

2. Jenkins

■ 사전준비

- ***Centos7 or Ubuntu 16.04***
- ***Java (Oracle or OpenJDK 1.8 higher)***
 - *\$ java -version*
 - *\$ javac -version*
- ***Maven***
 - *\$ yum install maven*
 - *\$ mvn -version*

■ 사전준비

- **Git**
- **Tomcat**
 - \$ wget <http://mirror.apache-kr.org/tomcat/tomcat-8/v8.5.42/bin/apache-tomcat-8.5.42.tar.gz>
 - \$ tar xvfz apache-tomcat-8.5.42.tar.gz
 - \$ mv apache-tomcat-8.5.42 /usr/local
- **환경 변수 설정**
 - \$ gedit /etc/profile
 - JAVA_HOME
 - CATALINA_HOME

■ 사전준비

■ Tomcat 환경 변수 설정

- \$ gedit /usr/local/apache-tomcat-8.5.42/conf/server.xml

(line : 69)

<Connector port="8080" protocol="HTTP/1.1"

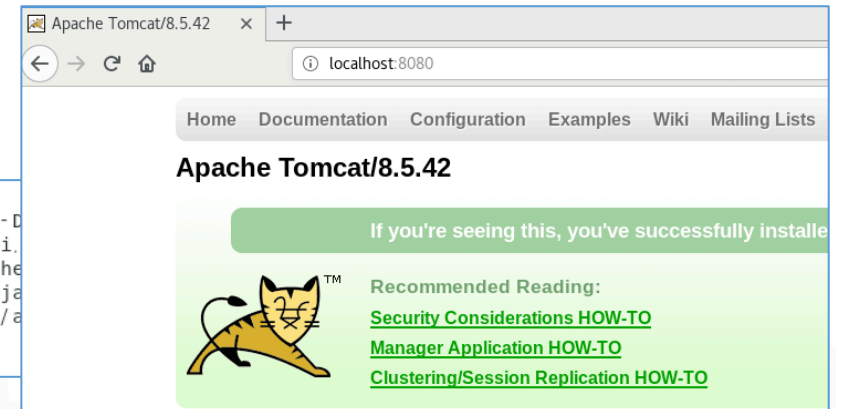
URIEncoding="UTF-8"

connectionTimeout="20000" redirectPort="8443" />

■ Tomcat 실행 및 테스트

- \$ /usr/local/apache-tomcat-8.5.42/bin/startup.sh
- \$ ps -ef | grep tomcat

```
[admin@centos7 bin]$ ps -ef | grep tomcat
root      20815      1  0  6월 18 ?        00:00:56 /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.212.b04-0.el7_6.x86_64/bin/java -D
. logging.config.file=/usr/local/apache-tomcat-8.5.42/conf/logging.properties -Djava.util.logging.manager=org.apache.juli
erLogManager -Djdk.tls.ephemeralDHKeySize=2048 -Djava.protocol.handler.pkgs=org.apache.catalina.webresources -Dorg.apache
. security.SecurityListener.UMASK=0027 -Dignore.endorsed.dirs=-classpath /usr/local/apache-tomcat-8.5.42/bin/bootstrap.jar
cal/apache-tomcat-8.5.42/bin/tomcat-juli.jar -Dcatalina.base=/usr/local/apache-tomcat-8.5.42 -Dcatalina.home=/usr/local/a
cat-8.5.42 -Djava.io.tmpdir=/usr/local/apache-tomcat-8.5.42/temp org.apache.catalina.startup.Bootstrap start
admin     45033   9183  0 00:03 pts/0    00:00:00 grep --color=auto tomcat
```



■ 사전준비

■ Tomcat 서비스 등록

- \$ /usr/local/apache-tomcat-8.5.42/bin/catalina.sh stop
- \$ gedit /usr/lib/systemd/system/tomcat.service

```
[Unit]
Description=tomcat8
After=network.target syslog.target

[Service]
Type=forking
Environment=JAVA_HOME={JAVA_HOME}
User=root
Group=root

ExecStart={CATALINA_HOME}/bin/startup.sh
ExecStop={CATALINA_HOME}/bin/shutdown.sh
```

```
UMask=0007
RestartSec=10
Restart=always

SuccessExitStatus=143
[Install]
WantedBy=multi-user.target
```

■ 사전준비

■ Tomcat 외부 접속 설정

- \$ gedit /etc/firewalld/zones/public.xml → **PORT 추가**
- \$ firewall-cmd --permanent --zone=public --add-po=8080/tcp
- \$ sudo firewall-cmd --permanent --zone=public --add-po=8080/tcp
- firewall-cmd --reload
- firewall-cmd --state

■ 사전준비

■ Tomcat Manager 설정

- \$ gedit /usr/local/apache-tomcat-8.5.42/conf/tomcat-users.xml

```
<role rolename="manager"/>
<role rolename="manager-gui" />
<role rolename="manager-script" />
<role rolename="manager-jmx" />
<role rolename="manager-status" />
<role rolename="admin"/>
<user username="admin" password="manager"
      roles="admin,manager,manager-gui, manager-script, manager-jmx, manager-status"/>
```

- \$ gedit /usr/local/apache-tomcat-8.5.42/conf/Catalina/localhost/manager.xml

```
<Context privileged="true" antiResourceLocking="false" docBase="${catalina.home}/webapps/manager">
    <Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="^.*$" />
</Context>
```

- \$ systemctl restart tomcat.service → ***http://localhost:8080/manager/html***

■ Jenkins 설치

■ Jenkins

- **지속적인 통합**
 - 프로젝트 빌드
 - 테스트
 - 배포
- **Maven, Ant, Gradle, Junit, Nexus**
- **Git, SVN**
- **Java, Python, Node.js**

→ Jenkins는 **Work flow**를 제어

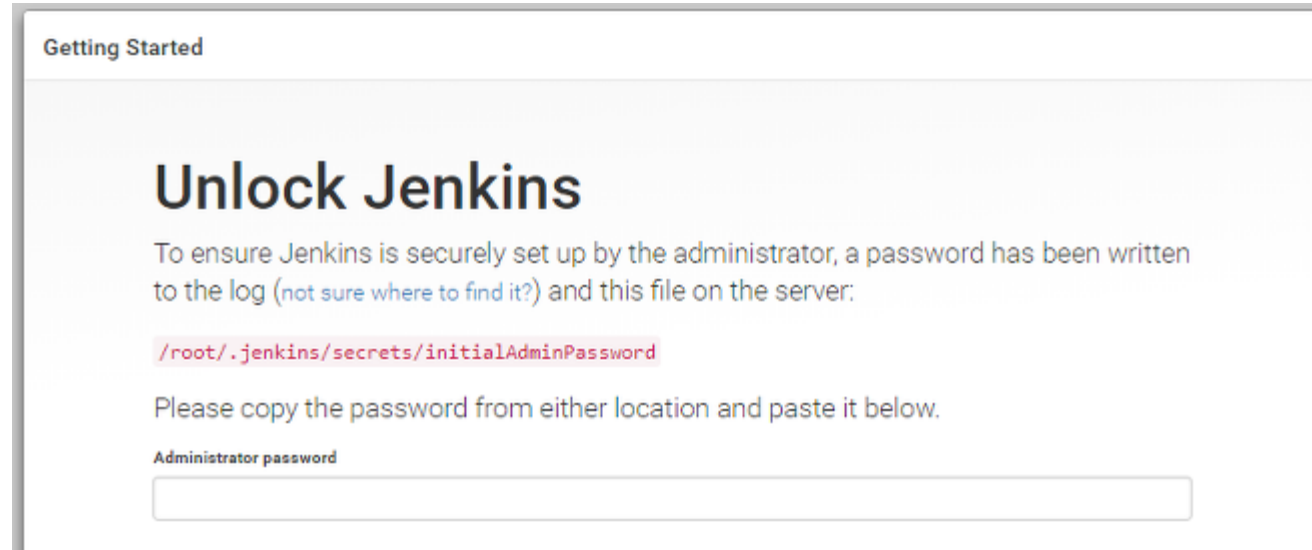
■ Jenkins 설치

■ Jenkins 설치 및 설정

- \$ yum install jenkins
- \$ gedit /etc/sysconfig/jenkins → **JENKINS_PORT="9090"**
- \$ firewall-cmd --permanent --zone=public --add-port=9090/tcp
- \$ firewall-cmd --reload
- \$ service jenkins start
- \$ service jenkins stop
- \$ service jenkins restart

■ Jenkins 설치

■ 브라우저에서 Jenkins 설정



- \$ cat /var/lib/jenkins/secrets/initialAdminPassword
 - 비밀번호 입력 후 로그인 → 설치 계속

■ Jenkins 설치

■ 관리자 계정 생성 (Admin)

Getting Started

Create First Admin User

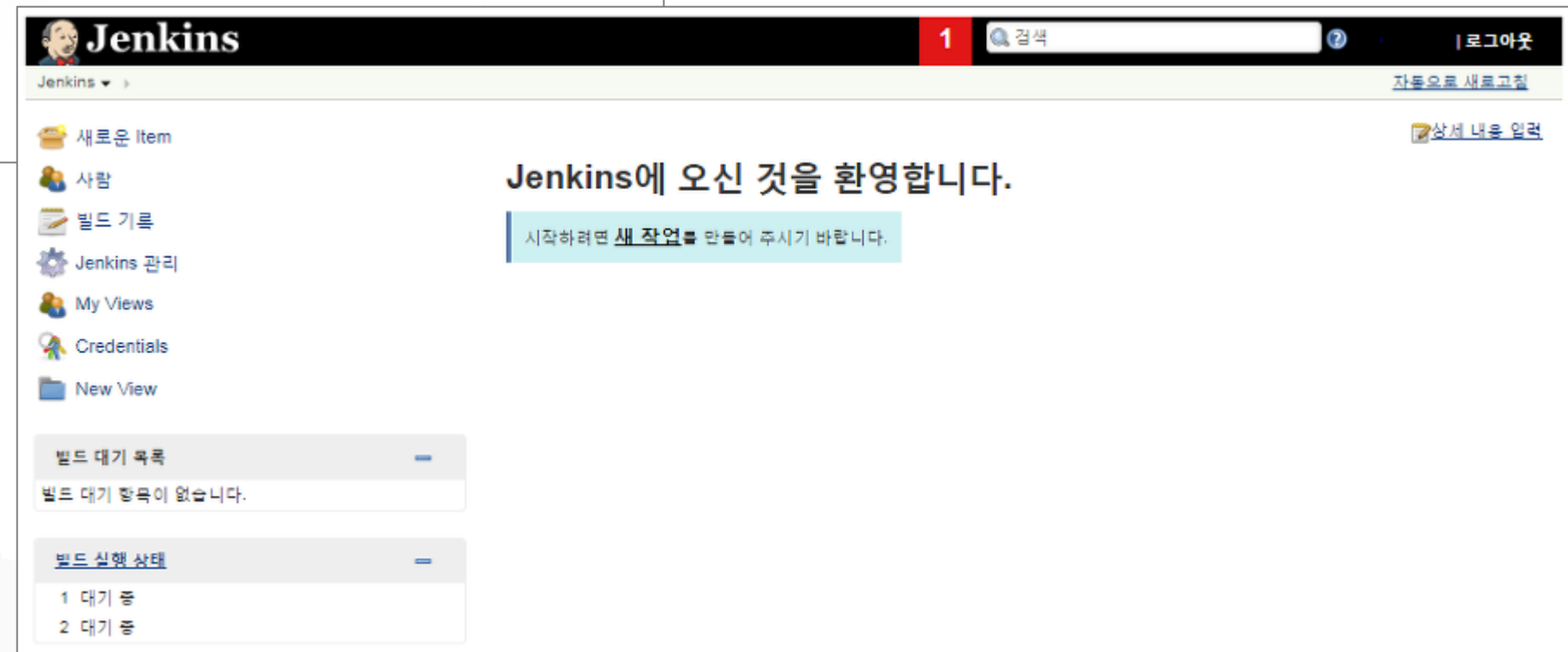
계정명:

암호:

암호 확인:

이름:

이메일 주소:



The screenshot shows the Jenkins dashboard after successful installation. The top navigation bar includes the Jenkins logo, a search bar, and a '로그아웃' (Logout) button. A red box with the number '1' highlights the search bar. The main content area displays a welcome message: 'Jenkins에 오신 것을 환영합니다.' (Welcome to Jenkins) and a tip: '시작하려면 새 작업을 만들어 주시기 바랍니다.' (Please create a new job to get started). On the left sidebar, there are links for '새로운 Item' (New Item), '사람' (People), '빌드 기록' (Build History), 'Jenkins 관리' (Jenkins Management), 'My Views', 'Credentials', and 'New View'. Below these, there are sections for '빌드 대기 목록' (Build Queue) and '빌드 실행 상태' (Build Execution Status).

Jenkins

1 검색

로그아웃

자동으로 새로고침

상세 내용 열람

Jenkins에 오신 것을 환영합니다.

시작하려면 **새 작업**을 만들어 주시기 바랍니다.

- 새로운 Item
- 사람
- 빌드 기록
- Jenkins 관리
- My Views
- Credentials
- New View

빌드 대기 목록

빌드 대기 항목이 없습니다.

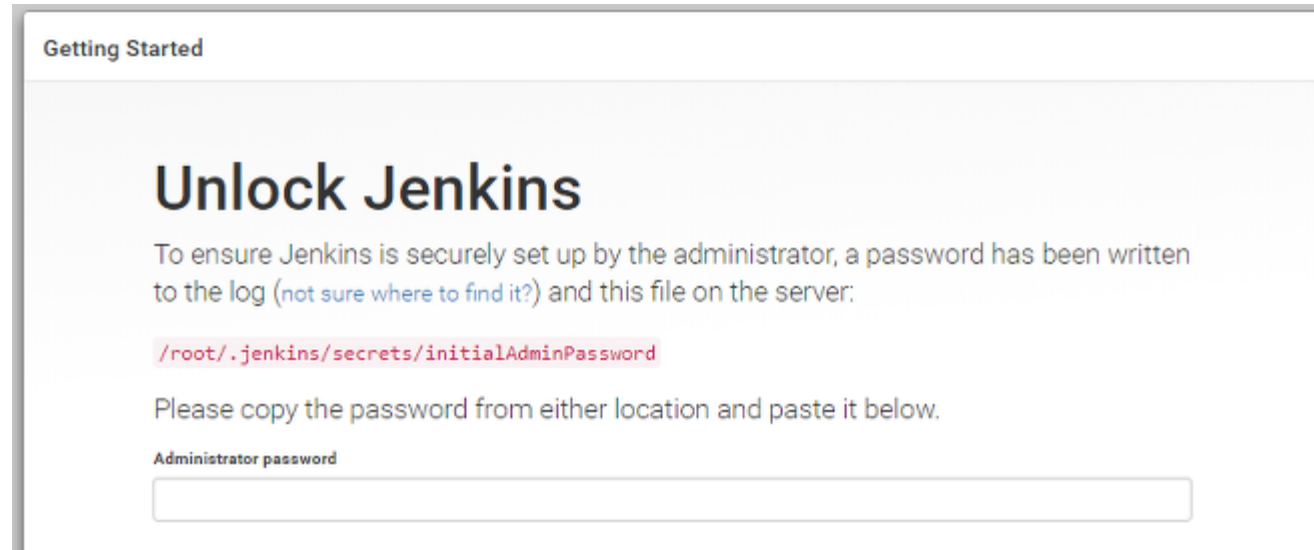
빌드 실행 상태

- 1 대기 중
- 2 대기 중

■ Dashboard 확인

■ Jenkins 설치

■ 브라우저에서 Jenkins 설정



- \$ cat /var/lib/jenkins/secrets/initialAdminPassword
 - 비밀번호 입력 후 로그인 → 설치 계속

■ Jenkins 배포

- Jenkins에서 사용할 애플리케이션 준비 → 웹 애플리케이션
 - *Git Repository에 Push*

1) Jenkins 관리 – Global Tool Configuration

Jenkins 관리

역방향 프록시 설정이 잘못된 것으로 파악되었습니다. [추가 정보](#) [해제](#)

- 시스템 설정
환경변수 및 경로 정보들을 설정합니다.
- 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 modified config files directly on disk.
- 플러그인 관리
Jenkins의 기능을 확장하기 위한 플러그인을 추가, 제거, 사용, 미사용으로 설정할 수 있습니다.

- *JDK*
- *Git*
- *Maven*

■ Jenkins 배포

- Jenkins에서 사용할 애플리케이션 준비 → 웹 애플리케이션

2) JDK 설정

- Add JDK

JDK

JDK installations

Add JDK

JDK

Name

jdk1.8

JAVA_HOME

/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.212.b04-0.el7_6.x86_64

☐ Install automatically

?

Delete JDK

3) Git 설정

Git

Git installations

Git

Name

git1.8.3.1

Path to Git executable

git

☐ Install automatically

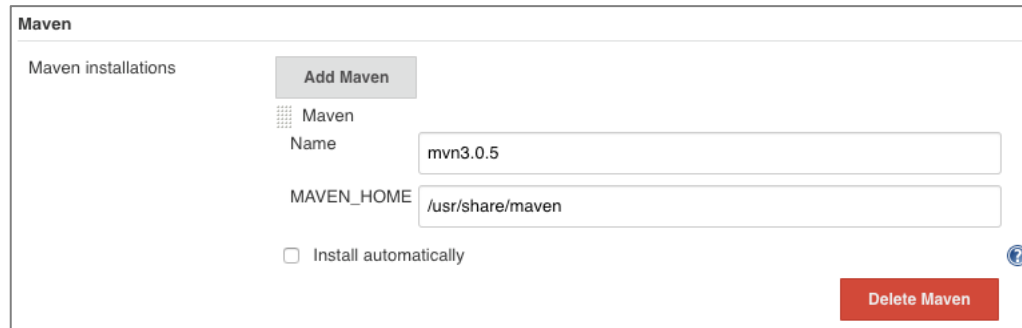
?

Delete Git

■ Jenkins 배포

4) JDK 설정

- Add Maven



The screenshot shows the 'Maven' configuration page in Jenkins. It features a table with one row for 'Maven'. The 'Name' field is set to 'mvn3.0.5' and the 'MAVEN_HOME' field is set to '/usr/share/maven'. There is an 'Add Maven' button at the top right of the table and a 'Delete Maven' button at the bottom right. A checkbox for 'Install automatically' is also present.

Maven									
Maven installations	<div><div>Add Maven</div><table><tr><td>Maven</td><td></td></tr><tr><td>Name</td><td>mvn3.0.5</td></tr><tr><td>MAVEN_HOME</td><td>/usr/share/maven</td></tr><tr><td><input type="checkbox"/> Install automatically</td><td></td></tr></table><div>Delete Maven</div></div>	Maven		Name	mvn3.0.5	MAVEN_HOME	/usr/share/maven	<input type="checkbox"/> Install automatically	
Maven									
Name	mvn3.0.5								
MAVEN_HOME	/usr/share/maven								
<input type="checkbox"/> Install automatically									

5) 새로운 Item

- Item 이름 → First Demo

- Freestyle project

- OK

■ Jenkins 배포

6) 소스 코드 관리 탭

- Git 주소 입력

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

소스 코드 관리

☐ None
☒ Git

Repositories

Repository URL

Credentials

Branches to build

Branch Specifier (blank for 'any')

■ Jenkins 배포

6) 소스 코드 관리/ 탭

- Build

- Goals → *clean package tomcat:redeploy -P production -D maven.test.skip=true*

Build

 Invoke top-level Maven targets

Maven Version

mvn3.0.5

Goals

clean package tomcat:redeploy -P production

고급...

Add build step ▼

■ Jenkins 배포

7) Build

- Dashboard → Build Now

대시보드로 돌아가기

상태

변경사항

작업공간

Build Now

Project 삭제

구성

Rename

Build History 추이

find x

#7 2019. 6. 18 오전 1:16

프로젝트로 돌아가기

상태

바뀐점

Console Output

View as plain text

빌드 정보 수정

Delete build '#7'

Git Build Data

No Tags

이전 빌드

콘솔 출력

```
Started by user Administrator
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/test
No credentials specified
> git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/jonecons
Fetching upstream changes from https://github.com/jonecons
> git --version # timeout=10
> git fetch --tags --progress https://github.com/jonecons
+refs/heads/*:refs/remotes/origin/*
> git rev-parse refs/remotes/origin/master^{commit} # tim
> git rev-parse refs/remotes/origin/origin/master^{commit}
Checking out Revision 839850alf330e91fb84e02513c86d8ab11c4
```

■ Jenkins 배포

7) Build

- **Error!**
 - **Caused: java.io.IOException: Cannot run program "mvn" (in directory "/root/.jenkins/workspace/guestbook"): error=2, 그런 파일이나 디렉터리가 없습니다**
 - **UTF-8 인코딩**
 - **profile "prodiction"**
 - **javax package (GlobalExceptionHandler) ...**

■ Jenkins 배포

8) Error 해결 - pom.xml

```
<properties>
  <org.springframework-version> 4.2.1.RELEASE </org.springframework-version>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
</properties>
```

```
<profiles>
  <profile>
    <id>production</id>
    <build>
      <resources>
        <resource>
          <directory>${project.basedir}/src/main/resources</directory>
          <excludes>
            <exclude>**/*.java</exclude>
          </excludes>
        </resource>
      </resources>
    </build>
  </profile>
</profiles>
```

<계속>

■ Jenkins 배포

8) Error 해결 - pom.xml

```
<plugins>
  <plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-resources-plugin</artifactId>
    <configuration>
      <encoding>UTF-8</encoding>
    </configuration>
  </plugin>
</plugins>
</build>
```

<계속>

■ Jenkins 배포

8) Error 해결 - pom.xml

```
<dependencies>
  <dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>3.0.1</version>
    <scope>provided</scope>
  </dependency>
  <dependency>
    <groupId>javax.servlet.jsp</groupId>
    <artifactId>jsp-api</artifactId>
    <version>2.0</version>
    <scope>provided</scope>
  </dependency>
</dependencies>
</profile>
</profiles>
```

<계속>

■ Jenkins 배포

8) Error 해결 - pom.xml

```
<build>
  <plugins>
    <plugin>
      ...
    </plugin>
    <plugin>
      <groupId>org.codehaus.mojo</groupId>
      <artifactId>tomcat-maven-plugin</artifactId>
      <configuration>
        <url>http://localhost:8080/manager/text</url>
        <path>/guestbook</path>
        <username>admin</username>
        <password>manager</password>
      </configuration>
    </plugin>
  </plugins>
</build>
```

■ Jenkins 배포

9) Re-build

```
[INFO] --- tomcat-maven-plugin:1.1:redeploy (default-cli) @ guestbook ---  
[INFO] Deploying war to http://localhost:8080/guestbook  
[INFO] OK - Deployed application at context path [/guestbook]  
[INFO] -----  
[INFO] BUILD SUCCESS  
[INFO] -----  
[INFO] Total time: 13.634s  
[INFO] Finished at: Tue Jun 18 01:16:41 KST 2019  
[INFO] Final Memory: 16M/82M  
[INFO] -----  
Finished: SUCCESS
```

 새로운 Item 사람 빌드 기록 Jenkins 관리 My Views Lockable Resources 상세 내용 입력

All

+

S	W	Name ↓	최근 성공	최근 실패	최근 소요 시간	
		test	36 sec - #8	23 hr - #6	19 sec	

아이콘: [S](#) [M](#) [L](#)[Legend](#) [RSS 모두](#) [RSS 실패](#) [RSS 최근 빌드](#)

■ Jenkins 배포

Q) 위 테스트 실행하도록 수정

- Test Case 추가
- Jenkins 설정 수정
- Re-build