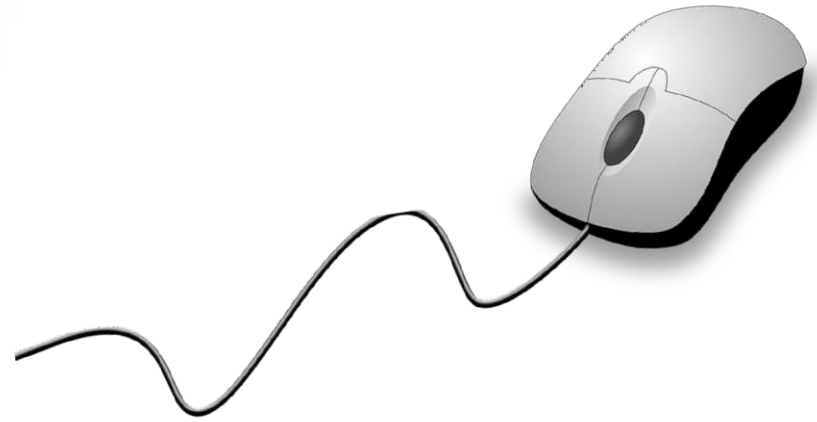


공개SW 솔루션 설치 & 활용 가이드

시스템SW > 자원관리



# Jenkins



## 제대로 배워보자

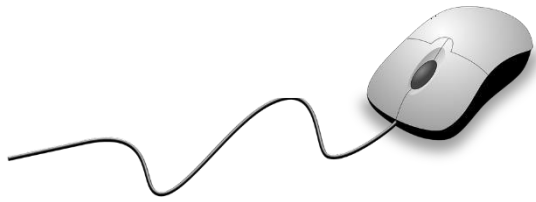
How to Use Open Source Software

---

Open Source Software Installation & Application Guide



오픈소스 소프트웨어 통합지원센터  
Open Source Software Support Center



# CONTENTS

1. 개요
2. 기능요약
3. 실행환경
4. 설치 및 실행
5. 기능소개
6. 활용예제
7. FAQ
8. 용어정리

# 1. 개요



Jenkins



<p><b>소개</b></p>	<ul style="list-style-type: none"> <li>• Java로 작성된 지속적 통합(Continuous Integration, CI)과 지속적 배포(continuous delivery, CD)를 위한 서버 또는 도구</li> <li>• 소프트웨어를 작성, 테스트 및 배포하거나 배포하는 것과 관련된 모든 종류의 작업을 자동화하는데 사용할 수 있는 오픈 소스 자동화 서버</li> </ul>		
<p><b>주요기능</b></p>	<ul style="list-style-type: none"> <li>• Jenkins를 사용하면 자동화를 통해 소프트웨어 개발 프로세스 가속화</li> <li>• Jenkins는 빌드, 문서, 테스트, 패키지, 스테이지, 배포, 정적 분석 등을 포함하여 모든 종류의 개발 라이프 사이클 프로세스 통합</li> </ul>		
<p><b>대분류</b></p>	<ul style="list-style-type: none"> <li>• 시스템 SW</li> </ul>	<p><b>소분류</b></p>	<ul style="list-style-type: none"> <li>• 자원관리</li> </ul>
<p><b>라이선스 형태</b></p>	<ul style="list-style-type: none"> <li>• MIT License</li> </ul>	<p><b>사전설치 솔루션</b></p>	<ul style="list-style-type: none"> <li>• Java 8 JRE or JDK</li> </ul>
<p><b>실행 하드웨어</b></p>	<ul style="list-style-type: none"> <li>• 256 MB RAM (512MB 권장) 10GB 이상의 디스크 공간 (최소 2GB)</li> </ul>	<p><b>버전</b></p>	<ul style="list-style-type: none"> <li>• 2.138.2(2018년 10월 기준)</li> </ul>
<p><b>특징</b></p>	<ul style="list-style-type: none"> <li>• Jenkins는 플러그인의 도움으로 지속적인 통합 실현</li> <li>• 플러그인을 사용하면 다양한 개발 단계를 통합 할 수 있음</li> <li>• 특정 도구를 통합하려면 해당 도구에 대한 플러그인을 설치,예) Git, Maven 2 프로젝트, Amazon EC2, HTML 게시자 등</li> </ul>		
<p><b>보안취약점</b></p>	<ul style="list-style-type: none"> <li>• 취약점 ID : CVE-2018-1999047</li> <li>• 심각도 : 6.5 MEDIUM(V3)</li> <li>• 취약점 설명 : UpdateCenter.java의 Jenkins 2.137 및 이전 버전, 2.121.2 및 이전 버전에 부적절한 인증 취약점이 존재하여 공격자가 업데이트 센터를 통해 예정된 Jenkins 다시 시작을 취소 할 수 있음</li> <li>• 대응방안 : 2.138 이상으로 업그레이드</li> <li>• 참고 경로 : <a href="https://jenkins.io/security/advisory/2018-08-15/#SECURITY-1076">https://jenkins.io/security/advisory/2018-08-15/#SECURITY-1076</a></li> </ul>		
<p><b>개발회사/커뮤니티</b></p>	<ul style="list-style-type: none"> <li>• Jenkins , CloudBees</li> </ul>		
<p><b>공식 홈페이지</b></p>	<ul style="list-style-type: none"> <li>• <a href="https://jenkins.io/">https://jenkins.io/</a>, <a href="https://www.cloudbees.com/">https://www.cloudbees.com/</a></li> </ul>		



## 2. 기능요약



Jenkins



- 원래는 "허드슨" 이라는 이름으로 잘 알려져 있지만, 2010년 오라클과의 상표권 문제로 인해 Jenkins로 이름 변경
- 공개SW CI(Continuous Integration) Tool로써 팀의 구성원들이 작업한 내용을 정기적으로 통합하는 것을 의미함
- 수많은 개발자들이 플러그인을 제공하면서 지속적인 버그 수정 및 새로운 개선 사항을 기대할 수 있음
- 특징
  - 훌륭한 커뮤니티 지원을 제공하는 오픈 소스 도구
  - 설치가 쉬움
  - 작업을 수월하게 만드는 1000 개 이상의 플러그인
  - 플러그인이 존재하지 않으면 코드를 작성하고 커뮤니티와 공유 가능
  - 무료
  - Java로 구축되었으므로 모든 주요 플랫폼 이식 가능



# 3. 실행환경



Jenkins



- Jenkins는 서블릿 컨테이너 위에서 구동되므로 Tomcat이나 Jetty같은컨테이너의설치 필요
- 내장 서블릿이 있으므로 전용패키지를 이용하여 설치하는 것도 가능하지만, 일반적인개발 환경과 비슷한 환경에서 설치 및 테스트를 하기위해 본 가이드에서는 tomcat을 사용하여 Jenkins 설치

설치환경	Ubuntu	
사전설치	Java 버전 8	Jenkins는 Java 기반 응용 프로그램이므로 Java 필수
	Apache Tomcat 버전 9	Jenkins war 파일을 배포하려면 Tomcat 필요
	Jenkins war File	Jenkins 설치하는데 필요



# 4. 설치 및 실행



Jenkins



## 세부 목차

1. Install Java Version 8
2. Install Apache Tomcat 9
3. Download Jenkins war File
4. Deploy Jenkins war File
5. Install Suggested Plugins



# 4. 설치 및 실행



Jenkins



## 4.1 Install Java Version 8

- 개발 혹은 테스트 서버에 java8을 설치하기 위해 다음 명령 실행
  - sudo apt-get install oracle-java8-installer0

```
mondayus@mondayus-VirtualBox:/etc/apt/sources.list.d$ sudo apt-get install oracle-java8-installer
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  gsfonst-x11 java-common oracle-java8-set-default
Suggested packages:
  binfmt-support visualvm ttf-baekmuk | ttf-unfonts | ttf-unfonts-core ttf-kochi-gothic | ttf-sazanami-gothic
  ttf-kochi-mincho | ttf-sazanami-mincho ttf-arphic-uming
The following NEW packages will be installed:
  gsfonst-x11 java-common oracle-java8-installer oracle-java8-set-default
```

- Jenkins를 설치하기 위해서는 tomcat을 이용하여 Jenkins war를 deploy 함, 특정 버전의 Apache Tomcat을 설치하기 위하여 먼저 다음 명령을 이용하여 wget 설치
  - sudo apt-get install wget

```
mondayus@mondayus-VirtualBox:~/Downloads$ sudo apt-get install wget
Reading package lists... Done
Building dependency tree
Reading state information... Done
wget is already the newest version (1.19.4-1ubuntu2.1).
0 upgraded, 0 newly installed, 0 to remove and 212 not upgraded.
```



# 4. 설치 및 실행



Jenkins



## 4.2 Install Apache Tomcat9(1/5)

- 이제 wget을 이용하여 Tomcat9 tar파일 최신버전 다운로드
  - wget <https://archive.apache.org/dist/tomcat/tomcat-9/v9.0.0.M10/bin/apache-tomcat-9.0.0.M10.tar.gz>

```
mondayus@mondayus-VirtualBox:~/Downloads$ wget https://archive.apache.org/dist/tomcat/tomcat-9/v9.0.0.M10/bin/apach
e-tomcat-9.0.0.M10.tar.gz
--2018-10-27 00:46:30-- https://archive.apache.org/dist/tomcat/tomcat-9/v9.0.0.M10/bin/apache-tomcat-9.0.0.M10.tar
.gz
Resolving archive.apache.org (archive.apache.org)... 163.172.17.199
Connecting to archive.apache.org (archive.apache.org)|163.172.17.199|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9335218 (8.9M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.0.M10.tar.gz'

apache-tomcat-9.0.0.M10.tar. 70%[=====] 6.27M 54.6KB/s eta 36s
```

- 다음 명령을 이용하여 다운로드 받은 tar파일의 압축을 풀어주고 디렉토리명 바꿈

```
mondayus@mondayus-VirtualBox:~/Downloads$ ls
apache-tomcat-9.0.0.M10.tar.gz
mondayus@mondayus-VirtualBox:~/Downloads$ tar xzf apache-tomcat-9.0.0.M10.tar.gz
mondayus@mondayus-VirtualBox:~/Downloads$ ls
apache-tomcat-9.0.0.M10 apache-tomcat-9.0.0.M10.tar.gz
mondayus@mondayus-VirtualBox:~/Downloads$ mv apache-tomcat-9.0.0.M10 tomcat9
mondayus@mondayus-VirtualBox:~/Downloads$ ls
apache-tomcat-9.0.0.M10.tar.gz tomcat9
mondayus@mondayus-VirtualBox:~/Downloads$
```





# 4. 설치 및 실행



Jenkins



## 4.2 Install Apache Tomcat9(2/5)

- Apache Tomcat의 username과 password를 수정하기 위해 다음 파일 수정  
- vi ~/Downloads/tomcat9/conf/tomcat-users.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements.  See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License.  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<tomcat-users xmlns="http://tomcat.apache.org/xml"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
              version="1.0">
<!--
NOTE: By default, no user is included in the "manager-gui" role required
```



# 4. 설치 및 실행



Jenkins



## 4.2 Install Apache Tomcat9(3/5)

- 위 내용을 지우고 다음 내용을 붙여넣기, username과 password로 입력한 test는 사용자의 편의대로 변경하도록 하고 저장

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="manager-gui"/>
  <role rolename="manager-script"/>
  <role rolename="manager-jmx"/>
  <role rolename="manager-status"/>
  <role rolename="admin-gui"/>
  <role rolename="admin-script"/>
  <user username="test" password="test" roles="manager-gui,manager-script,manager
    -jmx,manager-status,admin-gui,admin-script"/>
</tomcat-users>0
```



# 4. 설치 및 실행

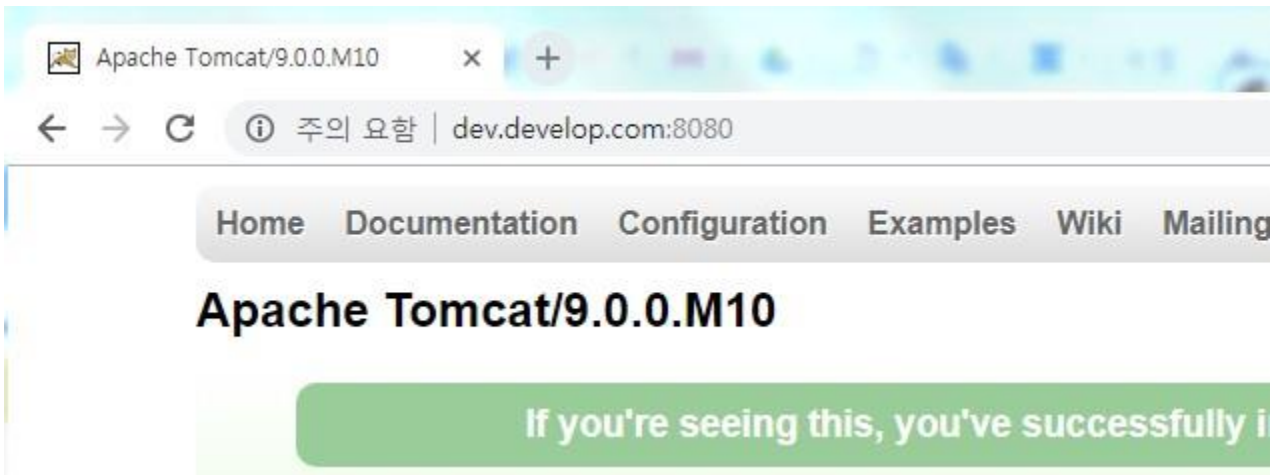


## 4.2 Install Apache Tomcat9(4/5)

- tomcat을 실행하기 위해 tomcat9의 bin디렉터리로 들어간 후 다음 명령 실행  
- ./startup.sh

```
mondayus@mondayus-VirtualBox:~/Downloads/tomcat9/bin$ ./startup.sh
Using CATALINA_BASE:   /home/mondayus/Downloads/tomcat9
Using CATALINA_HOME:   /home/mondayus/Downloads/tomcat9
Using CATALINA_TMPDIR: /home/mondayus/Downloads/tomcat9/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /home/mondayus/Downloads/tomcat9/bin/bootstrap.jar:/home/mondayus/Downloads/tomcat9/bin/tomcat-juli.jar
Tomcat started.
```

- tomcat이 실행, 브라우저를 열고 해당 서버 접근



# 4. 설치 및 실행



## 4.2 Install Apache Tomcat9(5/5)

- 본 가이드에서는 가상머신에 ubuntu를 설치하고 ubuntu에서 설치한 톰캣을 start하고 윈도우에서 접근
- 본 가이드에서는 가상머신의 ip를 host에 dev.develop.com으로 설정하여 http://dev.develop.com:8080/ 접근하였으므로, 각자 tomcat에 접근 할 때에는 'http://설치 한 서버의 ip:8080' 접근


주의 요약 | dev.develop.com:8080

Home Documentation Configuration Examples Wiki Mailing Lists Find Help

### Apache Tomcat/9.0.0.M10

The Apache Software Foundation  
<http://www.apache.org/>

If you're seeing this, you've successfully installed Tomcat. Congratulations!

 **Recommended Reading:**

- [Security Considerations HOW-TO](#)
- [Manager Application HOW-TO](#)
- [Clustering/Session Replication HOW-TO](#)

Server Status  
Manager App  
Host Manager

**Developer Quick Start**

- [Tomcat Setup](#)
- [First Web Application](#)
- [Realms & AAA](#)
- [JDBC DataSources](#)
- [Examples](#)
- [Servlet Specifications](#)
- [Tomcat Versions](#)



# 4. 설치 및 실행



Jenkins



## 4.3 Download Jenkins war File

- Jenkins war(web archive) file 을 다운로드 하기 위해서 다시 한 번 wget명령어를 이용, 다음 링크를 이용하여 Jenkins war의 release list 확인

Jenkins war release list : <http://updates.jenkins-ci.org/download/war/>

- 위 리스트에서 현 안정 버전인 2.138 을 다운, 본 가이드에서는 2.138.1 버전 설치, 아래 명령어를 이용하여 war 다운
  - **wget <http://updates.jenkins-ci.org/download/war/2.138.1/jenkins.war>**

```
mondayus@mondayus-VirtualBox:~/Downloads$ wget http://updates.jenkins-ci.org/download/war/2.138.1/jenkins.war
--2018-10-28 23:47:44-- http://updates.jenkins-ci.org/download/war/2.138.1/jenkins.war
Resolving updates.jenkins-ci.org (updates.jenkins-ci.org)... 52.202.51.185
Connecting to updates.jenkins-ci.org (updates.jenkins-ci.org)|52.202.51.185|:80.. connected.
HTTP request sent, awaiting response... 302 Found
Location: http://mirrors.jenkins-ci.org/war-stable/2.138.1/jenkins.war [following]
--2018-10-28 23:47:45-- http://mirrors.jenkins-ci.org/war-stable/2.138.1/jenkins.war
Resolving mirrors.jenkins-ci.org (mirrors.jenkins-ci.org)... 52.202.51.185
Reusing existing connection to updates.jenkins-ci.org:80.
```



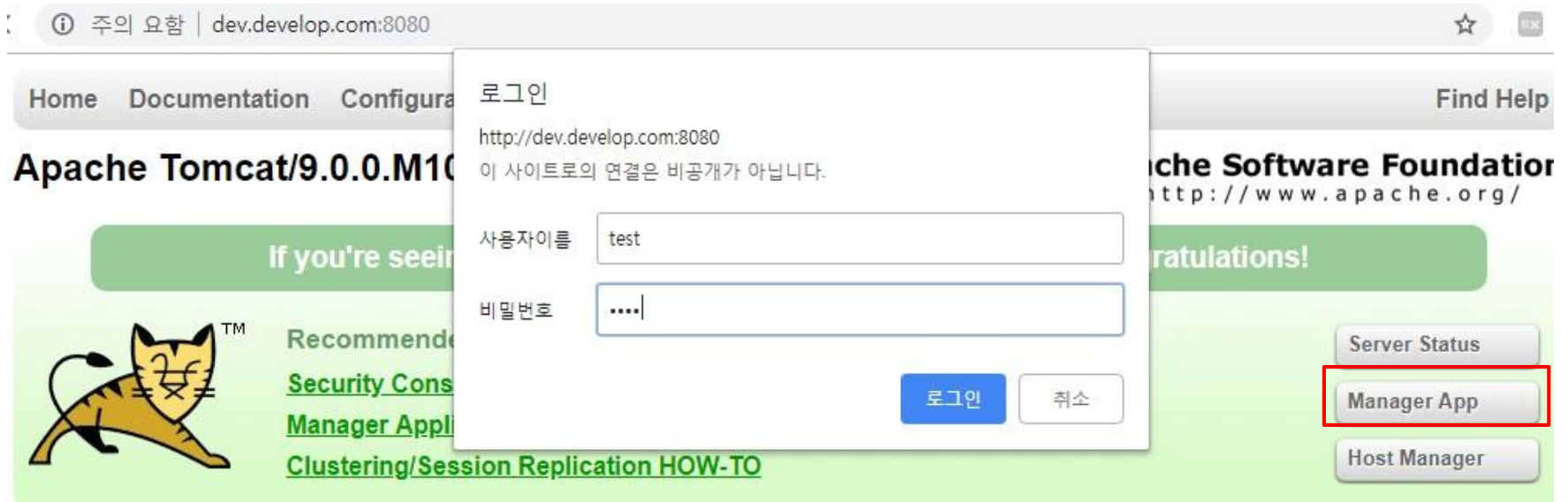


# 4. 설치 및 실행



## 4.4 Deploy Jenkins war File(1/3)

- Jenkins war file을 deploy하기 위해서 브라우저를 열고 이전, tomcat을 설치한 스텝에서 처럼 `http://dev.develop.com:8080/` 접속
- 그리고 Manager App 버튼 클릭  
로그인 다이얼로그가 뜨면 사용자이름과 비밀번호를 입력하고 로그인  
사용자이름 : test  
비밀번호 : test



# 4. 설치 및 실행



## 4.4 Deploy Jenkins war File(2/3)

- 이제 Tomcat web application manager page에 직접 접근이 가능, 스크롤을 아래로 내려 보면 Deploy 옵션버튼을 찾을 수 있음, jenkins.war를 내려받은 경로와 Context Path를 입력한 뒤에 Deploy버튼 누름

<a href="#">/manager</a>	<i>None specified</i>	Tomcat Manager Application	true	<u>1</u>	Start Stop Expire sess
--------------------------	-----------------------	----------------------------	------	----------	---------------------------

**Deploy**

Deploy directory or WAR file located on server

Context Path (required):

XML Configuration file URL:

WAR or Directory URL:

WAR file to deploy



# 4. 설치 및 실행



## 4.4 Deploy Jenkins war File(3/3)

- Deploy가 완료되면 이제 Tomcat web application manager page에서 Jenkins가 목록에 노출, ubuntu에 설치한 tomcat의 webapps에도 jenkins 디렉터리가 생성된 것 확인
- /jenkins를 누르고 다음 스텝으로 넘어감

<u>/host-manager</u>	<i>None specified</i>	Tomcat Host Manager Application	true	<u>0</u>
<u>/jenkins</u>	<i>None specified</i>	Jenkins v2.138.1	true	<u>0</u>
<u>/manager</u>	<i>None specified</i>	Tomcat Manager Application	true	<u>1</u>

```
mondayus@mondayus-VirtualBox:~/Downloads/tomcat9/webapps$ ls  
docs examples host-manager jenkins manager ROOT
```





# 4. 설치 및 실행



## 4.5 Install Suggested Plugins(1/5)

- "/jenkins를 클릭하면 unlock Jenkins page로 넘어감, Jenkins를 unlock하기 위해 일회성 password를 입력하라는 문구 확인
- 본 가이드의 경우 /home/mondayus/.jenkins/secrets/initialAdminPassword를 참조하여 패스워드를 확인, 패스워드를 입력 후 다음 버튼 누름



# 4. 설치 및 실행



## 4.5 Install Suggested Plugins(2/5)

- 아래 보이는 두 가지 옵션 중에서 본 가이드에서는 suggested plugins 선택

Getting Started

### Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

**Install suggested plugins**

Install plugins the Jenkins community finds most useful.

**Select plugins to install**

Select and install plugins most suitable for your needs.

Jenkins 2.138.2



# 4. 설치 및 실행



Jenkins



## 4.5 Install Suggested Plugins(3/5)

- Install Suggested Plugins를 클릭하면 아래와 같이 설치될 플러그인 리스트

Getting Started

# Getting Started

✓ Folders	Formatter			++ JDK Tool
🔄 Timestamper	🔄 Workspace Cleanup	🔄 Ant	🔄 Gradle	
🔄 Pipeline	🔄 GitHub Branch Source	🔄 Pipeline: GitHub Groovy Libraries	🔄 Pipeline: Stage View	
🔄 Git	🔄 Subversion	🔄 SSH Slaves	🔄 Matrix Authorization Strategy	
🔄 PAM Authentication	🔄 LDAP	🔄 Email Extension	🔄 Mailer	

++ - required dependency

Jenkins 2.138.2



# 4. 설치 및 실행



## 4.5 Install Suggested Plugins(4/5)

- 플러그인들이 설치되면 아래 페이지 넘어감
- 아래 빈 칸을 채운 후 저장 및 종료 버튼 누름

ⓘ 주의 요함 | dev.develop.com:8080/jenkins/ ☆ ✕

### Getting Started

# Create First Admin User

계정명:

암호:

암호 확인:

이름:

Jenkins 2.7.3 Continue as admin Save and Finish



# 4. 설치 및 실행



## 4.5 Install Suggested Plugins(5/5)

- Jenkins 준비
- Start using Jenkins 버튼을 누르면 바로 Jenkins Dashboard 향함

The screenshot shows the Jenkins web interface. At the top, the browser address bar displays 'dev.develop.com:8080/jenkins/'. The main content area is titled 'Getting Started' and features a large heading 'Jenkins is ready!' with the subtext 'Your Jenkins setup is complete.' Below this is a blue button labeled 'Start using Jenkins'. To the right, there is a navigation menu with icons for '새로운 Item', '사람', '빌드 기록', 'Jenkins 관리', 'My Views', and 'Credentials'. A search bar and a 'test' button are also visible. A message on the right says 'Jenkins에 오신 것을 환영합니다.' (Welcome to Jenkins) and '시작하려면 새 작업을 만들어 주시기 바랍니다.' (Please create a new job to get started). At the bottom left, the version 'Jenkins 2.7.3' is displayed.

# 5. 기능소개



**Jenkins**



세부 목차

1. Dashboard
2. New Item
3. People
4. Manage Jenkins



# 5. 기능소개



## 5.1 Dashboard

- Jenkins 기본 화면에는 현재 빌드 큐 및 Executor 상태가 표시되고 새 항목(New item)을 만들고, 사용자를 관리하고, 빌드 기록을 보고, Jenkins를 관리하고, 사용자 정의 보기를 보고, 자격증명을 관리하는 링크 제공

The screenshot shows the Jenkins dashboard with a sidebar on the left containing navigation links: New Item, People, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, My Views, and Credentials. The main content area features a table of build items and two summary panels.

S	W	Name ↓	Last Success
		<a href="#">admin_test</a>	N/A

Icon: [S](#) [M](#) [L](#)

**Build Queue**

No builds in the queue.

**Build Executor Status**

- 1 Idle
- 2 [admin\\_test](#) #1





# 5. 기능소개



Jenkins



## 5.2 New Item

- New Item 목록에는 Item을 정의 할 다양한 타입의 작업과 폴더 존재

### Enter an item name

» Required field



#### Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.



#### Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.



#### 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.



#### GitHub Organization

Scans a GitHub organization (or user account) for all repositories matching some defined markers.

if you want to create a new item from other existing, you can use this option:



Copy from

Type to autocomplete

OK



# 5. 기능소개

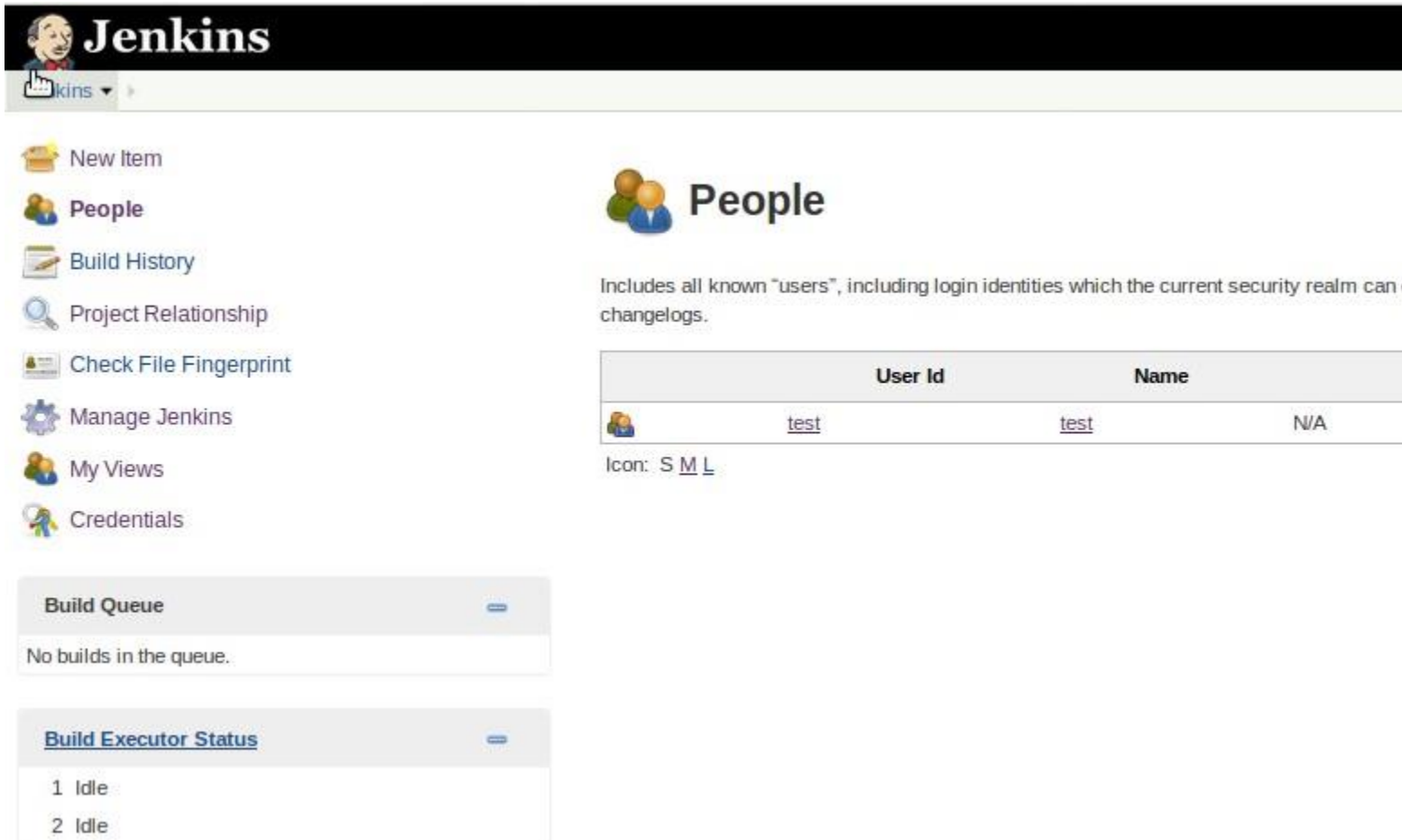


Jenkins



## 5.3 People

- People 목록에는 Jenkins에 접근 할 수 있는 사용자 리스트가 나타냄, 사용자를 추가하거나 삭제하기 위해서는 Manage Jenkins의 User Manage 메뉴 이용



The screenshot shows the Jenkins web interface. At the top left is the Jenkins logo and a dropdown menu. Below it is a sidebar with navigation options: New Item, People, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, My Views, and Credentials. The main content area is titled 'People' and includes a description: 'Includes all known "users", including login identities which the current security realm can e changelogs.' Below this is a table with columns 'User Id' and 'Name'. The table contains one row with 'test' in both columns and 'N/A' in the third column. Below the table are links for 'Icon: S M L'. At the bottom, there are two expandable sections: 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing '1 Idle' and '2 Idle').

User Id	Name	
<a href="#">test</a>	<a href="#">test</a>	N/A

Icon: [S](#) [M](#) [L](#)

**Build Queue** [-]  
No builds in the queue.

**Build Executor Status** [-]  
1 Idle  
2 Idle



# 5. 기능소개



Jenkins



## 5.4 Manage Jenkins

- Manage Jenkins에서는 command-line 인터페이스를 여는 옵션 및 유저 등록 메뉴를 포함하여 젠킨스 관리 페이지에서 수행 할 수 있는 여러 작업들이 목록화 되어있음

**Jenkins** search

Jenkins ▾

- New Item
- People
- Build History
- Project Relationship
- Check File Fingerprint
- Manage Jenkins**
- My Views
- Credentials

**Build Queue** -

No builds in the queue.

**Build Executor Status** -

1	Idle
2	Idle

### Manage Jenkins

**⚠ New version of Jenkins (2.138.2) is available for [download](#) ([changelog](#)).**

- Configure System**  
Configure global settings and paths.
- Configure Global Security**  
Secure Jenkins; define who is allowed to access/use the system.
- Configure Credentials**  
Configure the credential providers and types
- Global Tool Configuration**  
Configure tools, their locations and automatic installers.
- Reload Configuration from Disk**  
Discard all the loaded data in memory and reload everything from file system. Useful when you
- Manage Plugins**  
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- System Information**  
Displays various environmental information to assist trouble-shooting.
- System Log**  
System log captures output from `java.util.logging` output related to Jenkins.
- Load Statistics**  
Check your resource utilization and see if you need more computers for your builds.



# 6. 활용예제



Jenkins



## 세부 목차

1. 시작하기
2. New Item 생성
3. Item 설정
4. Build Now
5. Console Output - 빌드 로그 확인
6. war확인 - 빌드된 war 확인



# 6. 활용예제



Jenkins



## 6.1 시작하기

- 형상관리를 github를 이용하여 하고 있다면, 개략적인 개발 및 배포 플로우는 대체로 다음과 같음
  1. git에서 개발 branch를 기준으로 new branch 생성
  2. 로컬 환경에서 작업 및 테스트를 거친 후 개발 branch에 작업사항을 머지하고 개발branch기준으로 오류가 없는지 확인 후 프로젝트 추출(war or jar)
  3. 서버 환경에서의 테스트를 위해 추출물을(war or jar) 개발 서버 전송
  4. Tomcat등 서버 재시작으로 추출물 deploy
  5. 테스트 중 오류사항이 보이면 로컬에서 재 작업 후 추출물을 다시 생성하여 서버에 전송 그리고 서버 재시작으로 deploy (수정 사항이 있을 때 마다 반복)
  6. 테스트가 완료되면 master branch에 작업사항 merge
  7. 추후 운영 배포일에 master branch를 기준으로 추출물을 추출하여 서버에 전송 및 deploy
- 위 작업 중 반복되는 작업은 2 ~ 5이며, 이 과정을 jenkins 이용하여 손쉽게 진행, github push하는 순간 jenkins에서 pull하고 바로 빌드하여 서버에 deploy할 수 있기 때문임, Jenkins 설치를 마쳤으므로 실제 프로젝트에서 사용하는 배포 환경에서 Jenkins 활용



# 6. 활용예제



## 6.2 New Item 생성(1/2)

- 메인 좌측 화면에서 New Item 클릭

The screenshot shows the Jenkins dashboard. On the left sidebar, the 'New Item' button is highlighted with a red box. Below it are other navigation options: People, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, My Views, and Credentials. The main content area features a 'Welcome to Jenkins!' message and a light blue box with the text: 'Please create new jobs to get started.' Below the welcome message, there are two sections: 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing two idle executors).



# 6. 활용예제



Jenkins





## 6.2 New Item 생성(2/2)


- 프로젝트 명을 입력한 후 Freestyle Project 를 클릭하고 Ok 누름


### Enter an item name


*» Required field*

 **Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

 **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**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

 **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.

 **GitHub Organization**



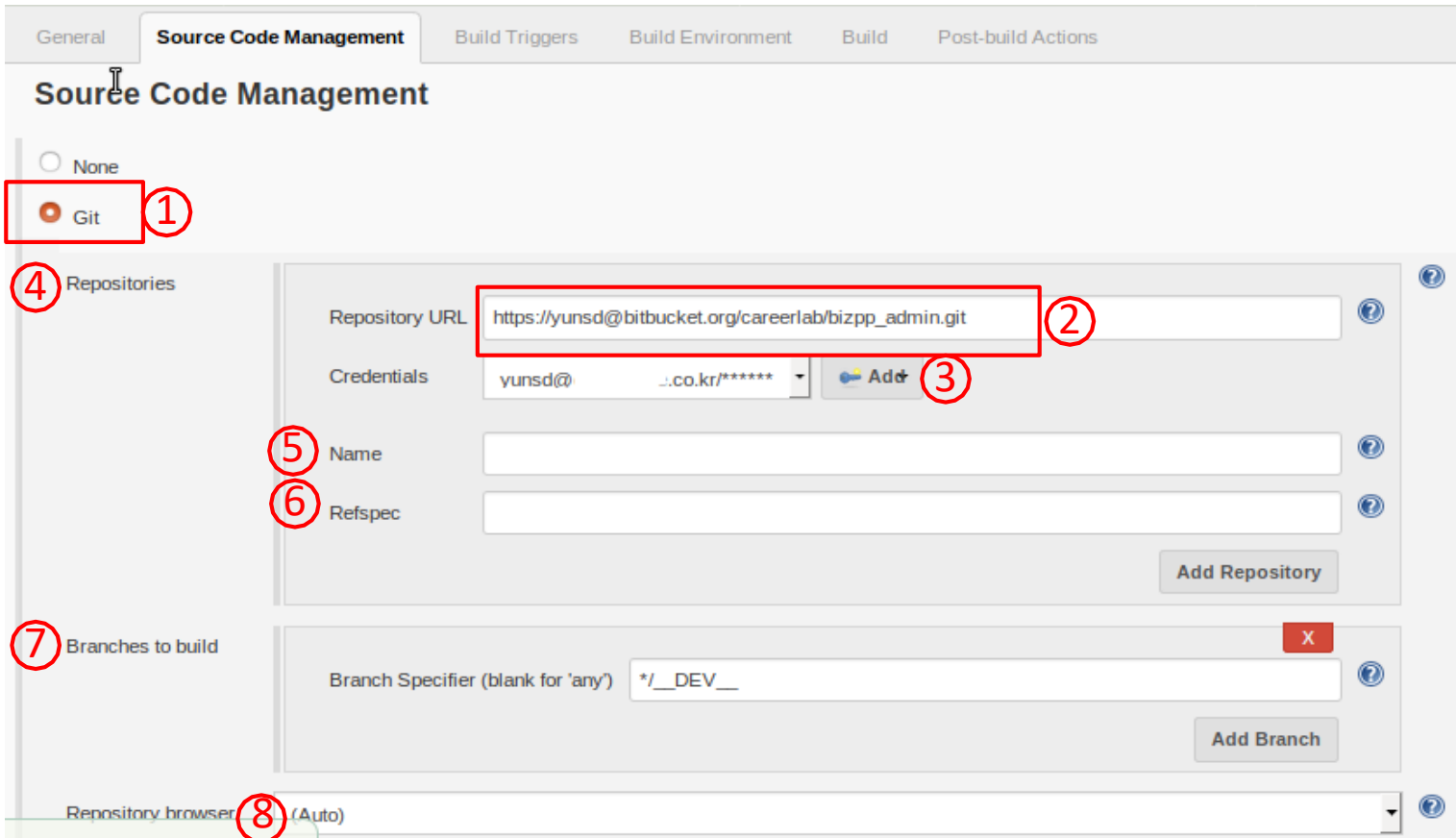


# 6. 활용예제



## 6.3 Item 설정(1/6)

- Source Code Management 에서 Git ①을 클릭하고 Repository URL ②에 git 정보 입력
- git저장소 연결에 계정이 필요한 경우 add ③버튼을 눌러 계정정보 추가



The screenshot shows the Jenkins 'Source Code Management' configuration page. The 'General' tab is selected. The 'Source Code Management' section is active, and the 'Git' radio button is selected (callout 1). Below this, the 'Repositories' section (callout 4) contains a form with the following fields: 'Repository URL' (callout 2) containing 'https://yunsd@bitbucket.org/careerlab/bizpp\_admin.git', 'Credentials' (callout 3) with a dropdown menu showing 'yunsd@...co.kr/\*\*\*\*\*' and an 'Add' button, 'Name' (callout 5), and 'Refspec' (callout 6). An 'Add Repository' button is at the bottom right of this section. Below the 'Repositories' section is the 'Branches to build' section (callout 7), which has a 'Branch Specifier (blank for 'any')' field containing '\*/\_DEV\_' and an 'Add Branch' button. At the bottom, there is a 'Repository browser' dropdown menu (callout 8) set to '(Auto)'. The page has tabs for 'General', 'Source Code Management', 'Build Triggers', 'Build Environment', 'Build', and 'Post-build Actions'.

# 6. 활용예제



Jenkins



## 6.3 Item 설정(2/6)

- **Advanced** ④ 버튼을 클릭하면 Name⑤, Refspec⑥ 두 가지 텍스트 박스 확인 가능  
대부분의 경우 아무런 입력값을 넣지 않아도 됨  
Name : 비워두면 젠킨스가 자동으로 고유한 이름 생성 생  
성된 이름은 git remote 명령에 사용된 것과 동일  
Refspec : 기본 refspec는 "+refs/heads/\*:refs/remotes/REPOSITORYNAME/\*"  
기본적으로 모든 브랜치 헤드는 remotes/repositoryname/branchname으로  
검색되지만 만약 한 가지 브랜치만 항상 검색 하고 싶다면 이 필드에 값 입력
- **Branch Specifier**⑦에는 젠킨스에서 추적 할 브랜치 입력이며, master 브랜치가 기본값
- **Repository browser**⑧ 는 기본적으로 드롭 다운 목록이 자동 설정되며, 대부분의 경우 기본값 가능





# 6. 활용예제



## 6.3 Item 설정(3/6)

- 스크롤을 내려서 빌드 설정 부분 이동 후 빌드 설정을 invoke Gradle 선택 저장  
만약 Maven으로 빌드하는 프로젝트의 경우에는 'Invoke top-level Maven targets' 선택



- gradle을 이용해 빌드하는 프로젝트이므로, 젠킨스에서 빌드가 수행되려면 gradle 설치 필요,  
잠시 터미널로 돌아가서 gradle을 설치 후 돌아옴

```
mondayus@mondayus-VirtualBox:~/Downloads/tomcat9/bin$ sudo apt install gradle
[sudo] password for mondayus:
Reading package lists... Done
Building dependency tree
```



# 6. 활용예제



Jenkins



## 6.3 Item 설정(4/6)

- 스크롤을 조금 내리면 Add build step 선택 박스 보임
- 클릭해서 Execute shell 메뉴를 선택하고 아래와 같이 command 입력
- 프로젝트를 빌드 후 해당 프로젝트를 가동하는 소스임
- 완료 되었다면 저장 버튼을 누르고 메인으로 감

The screenshot shows the Jenkins configuration page for a build step. The 'Invoke Gradle' section is visible with 'Gradle Version' set to '(Default)'. Below it, the 'Execute shell' section is active, showing a list of build steps in a dropdown menu. The 'Execute shell' option is highlighted. The command text area contains the following shell script:

```
radle -Dmaven.test.failure.ignore clean build
ouch /home/mondayus/Downloads/adminTomcat/bin/tomcat.pid
h /home/mondayus/Downloads/adminTomcat/bin/shutdown.sh
leep 5
h /home/mondayus/Downloads/adminTomcat/bin/adminDeploy.sh
UILD_ID=dontKillMe /home/mondayus/Downloads/adminTomcat/bin/startup.sh
```

At the bottom of the command area, there is a link: [e the list of available environment variables](#). An 'Advanced...' button is located at the bottom right of the configuration area.



# 6. 활용예제



Jenkins



## 6.3 Item 설정(5/6)

- 위 Execute shell에서 설정 한 대로 프로젝트의 빌드가 완료되었을 때 프로젝트를 구동하기 위해서는 톰캣이 한 세트 더 필요
- jenkins를 띄우기 위해 설치해 두었던 tomcat9를 복사해서 adminTomcat 하나 생성

```
mondayus@mondayus-VirtualBox:~/Downloads$ ll
total 16
drwxr-xr-x  4 mondayus mondayus 4096 11월 15 06:13 ./
drwxr-xr-x 24 mondayus mondayus 4096 11월 15 06:41 ../
drwxr-xr-x  9 mondayus mondayus 4096 11월 15 06:13 adminTomcat/
drwxr-xr-x  9 mondayus mondayus 4096 10월 27 00:50 tomcat9/
```

- jenkins에서 빌드 수행시 생성되는 war파일을 adminTomcat으로 가져오기 위해서 다음 설정
  - cd ./adminTomcat/bin
  - vi adminDeploy.sh

```
mondayus@mondayus-VirtualBox: ~/Downloads/adminTomcat/bin
File Edit View Search Terminal Help
# /bin/sh
rm -rf /home/mondayus/Downloads/adminTomcat/webapps/admin /home/mondayus/Downloads/adminTomcat/webapps/admin.war
mv /home/mondayus/.jenkins/workspace/helloJenkins/build/libs/admin.war /home/mondayus/Downloads/adminTomcat/webapps/admin.war
```



# 6. 활용예제



Jenkins



## 6.3 Item 설정(6/6)

- adminTomcat은 tomcat9를 복사한 것이므로 서버 설정 동일
- tomcat9와 함께 띄우기 위해서는 server.xml설정 변경
- 아래와 같이 port 설정 부분을 찾아서 각각 8005 -> 9005, 8080 -> 9090, 8443 -> 9443 변경

```
~/Downloads/adminTomcat/conf$ vi server.xml
```

```
<Server port="9005" shutdown="SHUTDOWN">  
  <Listener className="org.apache.catalina.startup.VersionLoggerListener" />
```

```
<Connector port="9090" protocol="HTTP/1.1"  
  connectionTimeout="20000"  
  redirectPort="8443" />
```

```
<Connector port="9443" protocol="org.apache.coyote.http11.Http11NioProtocol"  
  maxThreads="150" SSLEnabled="true">  
  <SSLHostConfig>  
    <Certificate certificateKeystoreFile="conf/localhost-rsa.jks"  
      type="RSA" />  
  </SSLHostConfig>  
</Connector>
```

- tomcat9를 복사 했기 때문에 한 가지 더 해줘야 할 것은 jenkins 관련 파일 삭제
- adminTomcat/webapps에서 jenkins.war와 jenkins 삭제

```
drwxr-x--- 11 mondayus mondayus 4096 10월 28 23:50 jenkins/  
-rw-r----- 1 mondayus mondayus 75714886 10월 28 23:50 jenkins.war
```

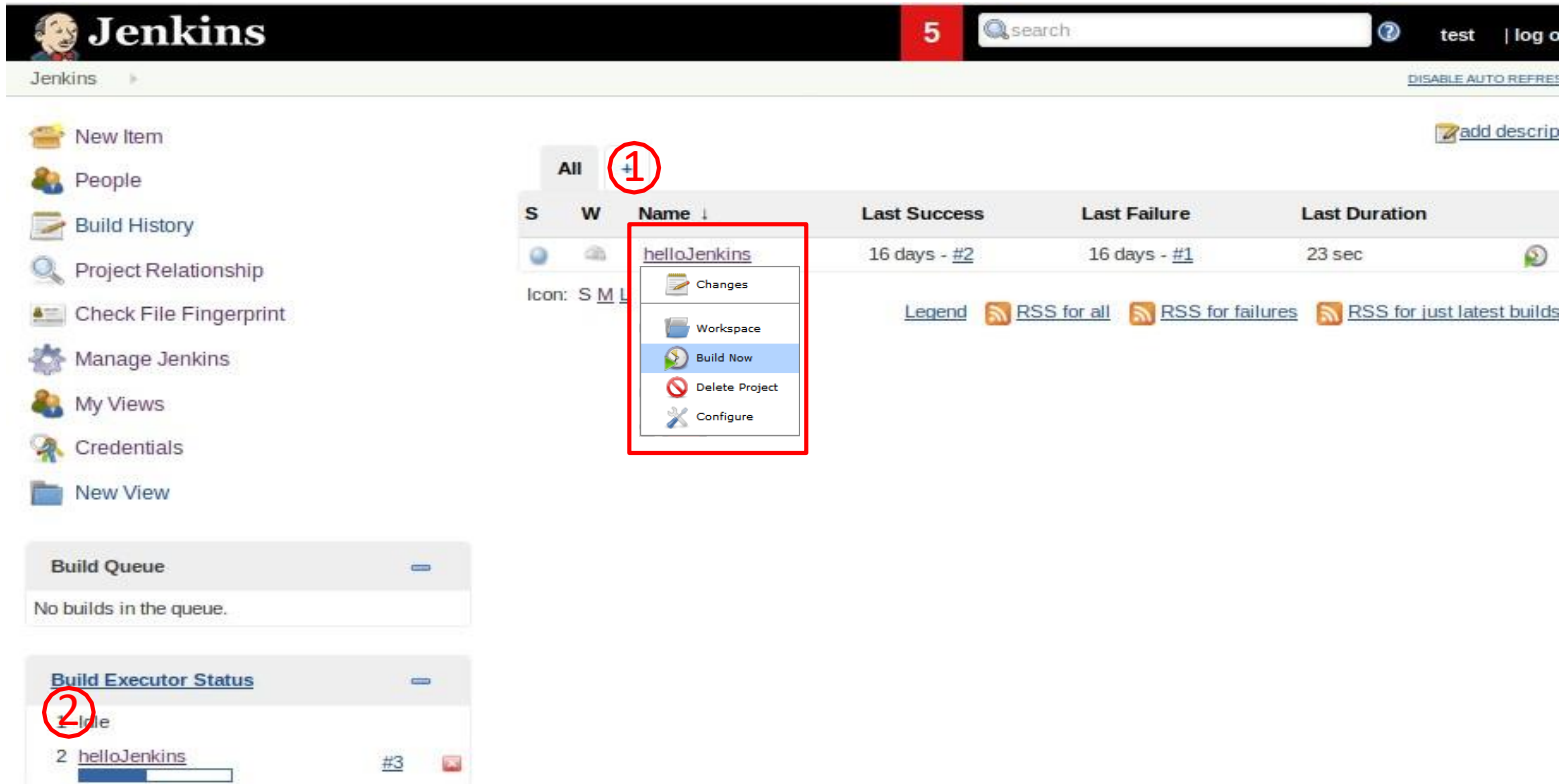


# 6. 활용예제



## 6.4 Build Now(1/2)

- 모든 설정이 완료되면 메인에 설정된 프로젝트 표출
- 프로젝트에 마우스를 가져다 대면 해당 프로젝트가 수행 할 수 있는 메뉴 표출
- 그 중에서 Build Now(1)를 눌러 빌드 수행
- 프로젝트가 빌드 되고 있는 것은 Build Queue에서 확인 가능(2)



The screenshot shows the Jenkins dashboard. At the top, there's a navigation bar with the Jenkins logo, a search bar, and a user profile 'test'. Below the navigation bar, there's a sidebar with various menu items like 'New Item', 'People', 'Build History', etc. The main content area displays a table of projects. The first project is 'helloJenkins', which has a context menu open over it. The context menu includes options like 'Changes', 'Workspace', 'Build Now', 'Delete Project', and 'Configure'. The 'Build Now' option is highlighted. Below the table, there's a 'Build Queue' section showing 'No builds in the queue.' and a 'Build Executor Status' section showing '2 helloJenkins #3'.

S	W	Name ↓	Last Success	Last Failure	Last Duration
		helloJenkins	16 days - #2	16 days - #1	23 sec

Build Queue: No builds in the queue.

Build Executor Status: 2 helloJenkins #3

# 6. 활용예제



## 6.4 Build Now(2/2)

- 빌드가 정상 수행되면 프로젝트 명 우측 아이콘이 햇님으로 변경되는 것 확인 가능
  - 비정상 수행되면 먹구름 아이콘 보여짐

The screenshot shows the Jenkins web interface. At the top, there's a navigation bar with the Jenkins logo, a red status indicator with the number '5', a search bar, and links for 'test' and 'log out'. Below this is a sidebar with various navigation options like 'New Item', 'People', 'Build History', etc. The main content area displays a table of build records for a project named 'helloJenkins'. The table has columns for 'S' (Success), 'W' (Warning), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. The 'S' column for the 'helloJenkins' row contains a sun icon, which is highlighted with a red box. Below the table, there are links for 'Icon: S M L' and several RSS feeds for all, failures, and latest builds. At the bottom, there are sections for 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing two 'Idle' executors).

S	W	Name ↓	Last Success	Last Failure	Last Duration
		<a href="#">helloJenkins</a>	13 min - #3	16 days - #1	29 sec





# 6. 활용예제



## 6.5 Console Output - 빌드 로그 확인

- 프로젝트 빌드가 정상 수행(2) 된 것은 콘솔 로그(1)를 통해서도 확인 가능

Jenkins > helloJenkins > #2

Back to Project

Status

Changes

**1 Console Output**

View as plain text

Edit Build Information

Delete Build

Git Build Data

No Tags

Previous Build

Executed Gradle Tasks

- help

### Console Output

```
Started by user test
Building in workspace /home/mondayus/.jenkins/workspace/helloJenkins
> git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://yunsd@bitbucket.org/careerlab/bizpp adm
Fetching upstream changes from https://yunsd@bitbucket.org/careerlab/bizpp adm
> git --version # timeout=10
using GIT_ASKPASS to set credentials
> git fetch --tags --progress https://yunsd@bitbucket.org/careerlab/bizpp adm
> git rev-parse refs/remotes/origin/_DEV_^[commit] # timeout=10
> git rev-parse refs/remotes/origin/origin/_DEV_^[commit] # timeout=10
Checking out Revision ba904f0abf5f56314624238f2fe7a80fa5098112 (refs/remotes/or
> git config core.sparsecheckout # timeout=10
> git checkout -f ba904f0abf5f56314624238f2fe7a80fa5098112
Commit message: "워드 변경"
> git rev-list --no-walk ba904f0abf5f56314624238f2fe7a80fa5098112 # timeout=10
[Gradle] - Launching build.
[helloJenkins] $ gradle
Starting a Gradle Daemon (subsequent builds will be faster)
The Task.leftShift(Closure) method has been deprecated and is scheduled to be re
at build_6xr4v5iJoamt5jjo0ef0cj3eo.run(/home/mondayus/.jenkins/workspaci
:help

Welcome to Gradle 3.4.1.

To run a build, run gradle <task> ...

To see a list of available tasks, run gradle tasks

To see a list of command-line options, run gradle --help

To see more detail about a task, run gradle help --task <task>

BUILD SUCCESSFUL 2

Total time: 16.761 secs
Build step 'Invoke Gradle script' changed build result to SUCCESS
Finished: SUCCESS
```



# 6. 활용예제



Jenkins



## 6.6 war확인 - 빌드된 war 확인

- 빌드가 정상적으로 수행 되었다면 ~/.jenkins/workspace/helloJenkins/build/libs 디렉터리 내에 war파일 생성

```
mondayus@mondayus-VirtualBox:~/jenkins/workspace/helloJenkins/build/libs$ ll
total 49600
drwxr-x--- 2 mondayus mondayus 4096 11월 15 06:30 ./
drwxr-x--- 6 mondayus mondayus 4096 11월 15 06:30 ../
-rw-r----- 1 mondayus mondayus 50780918 11월 15 06:30 admin.war
```

- 우리는 war파일이 자동으로 ~/Download/adminTomcat/webapps로 이동하도록 ~/Download/adminTomcat/bin/adminDeploy.sh에서 설정 해 두었으므로 위 war파일은 있는지 확인
- ~/Download/adminTomcat/webapps로 이동해보면 admin.war파일이 정상적으로 이동되어있고 deploy까지 성공해서 admin디렉터리가 생성되어 있는 것 확인 가능

```
mondayus@mondayus-VirtualBox:~/Downloads/adminTomcat/webapps$ ll
total 116712
drwxr-x--- 8 mondayus mondayus 4096 11월 15 06:48 ./
drwxr-xr-x 9 mondayus mondayus 4096 11월 15 06:13 ../
drwxr-x--- 6 mondayus mondayus 4096 11월 15 06:48 admin/
-rw-r----- 1 mondayus mondayus 119476257 11월 15 06:48 admin.war
```







**Q Jenkins가 허드슨과 역 호환이 가능합니까?**

**A** Jenkins는 허드슨과 코드 기반, 개발 방식, 프로젝트의 문화가 동일합니다. 따라서 기존 Hudson 배포 (즉, 버전 1.395 또는 이전 버전)가 Jenkins로 원활하게 업그레이드됩니다. Hudson이 이전 버전과 완전히 호환되는 것처럼 Hudson 및 Jenkins의 이전 버전과 계속 호환됩니다.

**Q Jenkins 커뮤니티에 참여하려면 어떻게해야합니까?**

**A** 방법에는 여러가지가 있습니다. 우리의 메일 링리스트에 가입하거나 버그 추적기에 파일 문제를 추가 또는 Wiki를 업데이트하거나 IRC 채널을 이용 할 수도 있습니다. 또한 트위터에서 우리를 팔로우 하는 방법도 있습니다.

다만, 버그 추적기와 위키 등록은 [jenkins-ci.org](http://jenkins-ci.org)에 계정이 있어야합니다.



# 8. 용어정리



Jenkins



용어	설명
CI	지속적인 통합(Continuous Integration)은 개별 개발자가 작업한 내용을 자주 통합하고 테스트하는 작업 관련 소프트웨어(Jenkins)를 사용하면, 버전 관리와 자동화된 빌드, 테스트, 리포팅을 통해 쉽고 최대한 빨리 오류를 발견하고 발생한 문제를 조기에 처리
IRC	인터넷 릴레이 챗(Internet Relay Chat, IRC)은 실시간 채팅 프로토콜로, 여러 사용자가 모여 대화를 나눌 수 있음
gradle	Gradle은 Groovy를 이용한 빌드 자동화 시스템, Groovy와 유사한 도메인 언어를 채용하였으며, 현재 안드로이드 앱을 만드는데 필요한 안드로이드 스튜디오의 공식 빌드 시스템, Java, C/C++, Python 등과 같은 여러 가지 언어를 지원
Maven	아파치 메이븐(Apache Maven)은 자바용 프로젝트 관리 도구, 아파치 엔트의 대안으로 만들어짐, 아파치 라이선스로 배포되는 오픈 소스 소프트웨어
github	깃허브(GitHub, /'gɪt,hʌb/, 원래 이름: Logical Awesome LLC)는 분산 버전 관리 툴인 깃(Git)을 사용하는 프로젝트를 지원하는 웹호스팅 서비스이다. 루비 온 레일스로 작성, GitHub는 영리적인 서비스와 오픈소스를 위한 무상 서비스를 모두 제공, 2009년의 Git 사용자 조사에 따르면 GitHub는 가장 인기있는 Git 호스팅 사이트, 또한 2011년의 조사에서는 가장 인기있는 오픈 소스 코드 저장소로 꼽힘



# Open Source Software Installation & Application Guide



이 저작물은 크리에이티브 커먼즈 [저작자표시-비영리-동일조건 변경허락 2.0 대한민국 라이선스]에 따라 이용하실 수 있습니다.