└ [Git]

└ [Repository URL]

애플리케이션github주소

└ [Credentials]

github관리자계정

└ [Branch Specifier]

.main

project Config [Jenkins]

localhost:9090/jenkins/job/project/configure

Dashboard > project > Configuration

Configure

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

소스 코드 관리

☐ None

☒ Git ?

Repositories ?

Repository URL ?
https://github.com/goodeeit/project.git

Credentials ?

goodeeit/*****

+ Add

고급...

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

*/main

저장 Apply

9. Project Setting

[Build Steps]

└ [Invoke top-level Maven targets]

└ [Maven Version]

MyMaven

└ [Goals]

clean package

project Config [Jenkins] x +

localhost:9090/jenkins/job/project/configure

Dashboard > project > Configuration

Configure

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

Build Steps

Invoke top-level Maven targets ?

Maven Version

MyMaven

Goals

clean package

고급...

Add build step

빌드 후 조치

빌드 후 조치 추가

저장 Apply

REST API Jenkins 2.375.1

9. Project Setting

[빌드 후 조치]

└ [Deploy war/ear to a container]

└ [WAR/EAR files]

****/*.war**

└ [Containers]

└ [Tomcat 9.x Remote]

└ [Credentials]

Tomcat관리자계정

└ [Tomcat URL]

http://localhost:9090

project Config [Jenkins] x +

localhost:9090/jenkins/job/project/configure

Dashboard > project > Configuration

Configure

- General
- 소스 코드 관리
- 빌드 유발
- 빌드 환경
- Build Steps
- 빌드 후 조치

빌드 후 조치

Deploy war/ear to a container

WAR/EAR files ?
**/*.war

Context path ?

Containers

Tomcat 9.x Remote

Credentials
tomcat-manager/*****

+ Add

Tomcat URL ?
http://localhost:9090

고급...

저장

Apply

10. Project Build & Deploy

[지금 빌드]

The screenshot shows the Jenkins web interface in a browser window. The address bar indicates the URL is `localhost:9090/jenkins/job/project/`. The Jenkins logo and name are at the top left, with a search bar and user controls on the top right. The breadcrumb navigation shows `Dashboard > project >`. On the left sidebar, the `지금 빌드` (Build Now) button is highlighted with a red rectangle. The main content area is titled `Project project` and includes a `고정링크` (Fixed Link) section. Below this, there is a `Build History` section with a search bar and a message stating `No builds`. At the bottom right, there are links for `Atom feed (현재)` and `Atom feed (실패)`. The footer shows `REST API` and `Jenkins 2.375.1`.

10. Project Build & Deploy

[빌드 실패 시]

└ [빌드 결과 #1]

└ [Console Output]

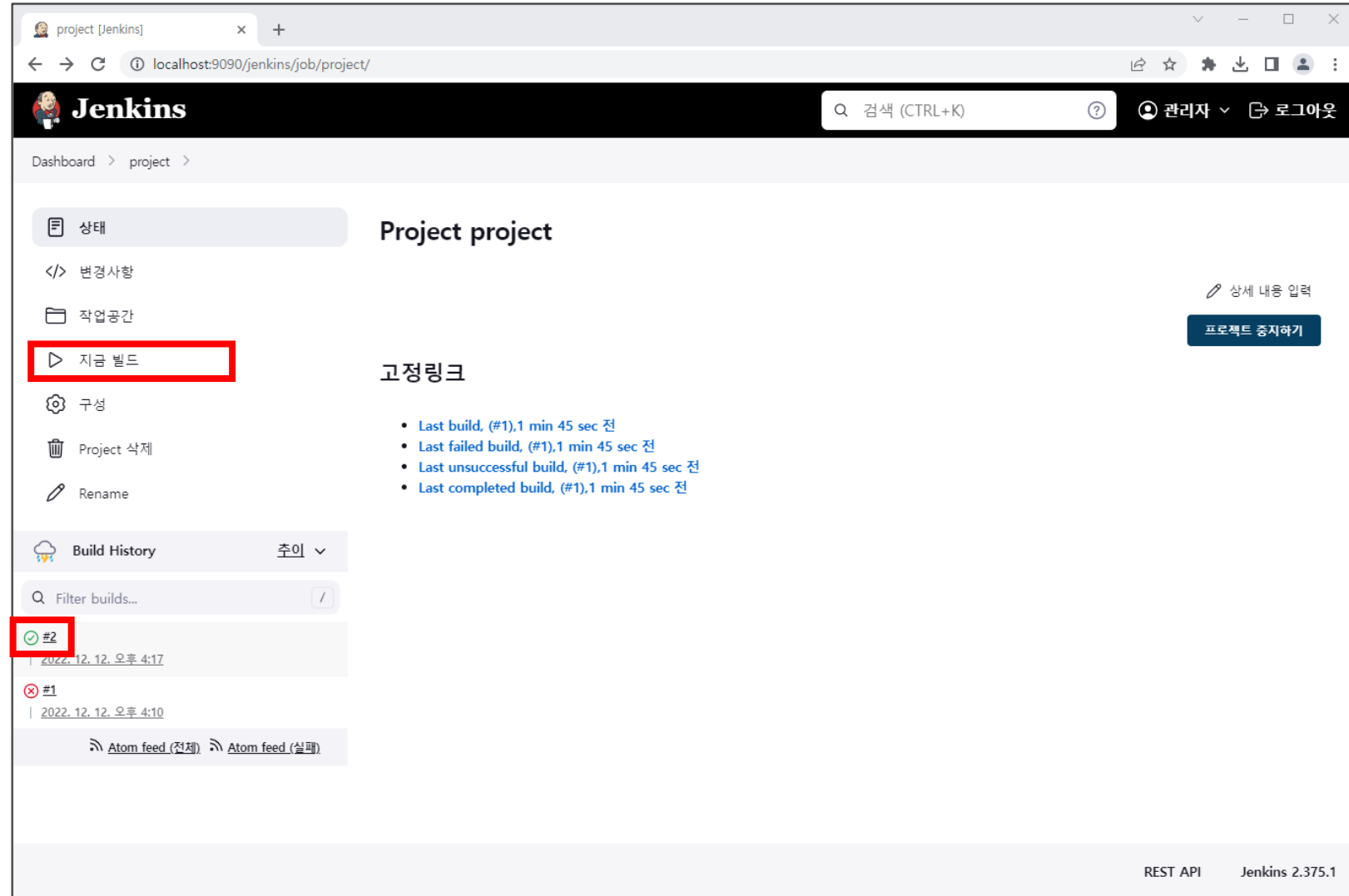
└ 실패 이유를 찾아서 수정한 뒤
다시 "지금 빌드" 수행

The screenshot shows the Jenkins web interface for a project named 'project'. The browser address bar indicates the URL is `localhost:9090/jenkins/job/project/`. The Jenkins logo and navigation bar are at the top. The left sidebar contains a menu with options: '상태' (Status), '변경사항' (Changes), '작업공간' (Workspace), '지금 빌드' (Build Now), '구성' (Configure), 'Project 삭제' (Delete Project), and 'Rename'. The '지금 빌드' button is highlighted with a red rectangle. The main content area shows 'Project project' with a '고정링크' (Fixed Link) section listing build history: 'Last build, (#1), 1 min 45 sec 전', 'Last failed build, (#1), 1 min 45 sec 전', 'Last unsuccessful build, (#1), 1 min 45 sec 전', and 'Last completed build, (#1), 1 min 45 sec 전'. Below this is the 'Build History' section, which includes a search bar and a list of builds. The first build, '#1', is highlighted with a red rectangle and has a red 'X' icon next to it, indicating it failed. A context menu is open for build #1, with the 'Console Output' option highlighted by a red rectangle. Other options in the menu include '바뀐점' (Changes), 'Atom feed (실패)' (Atom feed (Failed)), '빌드 정보 수정' (Edit Build Information), 'Delete build \'#1\'', and 'Git Build Data'. The bottom of the page shows the REST API link and the Jenkins version '2.375.1'.

10. Project Build & Deploy

[빌드 성공 시]

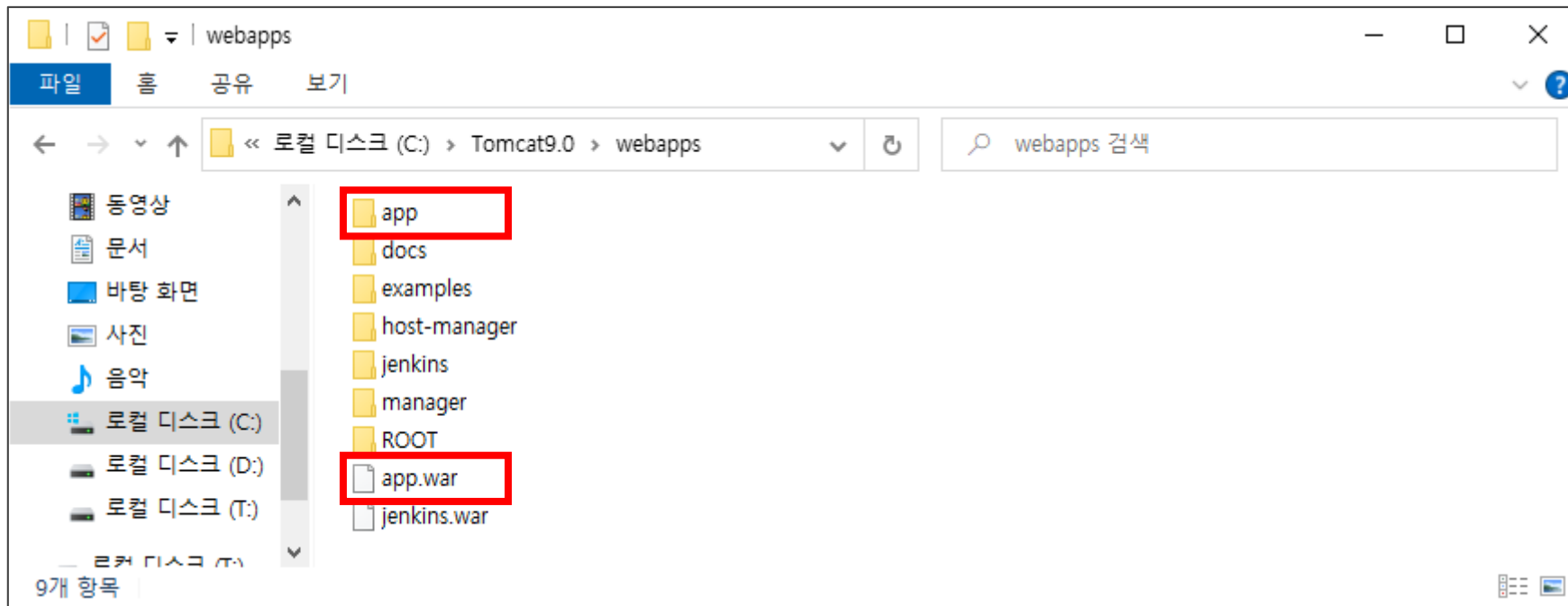
↳ [빌드 결과 #2]



The screenshot displays the Jenkins web interface for a project named 'project'. The browser address bar shows 'localhost:9090/jenkins/job/project/'. The Jenkins logo and navigation links are at the top. The left sidebar contains a menu with '상태' (Status), '변경사항' (Changes), '작업공간' (Workspace), '지금 빌드' (Build Now), '구성' (Configure), 'Project 삭제' (Delete Project), and 'Rename'. The '지금 빌드' button is highlighted with a red box. The main content area shows 'Project project' with a '상세 내용 입력' (Edit Description) button and a '프로젝트 중지하기' (Stop Project) button. Below this is a '고정링크' (Fixed Links) section with a list of build links: 'Last build, (#1), 1 min 45 sec 전', 'Last failed build, (#1), 1 min 45 sec 전', 'Last unsuccessful build, (#1), 1 min 45 sec 전', and 'Last completed build, (#1), 1 min 45 sec 전'. The 'Build History' section is visible below, showing a list of builds. Build #2 is highlighted with a red box, indicating it is the current build. Build #1 is also visible, marked as failed. The bottom of the page shows 'REST API' and 'Jenkins 2.375.1'.

10. Project Build & Deploy

톰캣에 배포되었는지 확인



10. Project Build & Deploy

`http://localhost:9090/app`

