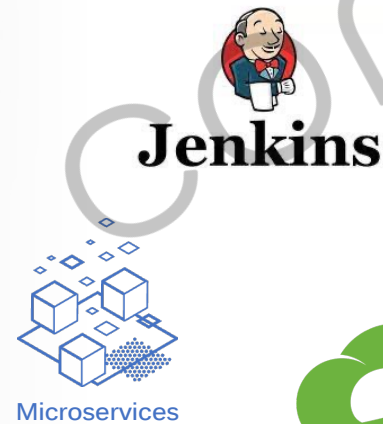


# Jenkins를 이용한 CI/CD Pipeline 구축



Spring  
Cloud

```
public static void main(String[] args) {
    // ...
}

</servlet>
<servlet-name>BoardController</servlet-name>
<servlet-class>com.joneconsulting.controller.BoardController</servlet-class>
<init-param>
<param-name>user_name</param-name>
<param-value>Kenneth Lee</param-value>
</init-param>
</servlet>

// ...
}

@Interface NextInnovationDelegate : NSObject <UIApplicationDelegate>
```

## 프로필

Dowon Lee



지식공유자 인증

5452 ☆ 4.8(420)

멘토링 활성



- 홈
- 강의
- 로드맵
- 수강후기
- 블로그

### 강의 (3)

최신순 ▼

20% 할인 (D-8)

Spring Cloud  
with Microservices  
Part 2

Spring Cloud로 개발하는 마이크로서비스 애플리케이션(MSA)

Dowon Lee

★★★★★ (174)

학습중

+3000명 독점 할인중

20% 할인 (D-8)

Spring Boot  
with RESTful Web Services  
Part 1

Spring Boot를 이용한 RESTful Web Services 개발

Dowon Lee

★★★★★ (308)

학습중

+2700명 독점 할인중

IntelliJ IDEA  
Java Web Programming

웹 애플리케이션 개발을 위한 IntelliJ IDEA 설정

Dowon Lee

★★★★★ (228)

학습중

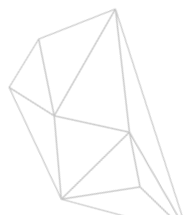
+3500명



## 목차

---

- Section 1: DevOps와 CI/CD
- Section 2: Jenkins를 이용한 CI/CD 사용
- Section 3: Jenkins + Infrastructure as Code
- Section 4: Jenkins + Ansible + Kubernetes 연동
- **Section 5: Advanced Jenkins 사용**
- Section 6: Public Cloud에 배포
- Appendix



## Section 5.

# Advanced Jenkins

- Delivery Pipeline 사용
- Jenkins Pipeline 구성
- SonarQube 사용
- Jenkins + SonarQube 연동
- Jenkins Master + Slaves 구성



- <https://www.sonarqube.org/>
- Continuous Integration + Analysis
  - Code Quality Assurance tool → Issues + Defect + Code Complexity
  - Detect Bugs & Vulnerabilities
  - Track Code Smells
  - Code Quality Metrics & History
- 17 languages 지원
  - Java, C#, JavaScript, TypeScript, CloudFormation, Terraform, Kotlin, Ruby, Go, Scala, Flex, Python, PHP, HTML, CSS, XML, VB.NET
- CI/CD integration
- Extensible, with 50+ community plugins



# Install SonarQube

njone company

- <https://www.sonarqube.org/downloads/>

Version: 9.5 | Release: June 2022 | [Getting Started](#) | [Release Notes](#) | [Upgrade Notes](#) | [Available From DockerHub](#)

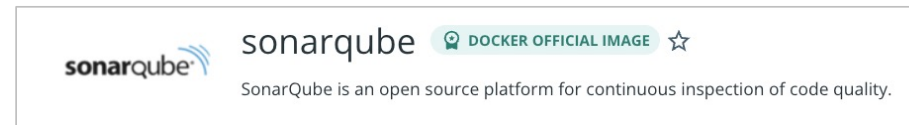
<div><h2>Community</h2><p>EDITION</p><p>Used and loved by 200,000+ companies</p><p><b>FREE &amp; OPEN SOURCE</b></p><p><a href="#">Download for free</a></p><p><b>All the following features:</b></p><ul style="list-style-type: none"><li>✓ Static code analysis for 17 languages <small>Java, C#, JavaScript, TypeScript, CloudFormation, Terraform, Kotlin, Ruby, Go, Scala, Flex, Python, PHP, HTML, CSS, XML and VB.NET</small></li><li>✓ Detect Bugs &amp; Vulnerabilities</li><li>✓ Review Security Hotspots</li><li>✓ Track Code Smells &amp; fix your Technical Debt</li><li>✓ Code Quality Metrics &amp; History</li><li>✓ CI/CD integration</li><li>✓ Extensible, with 50+ community plugins</li></ul></div>	<div><h2>Developer</h2><p>EDITION</p><p>Built for developers by developers</p><p><a href="#">Download</a></p><p><b>Community Edition plus:</b></p><ul style="list-style-type: none"><li>✓ C, C++, Obj-C, Swift, ABAP, T-SQL, PL/SQL support</li><li>✓ Detection of Injection Flaws in Java, C#, PHP, Python, JavaScript, TypeScript</li><li>✓ Analysis of feature and maintenance branches</li><li>✓ Pull Request decoration for:<ul style="list-style-type: none"><li>GitHub</li><li>Bitbucket</li><li>Azure DevOps</li><li>GitLab</li></ul></li></ul></div>	<div><h2>Enterprise</h2><p>EDITION</p><p>Designed to meet Enterprise Requirements</p><p><a href="#">Download</a></p><p><b>Developer Edition plus:</b></p><ul style="list-style-type: none"><li>✓ Portfolio Management &amp; PDF Executive Reports</li><li>✓ Project PDF reports</li><li>✓ Security Reports</li><li>✓ Project Transfer</li><li>✓ Parallel processing of analysis reports</li><li>✓ Support for Apex, COBOL, PL/I, RPG, VB6</li></ul></div>	<div><h2>Data Center</h2><p>EDITION</p><p>Designed for High Availability</p><p><a href="#">Download</a></p><p><b>Enterprise Edition plus:</b></p><ul style="list-style-type: none"><li>✓ Component redundancy</li><li>✓ Data resiliency</li><li>✓ Horizontal Scalability</li></ul></div>
---	---	---	--

# Install SonarQube

njone company

- [https://hub.docker.com/\\_/sonarqube](https://hub.docker.com/_/sonarqube)

- Windows, MacOS) **docker pull sonarqube**



```
$ docker run --rm -p 9000:9000 --name sonarqube sonarqube
```

- Apple Silicon, ARM, M1) **docker pull edowon0623/sonarqube:arm**

```
$ docker run --rm -p 9000:9000 --name sonarqube edowon0623/sonarqube:arm
```

edowon0623 / sonarqube

Description

This repository does not have a description

Tags and Scans

This repository contains 1 tag(s).

TAG	OS	PULLED	PUSHED
arm			2 minutes ago

VULNERABILITY SCANNING - DISABLED [Enable](#)

Docker commands

To push a new tag to this repository,

```
docker push edowon0623/sonarqube:tagname
```

Automated Builds

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions.



# Exercise 14 # Sonarqube + Maven project 1/4

njone company

## ■ Maven Project에 Plugin 설정 추가

- <https://docs.sonarqube.org/latest/analysis/scan/sonarscanner-for-maven/>

### | How to Fix Version of Maven Plugin

It is recommended to lock down versions of Maven plugins:

```
<build>
  <pluginManagement>
    <plugins>
      <plugin>
        <groupId>org.sonarsource.scanner.maven</groupId>
        <artifactId>sonar-maven-plugin</artifactId>
        <version>3.7.0.1746</version>
      </plugin>
    </plugins>
  </pluginManagement>
</build>
```

- pom.xml 파일에 추가
- Version 변경



## Exercise 14 # Sonarqube + Maven project 2/4

njone company

- SonarQube token 생성
  - <http://localhost:9000>

Log In to SonarQube

Login	<b>admin</b>
Password	<b>admin</b>

Update your password

This account should not use the default password.

Enter a new password

All fields marked with \* are required

Old Password \*

New Password \*

Confirm Password \*

# Exercise 14 # Sonarqube + Maven project 3/4

njone company

## ■ SonarQube token 생성

- My Account → Security → User Token 생성

The screenshot displays the SonarQube web interface. At the top, a search bar and a user profile dropdown are visible. The user profile dropdown is open, showing options: Administrator, My Account (highlighted with a red dashed box), and Log out. Below the search bar, the main navigation menu includes Profile, Security (highlighted with a red dashed box), Notifications, and Projects. The 'Security' tab is active, showing the 'Tokens' section. This section contains a text box with 'sonar-token', a dropdown menu set to 'User Token', and a 'Generate' button (all three are highlighted with a red dashed box). Below this, a table with columns 'Name', 'Type', 'Project', 'Last use', and 'Created' is shown, currently displaying 'No tokens'.

Administrator

My Account

Log out

Profile Security Notifications Projects

Tokens

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

Generate Tokens

sonar-token User Token Generate

Name	Type	Project	Last use	Created
No tokens				

# Exercise 14 # Sonarqube + Maven project 4/4


njone company





## ■ Maven build

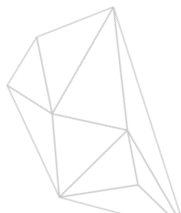
\$ mvn sonar:sonar -D**sonar.host.url=http://IP\_address:9000** -D**sonar.login=[the-sonar-token]**

```
[INFO] ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=com.njonecompany.web%3Aweb
[INFO] Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[INFO] More about the report processing at http://localhost:9000/api/ce/task?id=AYJr6jlkWTD7cEa5JiF
[INFO] Analysis total time: 7.602 s
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 10.154 s
[INFO] Finished at: 2022-08-05T11:51:28+09:00
[INFO] -----
```

## ■ SonarQube Projects 확인

 **cicd-web-project maven webapp** Passed Last analysis: 2 minutes ago

 Bugs	 Vulnerabilities	 Hotspots Reviewed	 Code Smells	Coverage	Duplications	Lines
<span>C</span>	<span>A</span>	<span>A</span>	<span>A</span>	0.0%	0.0%	166 <span>XS</span> XML, Java...



## Exercise 15 # Bad code detected & Rectified 1/3

njone company

- 샘플 프로젝트에 아래 코드 추가

```
16     @GetMapping("/")
17     public String index(Model model) {
18         // logger.debug("Welcome to njonecompany.com...");
19         System.out.println(x: "index is called by GET /");
20     }
```

\$ mvn clean compile package -DskipTests=true

\$ mvn sonar:sonar -D**sonar.host.url=http://IP:9000** -D**sonar.login=[the-sonar-token]**

# Exercise 15 # Bad code detected & Rectified 2/3

njone company

## ■ SonarQube Projects 확인

The screenshot displays the SonarQube web interface. On the left sidebar, a list of issues is shown for the file `...com/njonecompany/web/WebInitializer.java`. The issue **Replace this use of System.out or System.err by a logger.** is highlighted with a blue border. Below it, other issues include **Return an empty array instead of null.**, **Remove this unused "logger" private field.**, and **This block of commented-out lines of code should be removed.**

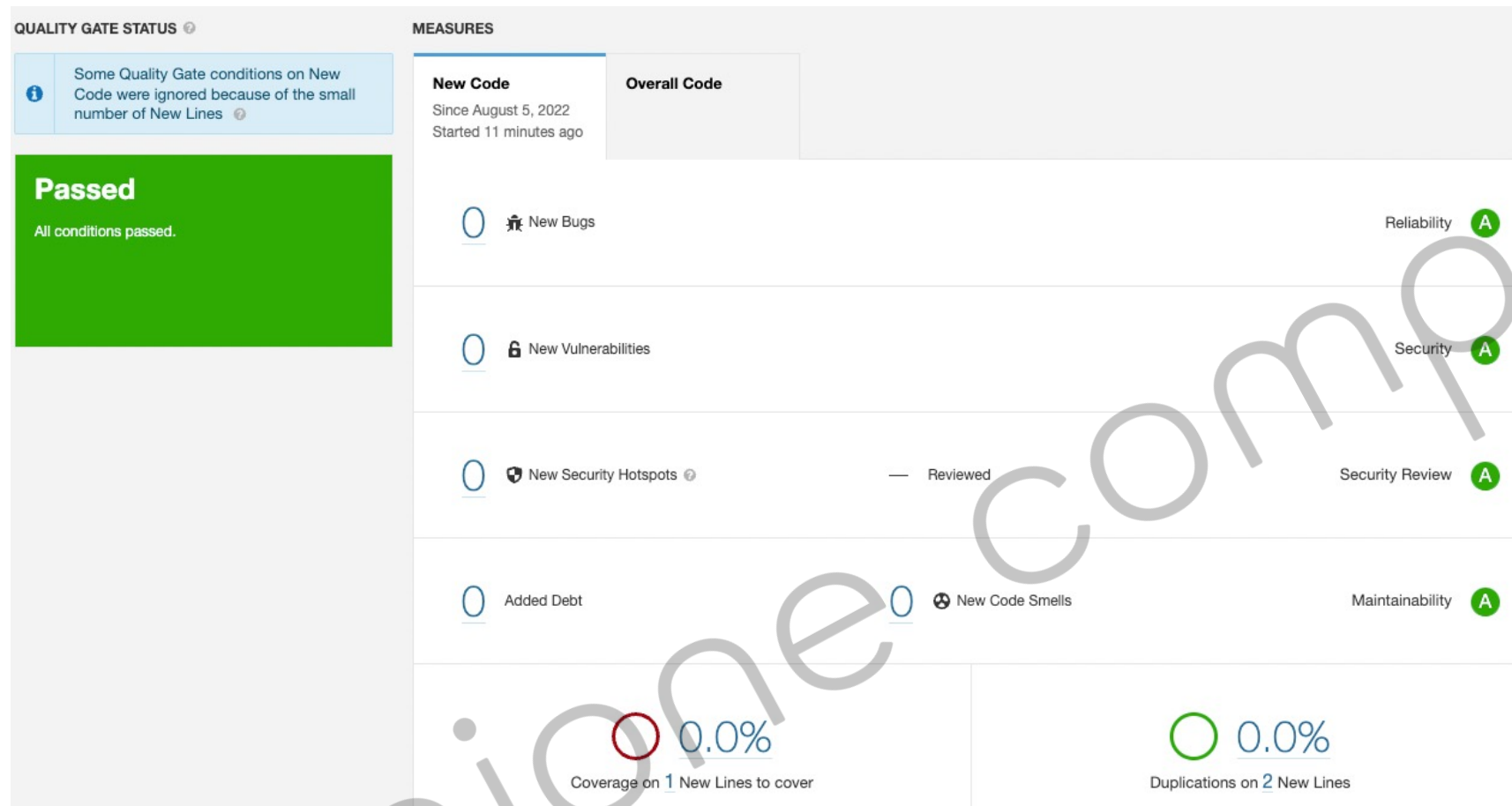
The main panel shows the source code of `src/main/java/com/njonecompany/web/controller/WelcomeController.java`. The code includes a `private final Logger logger` field and a `@GetMapping("/")` method. A red vertical bar on the left indicates the location of the selected issue. The issue details panel on the right shows the message **Replace this use of System.out or System.err by a logger.** and provides options to **Code Smell**, **Major**, **Open**, **Not assigned**, and **10min effort**. It also includes a **Comment** field and a **bad-practice, cert, owasp-a3** tag.

```
14 ... private final Logger logger = LoggerFactory.getLogger(WelcomeController.class);
15
16 @GetMapping("/")
17 public String index(Model model) {
18 ... // logger.debug("Welcome to njonecompany.com...");
19
20 System.out.println("index is called by GET /");
```

# Exercise 15 # Bad code detected & Rectified 3/3

njone company

- System.out.println → Slf4g Logger 로 변경 후 다시 빌드



# Jenkins + SonarQube

njone company

- Manage Jenkins → Manage Plugins → Available
  - SonarQube Scanner 설치

업데이트된 플러그인 목록

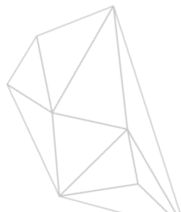
설치 가능

설치된 플러그인 목록

고급

Q Sonarqube

Install	Name ↓	Released
<input checked="" type="checkbox"/>	<div><div>SonarQube Scanner 2.14</div><div>External Site/Tool IntegrationsBuild Reports</div><div>This plugin allows an easy integration of <a href="#">SonarQube</a>, the open source platform for Continuous Inspection of code quality.</div></div>	8 mo 25 days ago





# Jenkins + SonarQube

njone company

- Manage Jenkins → Plugin Manager → Available
  - SonarQube Scanner for Jenkins
- Manage Jenkins → Manage Credentials → Add Credentials

SonarQube User Token

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

↑ Back to credential domains

+ Add Credentials

### New credentials

Kind

Secret text

Scope ?

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

Secret

.....

ID ?

sonarqube-token



# Jenkins + SonarQube

njone company

- Manage Jenkins → Configure System → SonarQube servers
  - Environment variables 선택
  - Name: **SonarQube 서버의 이름지정**
  - Server URL: **SonarQube 서버의 IP address**
  - Server authentication token: **Credentials에서 지정한Token 정보**

Name

SonarQube-server

Server URL

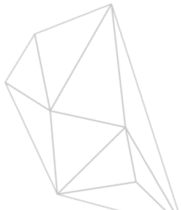
Default is http://localhost:9000

http://172.17.0.5:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

sonarqube-token



## Exercise 16 # Using SonarQube in Pipeline 1/2

njone company

### ■ My-First-Pipeline 수정

```
...  
stage('SonarQube analysis') {  
  steps {  
    withSonarQubeEnv('SonarQube-server') {  
      sh 'mvn sonar:sonar'  
    }  
  }  
}  
...
```

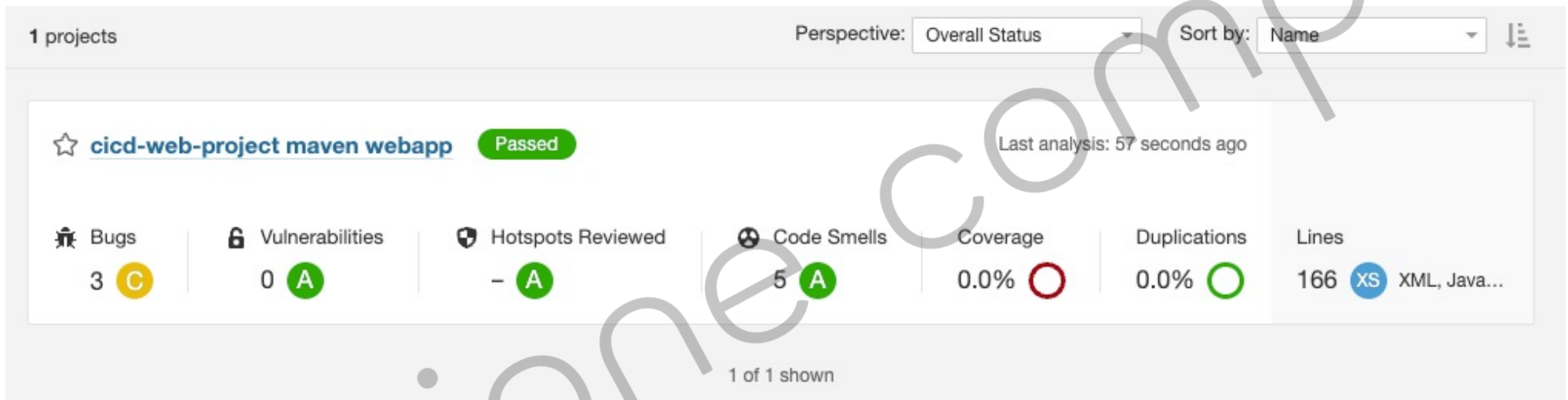
Name
SonarQube-server
Server URL
Default is http://localhost:9000
http://172.17.0.5:9000
Server authentication token
SonarQube authentication token. Mandatory when anonymous access is disabled.
sonarqube-token

# Exercise 16 # Using SonarQube in Pipeline 1/2

njone company

## ■ Build Now

```
[INFO] ANALYSIS SUCCESSFUL, you can find the results at: http://172.17.0.5:9000/dashboard?id=com.njonecompany.web%3Aweb
[INFO] Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[INFO] More about the report processing at http://172.17.0.5:9000/api/ce/task?id=AYJsn3WX5zij2QcD6GFy
[INFO] Analysis total time: 3.997 s
[INFO] -----
[INFO] BUILD SUCCESS
```



- Jenkins Master

- 기본 Jenkins 서버 → Master node
- 빌드 작업 예약 및 빌드 실행
- Slave에게 Build 할 Job 전달
- Slave 모니터링
- 빌드 결과 기록

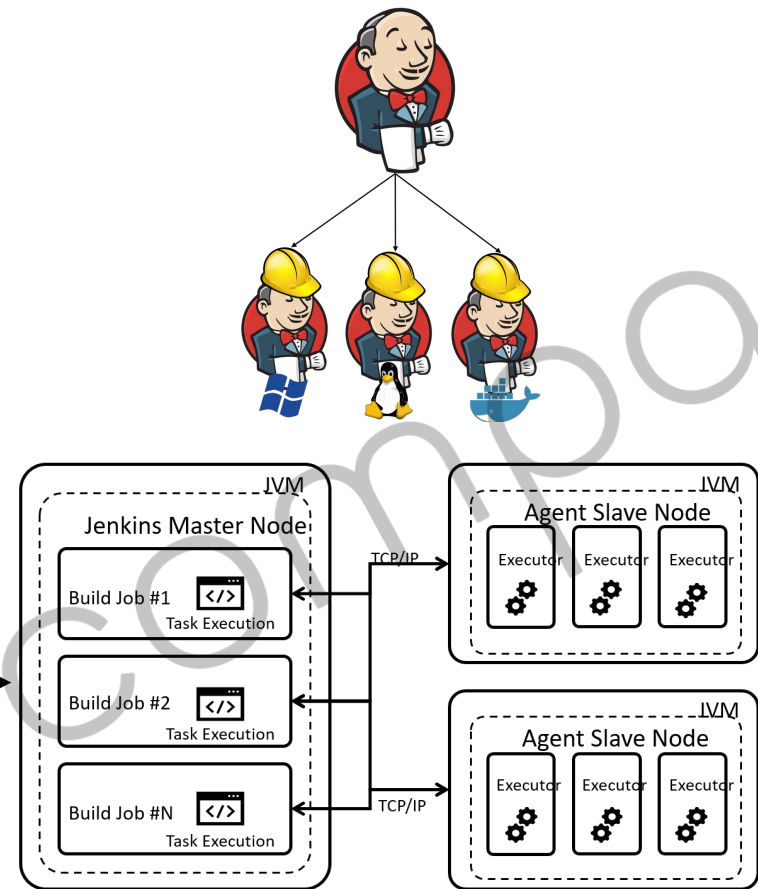
[illegible]

# Jenkins Master + Slaves

njone company

## ■ Jenkins Slave

- Remote에서 실행되는 Jenkins 실행 Node
- Jenkins Master의 요청 처리
- Master로부터 전달된 Job 실행
- 다양한 운영체제에서 실행 가능
- Jenkins 프로젝트 생성 시 특정 Slave를 선택하여 실행 가능



<https://digitalvarys.com/how-to-configure-jenkins-master-slave-setup/>



## Exercise 17 # Add a slave node 1/3

njone company

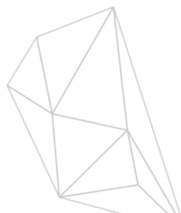
- 새로운 Server 추가

### Windows, MacOS)

```
$ docker run --privileged --name jenkins-node1 -itd -p 30022:22 \  
-e container=docker -v /sys/fs/cgroup:/sys/fs/cgroup --cgroupns=host \  
edowon0623/docker:latest /usr/sbin/init
```

### Apple silicon, m1)

```
$ docker run --privileged --name jenkins-node1 -itd -p 30022:22 \  
-e container=docker -v /sys/fs/cgroup:/sys/fs/cgroup --cgroupns=host \  
edowon0623/docker-server:m1 /usr/sbin/init
```



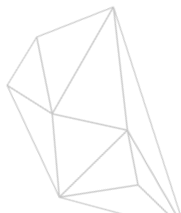




## Exercise 17 # Add a slave node 2/3

njone company

- Manage Jenkins → Manage Nodes → New Node
  - Node Name: ***slave1***
  - Description: ***Add a server as a slave machine***
  - Number of executors: **5**
  - Remote root directory: ***/root/slave1***
  - Labels: ***slave1***
  - Usage: ***Use this node as much as possible***
  - Launch method: ***Launch agents via SSH***
    - Host: ***[Slave server IP address] es) 172.17.0.3***
    - Port: **22**
    - Credentials: ***root/P@ssw0rd***



## Exercise 17 # Add a slave node 3/3

njone company

- My-First-Project 수정
  - **Restrict where this project can be run 선택**

☒ Restrict where this project can be run ?

Label Expression ?

slave1

Label slave1 matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

### ✓ 콘솔 출력

```
Started by user Administrator
Running as SYSTEM
Building remotely on slavel in workspace /root/slavel/workspace/My-First-Project
[My-First-Project] $ /bin/sh -xe /tmp/jenkins9356691432887153209.sh
+ echo 'Welcome to my first project using Jenkins'
Welcome to my first project using Jenkins
+ javac -version
javac 11.0.13
Triggering a new build of My-Second-Project
Finished: SUCCESS
```

```
[root@897f059a1ce0 workspace]# pwd
/root/slave1/workspace
[root@897f059a1ce0 workspace]# ls -al
total 16
drwxr-xr-x 4 root root 4096 Aug  5 05:05 .
drwxr-xr-x 5 root root 4096 Aug  5 05:06 ..
drwxr-xr-x 2 root root 4096 Aug  5 05:05 My-First-Project
drwxr-xr-x 5 root root 4096 Aug  5 05:06 My-Second-Project
```



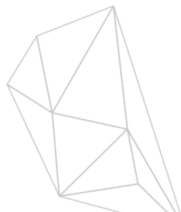
## Exercise 18 # Execute a pipeline on slaves 1/5

njone company

- New Item → Pipeline → My-Second-Pipeline
  - Copy from: **My-First-Pipeline**

```
pipeline {
  agent {
    label 'slave1'
  }
  tools {
    maven 'Maven3.8.5'
  }
  stages {
    stage('github clone') {
      steps {
        git branch: 'main', url: 'https://github.com/joneconsulting/cicd-web-project.git';
      }
    }
  }
}
```

- Build Now



# Exercise 18 # Execute a pipeline on slaves 2/5

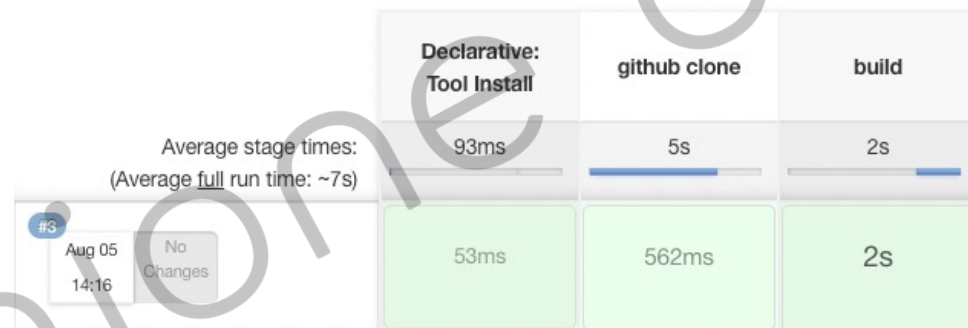
njone company

## ■ Pipeline 수정

### - **Build Stage 추가**

```
...
stage('build') {
  steps {
    sh '''
      echo build start
      mvn clean compile package -DskipTests=true
    '''
  }
}
...
```

## ■ Build Now





## Exercise 18 # Execute a pipeline on slaves 3/5

njone company

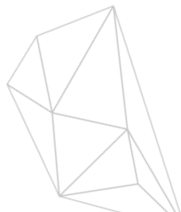
- 새로운 Server 추가 (Slave2)

### Windows, MacOS)

```
$ docker run --privileged --name jenkins-node2 -itd -p 40022:22 \  
-e container=docker -v /sys/fs/cgroup:/sys/fs/cgroup --cgroupns=host \  
edowon0623/docker:latest /usr/sbin/init
```

### Apple silicon, m1)

```
$ docker run --privileged --name jenkins-node2 -itd -p 40022:22 \  
-e container=docker -v /sys/fs/cgroup:/sys/fs/cgroup --cgroupns=host \  
edowon0623/docker-server:m1 /usr/sbin/init
```



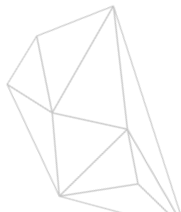
njone company



## Exercise 18 # Execute a pipeline on slaves 4/5

njone company

- Manage Jenkins → Manage Nodes → New Node
  - Node Name: **slave2**
  - Description: **Add a server as the second slave machine**
  - Number of executors: **2**
  - Remote root directory: **/root/slave2**
  - Labels: **slave2**
  - Usage: **Use this node as much as possible**
  - Launch method: **Launch agents via SSH**
    - Host: **[Slave server IP address] es) 172.17.0.4**
    - Port: **22**
    - Credentials: **root/P@ssw0rd**



njone

company







## Exercise 18 # Execute a pipeline on slaves 4/5

njone company

- 정상 접속 시

### Manage nodes and clouds

상태 다시  
읽기

S	이름 ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (aarch64)	In sync	38.53 GB	1024.00 MB	38.53 GB	0ms 
	slave1	Linux (aarch64)	In sync	38.53 GB	1024.00 MB	38.53 GB	30ms 
	slave2	Linux (aarch64)	In sync	38.53 GB	1024.00 MB	38.53 GB	27ms 
수집된 데이터		47 sec	47 sec	47 sec	47 sec	47 sec	47 sec



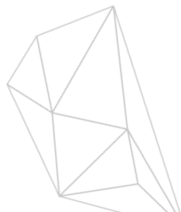
## Exercise 18 # Execute a pipeline on slaves 4/5

njone company

- 접속 오류 발생 시

```
Warning: no key algorithms provided; JENKINS-42959 disabled
SSHLauncher{host='172.17.0.4', port=22, credentialsId='root', jvmOptions='', javaPath='', prefixStartSlaveCmd='',
suffixStartSlaveCmd='', launchTimeoutSeconds=60, maxNumRetries=10, retryWaitTime=15,
sshHostKeyVerificationStrategy=hudson.plugins.sshslaves.verifiers.KnownHostsFileKeyVerificationStrategy,
tcpNoDelay=true, trackCredentials=true}
[08/05/22 05:22:17] [SSH] Opening SSH connection to 172.17.0.4:22.
Searching for 172.17.0.4 in /var/jenkins_home/.ssh/known_hosts
Searching for 172.17.0.4:22 in /var/jenkins_home/.ssh/known_hosts
[08/05/22 05:22:17] [SSH] WARNING: No entry currently exists in the Known Hosts file for this host. Connections
will be denied until this new host and its associated key is added to the Known Hosts file.
Key exchange was not finished, connection is closed.
SSH Connection failed with IOException: "Key exchange was not finished, connection is closed.", retrying in 15
seconds. There are 10 more retries left.
```

...



# Exercise 18 # Execute a pipeline on slaves 5/5

njone company

## ■ Pipeline 수정

### - Agent 수정

```
pipeline {  
  agent {  
    label 'slave2'  
  } ...  
}
```

## ■ Build Now



콘솔 출력

```
Started by user Administrator  
[Pipeline] Start of Pipeline  
[Pipeline] node  
Running on slave2 in /root/slave2/workspace/My-Second-Pipeline  
[Pipeline] {
```

Average stage times:  
(Average full run time: ~40s)

#6  
Aug 05  
14:47  
No  
Changes

Declarative:  
Tool Install

2s

45ms

github clone

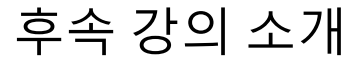
5s

680ms

build

33s

1s



- ~~Spring Cloud로 개발하는 마이크로서비스 애플리케이션~~
- **Jenkins를 이용한 CI/CD Pipeline 구축**
- *Microservice Architecture와 Patterns*
- *Spring Boot와 WebFlux를 이용한 Reactive RESTful API 개발*