

# DEPOLY

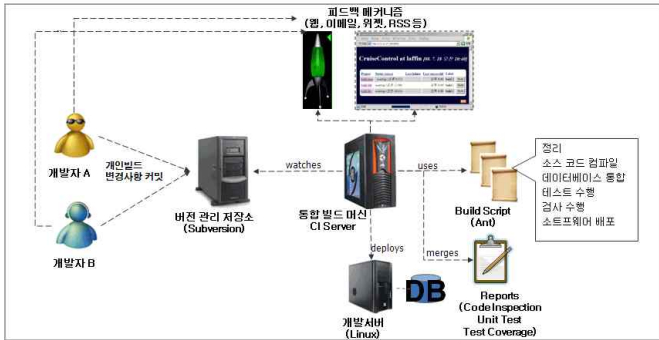
---

## CI/CD

- **Continuous Integration/Continuous Delivery**

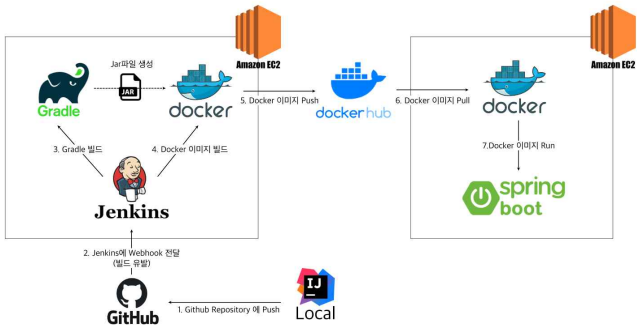
- **CI** : 지속적인 통합이 제대로 구현되면 애플리케이션 코드의 새로운 변경 사항이 정기적으로 빌드 및 테스트를 거쳐 공유 리포지토리에 병합
- **CD** : 지속적인 배포(또 다른 의미의 "CD": Continuous Deployment)란 개발자의 변경 사항을 리포지토리에서 고객이 사용 가능한 프로덕션 환경까지 자동으로 릴리스하는 것을 의미.

# CI/CD - git을 통한 배포



이미지참조 : [https://www.egovframe.go.kr/wiki/doku.php?id=egovframework:dev3.8:dep:build\\_tool:통합빌드](https://www.egovframe.go.kr/wiki/doku.php?id=egovframework:dev3.8:dep:build_tool:통합빌드)

## CI/CD – docker를 통한 배포



## DEPOLY

- 배포에 필요한 필수 애플리케이션(Window 22.10.17 기준)

1) Docker(docker hub)

2) Maven or Gradle

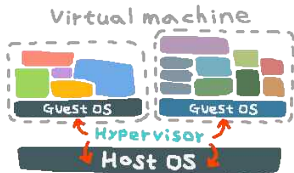
3) Git (git hub)

4) Java 8~11 Ver.

5) Jenkins

# Docker

- 컨테이너 기술을 기반으로 한 가상화 플랫폼
  - 가상화 : 하드웨어 공간 위에 가상의 머신을 만드는 기술
  - 컨테이너 : 컨테이너가 실행되고 있는 호스트 OS의 기능을 그대로 사용하면서 프로세스를 격리해 독립된 환경을 만드는 기술
- 리눅스의 응용 프로그램들을 프로세스 격리 기술들을 사용해 컨테이너로 실행하고 관리





# Docker

- <https://www.docker.com/products/docker-desktop/>

-> docker 다운 및 설치

## Docker Desktop

Install Docker Desktop – the fastest way to containerize applications.

Mac with Intel Chip

Mac with Apple Chip

MOST COMMON

Also available for [Windows](#) and [Linux](#)





# Docker

- Docker 실행 화면 (Window에서 Docker 명령어 사용할 경우 Docker Desktop 실행 되어야함)

The screenshot shows the Docker Desktop application window. The left sidebar contains navigation options: Containers, Images, Volumes, Dev Environments (with a 'beta' badge), Extensions (with a 'beta' badge), and Add Extensions. The main panel is titled 'Images on disk' and shows a summary of 2 images with a total size of 66.8 MB. Below this, the 'Images' section is active, displaying a table of local images. The table has columns for NAME, TAG, IMAGE ID, CREATED, and SIZE. Two images are listed: 'alpine/git' and 'docker101tutorial'. At the bottom of the window, a status bar shows system metrics: RAM 1.32GB, CPU 0.23%, and connection status 'Connected to Hub'. The version 'v4.11.1' is also displayed.

**Images on disk** 2 Images Total size: 66.8 MB IN USE UNUSED [Clean up...](#)

**Images** [Give feedback](#)

LOCAL REMOTE REPOSITORIES

Search

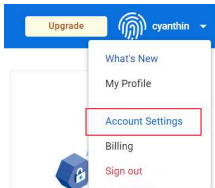
☐ In use only

NAME ↑	TAG	IMAGE ID	CREATED	SIZE
alpine/git	latest	df6434eeda49	about 2 months ago	43.44 MB
docker101tutorial	latest	8fe0cd1ca944	about 1 month ago	28.9 MB

RAM 1.32GB CPU 0.23% Connected to Hub v4.11.1

# docker

- dockerhub 가입하고 접근 토큰 발급
  - <https://hub.docker.com/>
    - -> docker를 사용하기 위해서 위 사이트 가입
- 토큰 발급
  - Account Setting -> Security -> New access Token



# Docker

## • Docker 명령어

<code>docker build -t yedam_quiz .</code>	//이미지 빌드
<code>docker images</code>	//이미지 리스트
<code>docker rmi ____</code>	//이미지 삭제
<code>docker run -d -p 80:80 yedam_quiz</code>	//컨테이너 시작 -d는 백그라운드 실행(접속포트/내부서버포트)
<code>docker stop ____</code>	//컨테이너 중지
<code>docker rm ____</code>	//컨테이너 삭제
<code>docker ps -a</code>	//컨테이너 리스트
<code>docker logs yedam_quiz</code>	//컨테이너 로그보기
<code>docker exec -it 컨테이너Id or 컨테이너이름 /bin/sh</code>	//컨테이너 안으로 들어가기

# Maven

- <https://maven.apache.org/download.cgi>

The screenshot shows the Apache Maven Project website for version 3.8.6. The page is titled "Downloading Apache Maven 3.8.6" and includes a sidebar with navigation links, a "System Requirements" section, and a "Files" section with a table of download links.

**Apache Maven Project**  
<http://maven.apache.org/>

Download | Get Sources | Last Published: 2022-10-08

## Downloading Apache Maven 3.8.6

Apache Maven 3.8.6 is the latest release and recommended version for all users.

### System Requirements

<b>Java Development Kit (JDK)</b>	Maven 3.3+ requires JDK 1.7 or above to execute - they still allow you to build against 1.3 and other JDK versions by using <a href="#">Toolchains</a>
<b>Memory</b>	No minimum requirement
<b>Disk</b>	Approximately 13MB is required for the Maven installation itself. In addition to that, additional disk space will be used for your local Maven repository. The size of your local repository will vary depending on usage but expect at least 500MB.
<b>Operating System</b>	No minimum requirement. Start up scripts are included as shell scripts and Windows batch files.

### Files

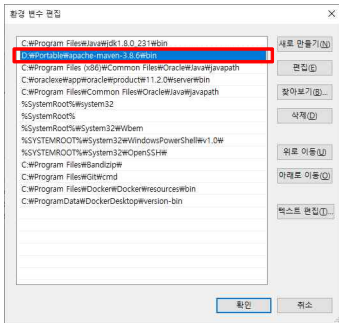
Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the [installation instructions](#). Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted download/distributions, it is highly recommended to [verify the signature](#) of the release bundles against the public [PGP's](#) used by the Apache Maven developers.

	Link	Checksums	Signature
Binary tar.gz archive	<a href="#">apache-maven-3.8.6-bin.tar.gz</a>	<a href="#">apache-maven-3.8.6-bin.tar.gz.sha1</a> <a href="#">apache-maven-3.8.6-bin.tar.gz.sha256</a>	<a href="#">apache-maven-3.8.6-bin.tar.gz.asc</a>
Binary zip archive	<a href="#">apache-maven-3.8.6-bin.zip</a>	<a href="#">apache-maven-3.8.6-bin.zip.sha1</a> <a href="#">apache-maven-3.8.6-bin.zip.sha256</a>	<a href="#">apache-maven-3.8.6-bin.zip.asc</a>
Source tar.gz archive	<a href="#">apache-maven-3.8.6-src.tar.gz</a>	<a href="#">apache-maven-3.8.6-src.tar.gz.sha1</a> <a href="#">apache-maven-3.8.6-src.tar.gz.sha256</a>	<a href="#">apache-maven-3.8.6-src.tar.gz.asc</a>
Source zip archive	<a href="#">apache-maven-3.8.6-src.zip</a>	<a href="#">apache-maven-3.8.6-src.zip.sha1</a> <a href="#">apache-maven-3.8.6-src.zip.sha256</a>	<a href="#">apache-maven-3.8.6-src.zip.asc</a>

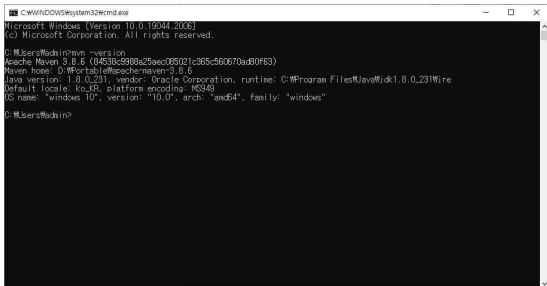
- Release Notes
- Reference Documentation
- Apache Maven Website As Documentation Archive
- All current release sources (plugins, shared libraries, ...) available at <https://downloads.apache.org/maven/>
- Latest source code from source repository
- Distributed under the Apache License, version 2.0

# Maven



# Maven

- mvn -version
- mvn package -DskipTests=true install (소스코드 테스트 하지 않고 배포용 jar생성)



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.19044.2006]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Wadain>mvn -version
Apache Maven 3.8.6 (84538c9988a25aec085021c365c560670ad30f63)
Maven home: D:\Portable\apache-maven-3.8.6
Java version: 1.8.0_231, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk1.8.0_231\jre
Default locale: ko_KR, platform encoding: MS949
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"

C:\Users\Wadain>
```

# Git

- <https://git-scm.com/download/win>

git --local-branching-on-the-cheap

Search entire site...

**Download for Windows**

Click here to download the latest (2.38.0) 64-bit version of Git for Windows. This is the most recent maintained build. It was released 13 days ago, on 2022-10-05.

Other Git for Windows downloads

Windows Installer

32-bit Git for Windows Setup

64-bit Git for Windows Setup

Portable ("OneDrive-compatible")

32-bit Git for Windows Portable

64-bit Git for Windows Portable

Using winget tool

Install winget tool if you don't already have it, then type this command in command prompt or PowerShell:

```
winget install --id Git.Git --s --source winget
```

The current source code release is version 2.38.0. If you want the newer version, you can build it from the source code.

**Now What?**

Now that you have downloaded Git, it's time to start using it.

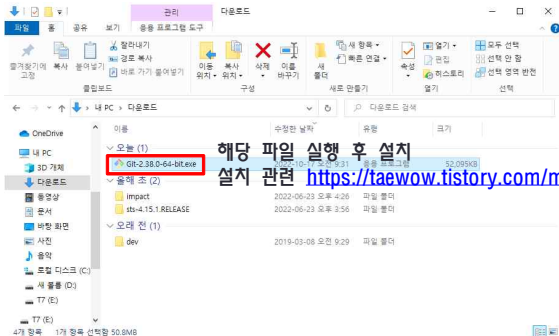
**Read the Book**  
Dive into the free Git book and learn at your own pace.

**Download a GUI**  
Several free and commercial GUI tools are available for the Windows platform.

**Get Involved**  
A large, helpful Git community is available to answer your questions.

컴퓨터 환경에 맞게 설치

# Git



해당 파일 실행 후 설치  
설치 관련

<https://taewow.tistory.com/m/13>

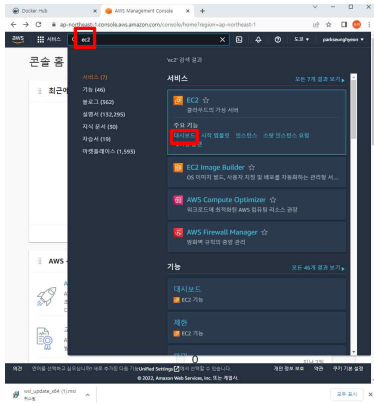
내용 참고



## EC2 - 생성

• Aws 사이트 접속 후 회원가입

-> 서버로 사용할 EC2(Linux환경) 세팅



## EC2 - 생성

아시와 해당영 (도표) 리전에서 다른 Amazon EC2 리소스를 사용하고 있음:

리소스	리소스	리소스
인스턴스(일괄 열기)	로드 밸런서	배치 그룹
보안 그룹	블록	스냅샷
인스턴스	전송 호스트	키 페어
탄력적 GP		

AWS Launch Wizard for SQL Server를 사용하여 AWS에서 Microsoft SQL Server Always On 가용성 그룹을 손쉽게 크기 조정, 수정 및 복원할 수 있습니다. 자세히 알아보기

**인스턴스 시작**  
시작하려면 클라우드의 가장 쉬운 Amazon EC2 인스턴스를 시작하십시오.

**인스턴스 시작** **시작 가이드라인**

참고: 인스턴스는 아시아 태평양 (도표) 리전에서 시작됩니다.

**예약된 미번트**

확실히 해당영 (도표)  
예약된 인스턴트 주문

**서비스 상태**  
AWS Health 대시보드

리전  
중심적 해당영 (도표)  
이 서비스가 정상적으로 작동 중입니다.

**영역**

영역 이름	영역 ID
ap-northeast-1a	apne1-az4
ap-northeast-1c	apne1-az1
ap-northeast-1d	apne1-az2

**추가 정보**  
시작 안내서  
설명서  
모든 EC2 리소스  
요금  
요금  
문서

**도움말 주제**  
Amazon EC2 리소스를 사용하여나 종료하려면 어떻게 해야 하나요?

계산 인스턴스를 관리하고 싶으십니까? 새로 추가된 다음 기능인 **Unified Settings**에서 관리할 수 있습니다.

© 2023, Amazon Web Services, Inc. 또는 계열사. 개인 정보 보호 약관 추가 기본 정보

# EC2 - 생성

## 인스턴스 시작 정보

Amazon EC2를 사용하면 AWS 클라우드에서 실행되는 가상 머신 또는 인스턴스를 생성할 수 있습니다. 아래의 간단한 단계에 따라 빠르게 시작할 수 있습니다.

### 이름 및 태그 정보

이름

dockerDeploy

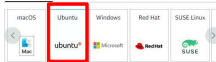
추가 태그 추가

### 애플리케이션 및 OS 이미지(Amazon Machine Image) 정보

AMI는 인스턴스를 시작하는 데 필요한 소프트웨어 구성(운영 체제, 애플리케이션 서버 및 애플리케이션)이 포함된 컴퓨팅 이미지입니다. 아래에서 찾고 있는 항목이 표시된 AMI를 검색하거나 찾아보십시오.

Q 수천 개의 애플리케이션 및 OS 이미지를 포함하는 전체 카탈로그 검색

#### Quick Start



Amazon Machine Image(AMI)

더 많은 AMI 찾아보기  
AWS, Marketplace 및 커뮤니티의 AMI 포함

인스턴스 유형을 선택하십시오

docker

키 페어 생성

### 네트워크 설정 정보

네트워크 정보

sgp-039e0bc75e21107e0

서브넷 정보

기본 설정 없음(가용 영역의 기본 서브넷)

대용량 및 자동 확장 정보

활성화

명확화(보안 그룹) 정보

보안 그룹은 인스턴스에 대한 트래픽을 제어하는 방화벽 규칙 세트입니다. 특정 트래픽이 인스턴스에 도달하도록 허용하는 규칙을 선택하십시오.

☒ 보안 그룹 설정

☐ 기본 보안 그룹 선택

다른 규칙을 사용하여 LaunchWizard-11이라는 새 보안 그룹을 생성합니다.

☒에서 SSH 트래픽 허용

인스턴스 연결에 도움

위치 부근

0.0.0.0/0

☐ 인터넷에서 HTTP 트래픽 허용

대용량 및 웹 서버를 생성할 때 인스턴스를 설정하려면

☐ 인터넷에서 HTTP 트래픽 허용

대용량 및 웹 서버를 생성할 때 인스턴스를 설정하려면

⚠ 소스가 0.0.0.0/0인 규칙은 모든 IP 주소에서 인스턴스에 액세스하도록 허용합니다. 알려진 IP 주소의 액세스만 허용하려면 보안 그룹을 설정하는 것이 좋습니다.

### 키 페어 생성

키 페어를 사용하면 인스턴스에 안전하게 연결할 수 있습니다.

아래에 키 페어의 이름을 입력합니다. 메시지가 표시되면 프라이빗 키를 사용자 컴퓨터에 업로드하고 액세스 가능한 위치를 저장합니다. 나중에 인스턴스에 연결할 때 필요합니다. [전체 설명 보기](#)

키 페어 이름

키 페어 이름 입력

이름은 최대 35개의 ASCII 문자를 포함할 수 있습니다. 선택하는 유일한 문자를 포함할 수 있습니다.

키 페어 유형

☒ RSA

RSA 암호화는 프라이빗 및 공용 키 페어

☐ EC2 SSH

EC2 SSH 암호화된 프라이빗 및 공용 키 페어(Microsoft 인스턴스는 지원하지 않음)

프라이빗 키 파일 업로드

☒ pem

OpenSSH 및 기타 사용

☐ ppk

PUTTY와 기타 사용

완료

키 페어 생성

# EC2 - 생성

다음 규칙을 사용하여 **launch-wizard-1**(미)라는 새 보안 그룹을 생성합니다.

- ☒ **에서 SSH 프록시 허용**  
인스턴스 연결의 도출 위치: 무한
- ☐ **인터넷에서 HTTPS 프록시 허용**  
작을 클라이언트 서버를 생성할 때 인스턴스를 설정하려면
- ☐ **인터넷에서 HTTP 프록시 허용**  
작을 클라이언트 서버를 생성할 때 인스턴스를 설정하려면

**소스가 0.0.0.0/0인 규칙은 모든 IP 주소에서 인스턴스에 액세스하도록 허용합니다. 알려진 IP 주소의 액세스만 허용하도록 보안 그룹을 설정하는 것이 좋습니다.**

**스토리지 구성** 어드밴스드

1x  GiB  루트 볼륨

**프리 티어를 사용할 수 있는 고객은 최대 30GB의 EBS 볼륨(SSD) 또는 마그네틱 스토리지를 사용할 수 있습니다.**

**새 볼륨 추가**

선택한 AMI에 인스턴스가 허용하는 것보다 많은 인스턴스 스토어 볼륨이 포함되어 있습니다. AMI에서 자용 0개의 인스턴스 스토어 볼륨에만 액세스할 수 있습니다.

0 x 파일 시스템 명칭

**요약**

인스턴스 개수:

소프트웨어 이미지(AMI)  
Canonical, Ubuntu, 22.04 LTS, ... 더 보기  
ami-01f4b076d2981b45

가성 서버 유형(인스턴스 유형)  
t2.micro

발행액(보안 그룹)  
새 보안 그룹

스토리지(볼륨)  
1개의 볼륨 - 30GiB

**프리 티어: 프리 티어는 월별 프리 티어 AMI에 대한 t2.micro 또는 t2.micro를 사용할 수 있는 고객과 t2.micro 인스턴스 사용량 750시간, EBS 스토리지 300GB, IO 2백만 개, 수렴 및 1GB, 인스턴스 대역폭 300GB가 포함됩니다.**

취소 **인스턴스 제작**

# EC2 - 보안

Amazon EC2 Management Console - 리소스 (Resources)

아시안 태평양 (서울) 리전에서 다음 Amazon EC2 리소스를 사용하고 있음:

리소스	수량	타입	상태	타입	수량
인스턴스(실용형)	1	호스트 열린서	0	백지 그룹	0
보안 그룹	6	폴딩	1	스냅샷	0
인스턴스	1	전용 호스트	0	키 페어	1
탄력적 BP	0				

[AWS Launch Wizard for SQL Server를 사용하여 AWS에서 Microsoft SQL Server Always On 가동형 그룹을 손쉽게 크기 조정, 구성 및 배포할 수 있습니다. 자세히 알아보기](#)

**인스턴스 시작**  
시작하려는 플랫폼과 가장 적합한 Amazon EC2 인스턴스를 시작하십시오.

[인스턴스 시작](#) [서버 마이그레이션](#)

참고: 인스턴스는 아시아 태평양 (서울) 리전에서 시작됩니다.

**예약된 인스턴트**

하위계 태평양 (서울)  
예약된 인스턴트 그룹

**서버 마이그레이션**

**서비스 상태**  
리전: 아시아 태평양 (서울)  
상태: 이 서비스가 정상적으로 작동 중입니다.

**영역**

영역 이름	영역 ID
ap-northeast-2a	apne2-az1
ap-northeast-2b	apne2-az2
ap-northeast-2c	apne2-az3
ap-northeast-2d	apne2-az4

[추가 영역 살펴보기](#)

**계정 구성**

지정되는 플랫폼

- VPC

기본 VPC  
vpc-02fa379d217d8ffa6

설정

EBS 암호화

영역

EC2 직접 콘솔

기본 프로파일 사용

콘솔 설정

**추가 정보**

시작 안내서

설명서

모든 EC2 리소스

표준

요금

콘솔

**도움말 주제**

Amazon EC2 리소스를 시작하거나 종료하려면 어떻게 해야 하나요?

SSH를 사용하는 사용자 계정용 EC2 Linux 인스턴스에 추가

SSH를 사용하여 Linux 인스턴스에 연결

© 2022 Amazon Web Services, Inc. 또는 제공사. 개인 정보 보호 | 약관 | 추가 정보 | 설정

# EC2 - 보안

The screenshot displays the AWS Management Console for EC2 instances. The left sidebar shows navigation options like 'New EC2 Experience', 'EC2 대시보드', and '인스턴스'. The main area shows a list of instances. One instance, named 'docker', is highlighted with a red box around its ID 'i-05a9003be6066795f'. The instance is in a 'Running' state, as indicated by the green checkmark and the text '인스턴스 상태: running' in the search bar.

Name	인스턴스 ID	인스턴스 상태	인스턴스 유형	상태 검사	필요 상태	기본 영역	지정된 IPv4 DNS	지정된 IPv4...	당첨된 IP
docker	i-05a9003be6066795f	Running	t2.micro	2/2개 통과...	필요 없음	ap-northeast-2a	ec2-3-96-65-207.ap-no...	3.96.65.207	-

인스턴스 선택

© 2022, Amazon Web Services, Inc. 또는 계열사. 개인 정보 보호 약관

# EC2 - 보안

인스턴스 세부 정보 | EC2 Main

ap-northeast-2-console.amazonaws.com/ec2/home?region=ap-northeast-2&instancesDetails&instanceId=i-05a3003be608d795f

새 서비스 | 서비스 | 검색 | 리소스 | 로그 | 알림 | 설정 | [Alt+H]

New EC2 Experience

EC2 > 인스턴스 > i-05a3003be608d795f

### i-05a3003be608d795f (docker)에 대한 인스턴스 요약

less than a minute 전에 업데이트됨

<b>인스턴스 ID</b> i-05a3003be608d795f (docker)	<b>공유된 IPv4 주소</b> 5.96.85.207   <a href="#">재할당 주소</a>	<b>프라이빗 IPv4 주소</b> 172.31.8.32
<b>IPv6 주소</b> -	<b>인스턴스 상태</b> 실행 중	<b>마블릭 IPv4 DNS</b> ec2-5-96-85-207.ap-northeast-2.compute.amazonaws.com   <a href="#">재할당 주소</a>
<b>호스트 이름 유형</b> IP 이름: ip-172-31-8-32.ap-northeast-2.compute.internal	<b>프라이빗 IP DNS 이름(IPv4만 해당)</b> ip-172-31-8-32.ap-northeast-2.compute.internal	<b>탄력적 IP 주소</b> -
<b>관리되는 리소스 DNS 이름</b> IPv4 이름	<b>인스턴스 유형</b> t2.micro	<b>AWS Compute Optimizer 찾기</b> ① 권장 사항을 위해 AWS Compute Optimizer에 통합되었습니다.   <a href="#">자세히 알아보기</a>
<b>자동 할당된 IP 주소</b> 3.36.85.207 (마블릭 IP)	<b>VPC ID</b> vpc-02ba379221748ff1a6	<b>Auto Scaling 그룹 이름</b> -
<b>IAM 역할</b> -	<b>서브넷 ID</b> subnet-0a5f9436a75e68cd	

세부 정보 | **보안** | 네트워크 | 스토리지 | 상태 관리 | 모니터링 | 태그

▼ 보안 세부 정보

IAM 역할  
-

소유자 ID  
287262597802

시작 시간  
Wed Oct 12 2022 10:51:40 GMT+0900 (한국 표준시)

**보안 그룹**  
[sg-031b0625d914ab995 \(launch-sg-4\)](#)

▼ 인스턴드 규칙

규칙 이름: sg-031b0625d914ab995

보안 그룹 규칙 ID	포트 범위	프로토콜	원본	보안 그룹
sg-031b0625d914ab995	85	TCP	0.0.0.0/0	launch-sg-4

© 2022, Amazon Web Services, Inc. 또는 계열사. 개인 정보 보호 | 약관 | 추가 기술 설명

# EC2 - 보안

The screenshot shows the AWS Management Console for the 'launch-wizard-4' security group. The left sidebar contains navigation links for EC2, IAM, and other services. The main content area displays the 'sg-0f6385cdfa74dc9e0 - launch-wizard-4' security group details.

**세부 정보**

보안 그룹 이름 launch-wizard-4	보안 그룹 ID sg-0f6385cdfa74dc9e0	생성 launch-wizard-4 created 2022-10-12T01:31:06:000Z	VPC ID vpc-02ba3792217d8ffa6
소유자 287202597802	인바운드 규칙 수 2 제한 항목	아웃바운드 규칙 수 1 제한 항목	

**인바운드 규칙** | 아웃바운드 규칙 | 태그

이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다. [Reachability Analyzer 실행](#)

**인바운드 규칙 (2)**

모든 그룹 규칙 보기

<input type="checkbox"/>	Name	보안 그룹 규칙 ID	IP 버전	행위	프로토콜	포트 범위	소스	설명
<input type="checkbox"/>	-	sgn-031b9623d8f4ab595	IPv4	사용자 지정 TCP	TCP	85	0.0.0.0/0	-
<input type="checkbox"/>	-	sgn-023838a43eff77e71	IPv4	SSH	TCP	22	0.0.0.0/0	-

© 2022, Amazon Web Services, Inc. 또는 계열사. 개인 정보 보호 | 약관 | 추가 기능 설명



# EC2 - 보안

EC2 > 보안 그룹 > sg-06385cd87463e0 - launch-wizard-4 > 인바운드 규칙 편집

### 인바운드 규칙 편집

인바운드 규칙은 인스턴스에 도달하도록 허용된 수신 트래픽을 제어합니다.

#### 인바운드 규칙

보안 그룹 규칙 ID	유형	프로토콜	포트 범위	소스	설명 - 선택 사항
sg-06385cd87463e0	사용자 지정 TCP	TCP	85	사용자 지정	
sg-025936a40e773e71	SSH	TCP	22	사용자 지정	

규칙 추가

취소 변경 사항 무시하기 규칙 저장

© 2022, Amazon Web Services, Inc. 또는 계열사. 개인 정보 보호 약관 우리 기업 소개

SSH(Secure Shell) : 서버에 원격 접속할 때 외부의 보안 위협으로부터 보호.  
 인바운드 : 해당 서버에 접속할 수 있는 IP, Port 규칙 설정

# EC2 - 접속

- <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

The screenshot shows a web browser window displaying the PuTTY download page. The page title is "Download PuTTY: latest release (0.77)". Below the title, there are links for Home, FAQ, Feedback, License, Updates, Mirrors, Keys, Links, and Team. There are also links for Download: Stable, Pre-release, Source, Docs, Changelog, and Wishlist.

The page contains the following text:

This page contains download links for the latest released version of PuTTY. Currently this is 0.77, released on 2022-05-27.

When new releases come out, this page will update to contain the latest, so this is a good page to bookmark or link to. Alternatively, here is a [permanent link to the 0.77 release](#).

Release versions of PuTTY are versions we think are reasonably likely to work well. However, they are often not the most up-to-date version of the code available. If you have a problem with this release, then it might be worth trying out the [pre-release builds of 0.78](#) or the [development snapshots](#), to see if the problem has already been fixed in those versions.

**Package files**

You probably want one of these. They include versions of all the PuTTY utilities (except the new and slightly experimental Windows pterm).

(Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

We also publish the latest PuTTY installers for all Windows architectures as a free-of-charge download at the [Microsoft Store](#); they usually take a few days to appear there after we release them.

<b>Linux (Windows container)</b>		
64-bit x86:	<a href="#">putty-0.77-1-3.77-installer.exe</a>	<a href="#">(signature)</a>
32-bit x86:	<a href="#">putty-0.77-1-3.77-installer.exe</a>	<a href="#">(signature)</a>
<b>Unix source archive</b>		
.tar.gz:	<a href="#">putty-0.77.tar.gz</a>	<a href="#">(signature)</a>

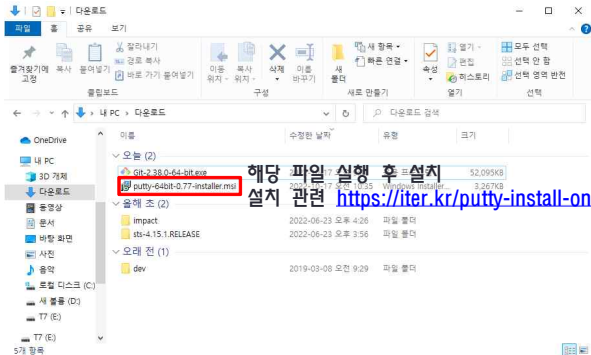
**Alternative binary files**

The installer packages above will provide versions of all of these (except PuTTYtel and pterm), but you can download standalone binaries one by one if you prefer.

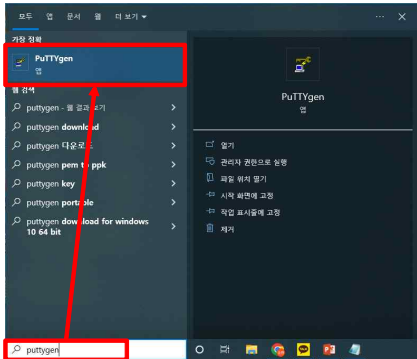
(Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

<b>putty.exe (the SSH and Telnet client itself)</b>		
64-bit x86:	<a href="#">putty.exe</a>	<a href="#">(signature)</a>
64-bit ARM:	<a href="#">putty.exe</a>	<a href="#">(signature)</a>
32-bit x86:	<a href="#">putty.exe</a>	<a href="#">(signature)</a>

## EC2 - 접속



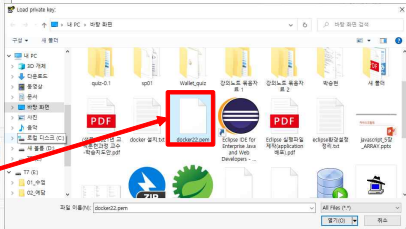
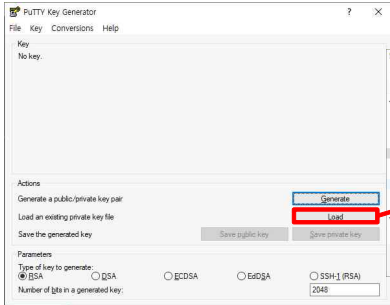
## EC2 - 접속



1) Win키 누른 후 puttygen 입력

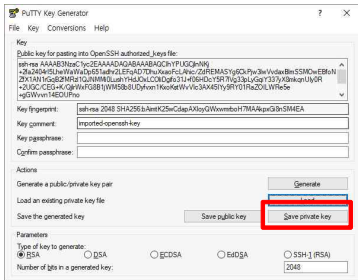
2) 검색 결과 후 애플리케이션 실행

# EC2 - 접속



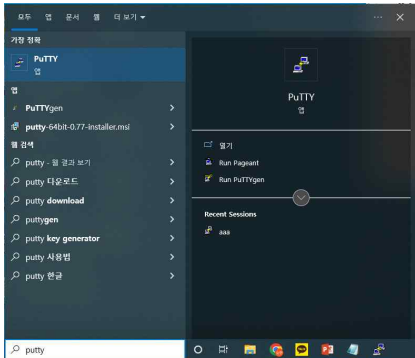
EC2 생성시 만든 Key pair 파일(.pem) 선택

## EC2 - 접속



Putty 접속할 때 사용할 PPK 파일 생성 경로  
지정 후 완료

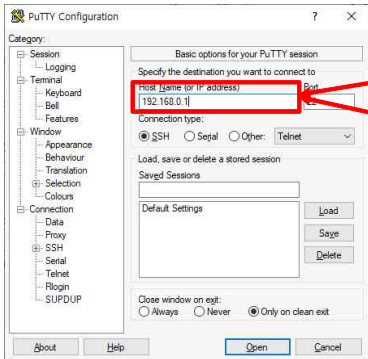
## EC2 - 접속



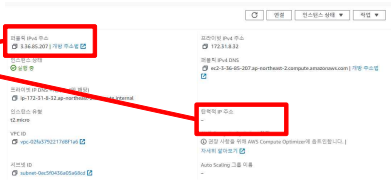
1) Win키 누른 후 putty 입력

2) 검색 결과 후 애플리케이션 실행

## EC2 - 접속

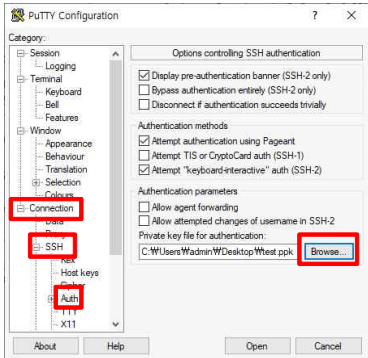


### 1) EC2 탄력적 IP 또는 퍼블릭 IP 입력

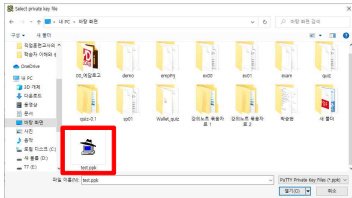




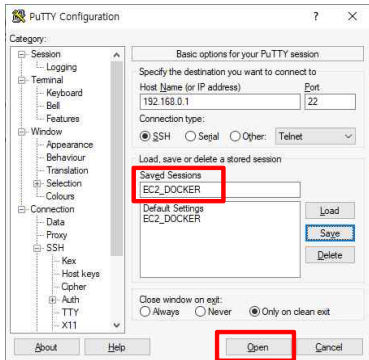
## EC2 - 접속



2) Coonection > SSH > Auth  
-> 제작한 PPK 파일 삽입



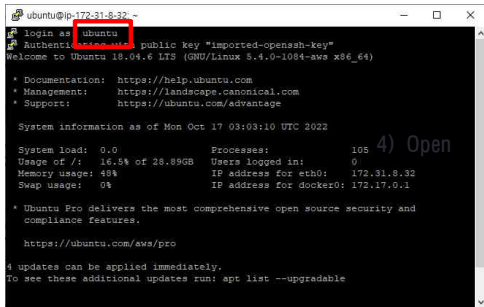
## EC2 - 접속



- 3) Saved Sessions에서 설정한 내용 저장  
 -> Save : 설정 기준 IP, PPK 데이터 저장  
 -> Load : 저장 기준 IP, PPK 데이터 불러오기  
 -> Delete : Save 데이터 삭제

### 4) Open

## EC2 - 접속

A terminal window titled 'ubuntu@ip-172-31-8-32: ~' with standard window controls. The terminal shows the login process for the 'ubuntu' user. The prompt 'login as:' is followed by 'ubuntu', which is highlighted with a red rectangle. Below this, it says 'Authentication: public key "imported-openssh-key"', 'Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1084-aws x86\_64)', and a list of links for documentation, management, and support. Then, it displays system information as of Mon Oct 17 03:03:10 UTC 2022, including system load, usage of /, memory usage, swap usage, processes, users logged in, and IP addresses for eth0 and docker0. It also mentions Ubuntu Pro and provides a link to the Ubuntu Pro page. Finally, it states that 4 updates can be applied immediately and provides a command to see these updates: 'apt list --upgradable'.

```
ubuntu@ip-172-31-8-32: ~  
login as: ubuntu  
Authentication: public key "imported-openssh-key"  
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1084-aws x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Mon Oct 17 03:03:10 UTC 2022  
  
System load:  0.0          Processes:           105  
Usage of /:   16.5% of 28.89GB  Users logged in:    0  
Memory usage: 48%          IP address for eth0: 172.31.8.32  
Swap usage:   0%           IP address for docker0: 172.17.0.1  
  
* Ubuntu Pro delivers the most comprehensive open source security and  
  compliance features.  
  
https://ubuntu.com/aws/pro  
  
4 updates can be applied immediately.  
To see these additional updates run: apt list --upgradable
```

5) login as :

-> ubuntu 입력

: ubuntu 계정으로 로그인

6) 접속 완료

4) Open

## EC2 - Docker install in linux

- 참조사이트 : <https://docs.docker.com/engine/install/ubuntu/>

- EC2 Docker 설치 가능 Ubuntu version

- 1) Ubuntu Jammy 22.04 (LTS)

- 2) Ubuntu Impish 21.10

- 3) Ubuntu Focal 20.04 (LTS)

- 4) Ubuntu Bionic 18.04 (LTS) 아래 예제에서 사용할 버전

## EC2 - Docker install in linux

- Uninstall docker(기존에 설치가 되어있을 때 사용)

```
$ sudo apt-get remove docker docker-engine docker.io containerd runc
```

- 설치가능한 리스트 업데이트

```
$ sudo apt-get update
```

- 설치에 필요한 패키지 설치

```
$ sudo apt-get install ca-certificates
```

```
$ sudo apt-get install curl
```

```
$ sudo apt-get install gnupg
```

```
$ sudo apt-get install lsb-release
```

## EC2 - Docker install in linux

- Docker gpg 암호화

```
$ sudo mkdir -p /etc/apt/keyrings
```

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --  
dearmor -o /etc/apt/keyrings/docker.gpg
```

- gpg 암호화 참고 사이트

[https://librewiki.net/wiki/시리즈:암호의\\_암도\\_몰라도\\_쉽게\\_하는\\_GPG](https://librewiki.net/wiki/시리즈:암호의_암도_몰라도_쉽게_하는_GPG)

## EC2 - Docker install in linux

- .

```
$ echo "deb [arch=$(dpkg --print-architecture) signed-  
by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu  
\ $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list >  
/dev/null
```

## EC2 - Docker install in linux

- **Install docker engine**

```
$ sudo apt-get update
```

```
$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```



## EC2 - Docker install in linux

- **[옵션]특정 버전 설치**
  - **사용가능한 repo 확인 후 설치**

```
$ apt-cache madison docker-ce
```

```
$ sudo apt-get install docker-ce=<VERSION_STRING> docker-ce-  
cli=<VERSION_STRING> containerd.io docker-compose-plugin
```

-> <VERSION\_STRING>은 다음 페이지부터 설명




## EC2 - Docker install in linux

- **사용가능한 repo 확인 후 설치**

```
$ apt-cache madison docker-ce
```

```
$ sudo apt-get install docker-ce=<VERSION_STRING> docker-ce-  
cli=<VERSION_STRING> containerd.io docker-compose-plugin
```



A diagram with two red boxes. The first box contains the text "<VERSION\_STRING>" from the command above. A red arrow points from this box to a second red box below it, which contains the text "5:20.10.16~3-0~ubuntu-jammy".

```
$ sudo apt-get install docker-ce=5:20.10.16~3-0~ubuntu-jammy docker-ce-  
cli=5:20.10.16~3-0~ubuntu-jammy containerd.io docker-compose-plugin
```

- **Service 시작**

```
$ sudo service docker start
```

```
$ systemctl status docker(서비스 정상 작동 확인)
```

# JENKINS

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

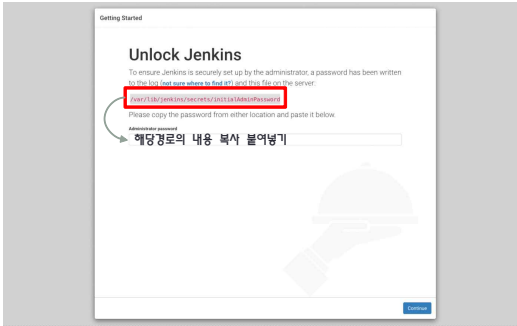
-> window 다운 후 설치



# JENKINS

- localhost:800 접속
- ID, PW 설정
- 참고 사이트

<https://lock.tistory.com/m/3>



# JENKINS

Dashboard

새로운 item

시작

빌드 기록

jenkins 관리

My Views

새로운 뷰

빌드 대기 목록

빌드 실행 상태

1. 대기 중

2. 대기 중

## 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 →

# JENKINS

The screenshot shows the Jenkins web interface in a browser window. The address bar indicates the URL is `localhost:8080/manage`. The top navigation bar includes the Jenkins logo, a search bar, and the user `admin` with a logout button.

**Dashboard**

**Jenkins 관리**

Jenkins 인스턴스 2.345.3을 설치할 수 있습니다. (변경사항)

현재 설치된 다른 구성 요소에 대한 경고가 게시되었습니다.  
 Jenkins 2.345.3 core and libraries  
[HTTP/2 denial of service vulnerability in Javalin.Jetty](#)

**System Configuration**

- 시스템 설정: 환경변수 및 경로 정보를 설정합니다.
- Global Tool Configuration: Configure tools, their locations and automatic installers.
- 플러그인 관리**: Jenkins의 기능을 확장하기 위한 플러그인을 추가, 제거, 사용, 역사적으로 설정할 수 있습니다. (This item is highlighted with a red box in the original image.)
- 노드 관리: Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

**Security**

- Configure Global Security: Secure Jenkins; define who is allowed to access/use the system.
- Manage Credentials: Configure credentials
- Configure Credential Providers: Configure the credential providers and types
- Manage Users: Create/delete/modify users that can log in to this Jenkins.

**Status Information**

- 시스템 정보: 최신 버전을 동기화하여 다양한 환경 정보를 보
- System Log: Runtime log contains redirect from
- 부하 통계: Check your resource utilization and see if
- About Jenkins: See the version and license information.

## JENKINS

The screenshot shows the Jenkins Plugin Manager interface. The top navigation bar includes the Jenkins logo, a search bar, and user information (admin). The main content area is titled 'Plugin Manager' and features a search bar with the text 'ssh'. Below the search bar, there is a table of plugins. The 'Publish Over SSH' plugin is highlighted with a red box. The table has columns for 'Install', 'Name', and 'Released'.

Install	Name	Released
<input type="checkbox"/>	SSH 2.6.1 <b>Build Wrappers</b> This plugin executes shell commands remotely using SSH protocol.  Warning: This plugin version may not be safe to use. Please review the following security notices: <ul style="list-style-type: none"> <li>CSRF vulnerability and missing permission checks allow capturing credentials</li> <li>Missing permission check allows enumerating credentials IDs</li> </ul>	4 yr 6 mo ago
<input checked="" type="checkbox"/>	Publish Over SSH 1.24 <b>Artifact Uploaders</b> <b>Build Tools</b> Send build artifacts over SSH	7 mo 19 days ago
<input type="checkbox"/>	SSH Agent 2.95.0 (Official Jenkins) This plugin allows you to provide SSH credentials to build via a sub-agent in Jenkins.	5 mo 2 days ago
<input type="checkbox"/>	SSH Pipeline Steps 2.0.0 (Official Jenkins) <b>pipeline</b> Jenkins pipeline steps which provides SSH facilities such as command execution or file transfer for continuous delivery.	5 mo 6 days ago

At the bottom of the page, there are buttons for 'Install without restart', 'Download now and install after restart', and 'Update information obtained: 8 min 7 sec ago'. There is also a '재시작' (Restart) button.

설치 완료 후 재시작

안될 시 url -&gt; localhost:800/restart



# JENKINS

## • EC2 <-> Jenkins SSH 통신 설정

The screenshot shows the Jenkins web interface. In the left sidebar, the 'Jenkins 관리' (Jenkins Management) option is highlighted with a red box. The main content area displays the 'Jenkins 관리' (Jenkins Management) page, which includes a yellow warning banner about security issues with built-in mode. Below the banner, the 'System Configuration' section is visible, with the '시스템 설정' (System Configuration) option highlighted by a red box. Other options in the 'System Configuration' section include 'Global Tool Configuration', '플러그인 관리' (Plugin Management), and '노드 관리' (Node Management). The 'Security' section includes 'Configure Global Security', 'Manage Credentials', 'Configure Credential Providers', and 'Manage Users'. The 'Status Information' section includes '시스템 정보' (System Information), 'System Log', '부하 통계' (Load Statistics), and 'About Jenkins'. The 'Troubleshooting' section is also visible at the bottom.

# JENKINS

## • EC2 <-> Jenkins SSH 통신 설정

Install Docker Engine on Ubuntu | Configure System (jenkins)

localhost:8080/configure

Dashboard > Configure System >

Key

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEA...
-----END RSA PRIVATE KEY-----
```

EC2 에서 다운 받은 Key Pair(PEM 파일)  
안에 내용 모두 복사해서 붙여넣기

☐ Disable exec

SSH Servers

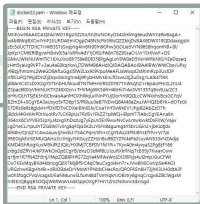
SSH Server

Name

dev-jenkins

저장

Apply



# JENKINS

## • EC2 <-> Jenkins SSH 통신 설정

-----END RSA PRIVATE KEY-----

☐ Disable exec

SSH Servers

SSH Server

Name: SSH 통신 설정 이름 지정

dev-server

Hostname: EC2의 퍼블릭 IP 또는 탄력적 IP 입력

535.85.207

Username: EC2 Linux 계정 입력

ubuntu

Remote Directory:

/home/ubuntu

다음...

Success

Test Configuration

Success 되면 SSH 통신 가능  
확인 후 저장

추가

다음...

저장 Apply

# JENKINS

## • Jenkins 사용 도구 설정

The screenshot shows the Jenkins web interface. The browser address bar displays 'localhost:8080/manage/'. The Jenkins logo and 'admin' user name are visible in the top navigation bar. The left sidebar contains a menu with 'Jenkins 관리' highlighted by a red rectangle. The main content area is titled 'Jenkins 관리' and includes a yellow warning banner about security in built-in mode. Below this, the 'System Configuration' section is visible, with 'Global Tool Configuration' highlighted by a red rectangle. Other options in this section include 'Plugin Manager', 'Node Manager', 'Configure Global Security', 'Manage Credentials', 'Configure Credential Providers', and 'Manage Users'. The 'Status Information' section at the bottom includes 'System Log', 'Resource Usage', and 'About Jenkins'.

**Jenkins 관리**

Building on the built-in mode can be a security issue. You should set up distributed builds. See the documentation. [Set up agent](#) [Set up cloud](#) [Disable](#)

**System Configuration**

- Global Tool Configuration**  
Configure tools, their locations and automatic installers.
- 플러그인 관리자**  
Jenkins의 기능을 확장하기 위한 플러그인을 추가, 제거, 사용, 막아용으로 설정할 수 있습니다.
- 노드 관리자**  
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

**Security**

- Configure Global Security**  
Secure Jenkins: define who is allowed to access/use the system.
- Manage Credentials**  
Configure credentials.
- Configure Credential Providers**  
Configure the credential providers and types.
- Manage Users**  
Create/delete/modify users that can log in to this Jenkins.

**Status Information**

- 시스템 정보**  
문제 해결을 돕기 위한 다양한 환경 정보를 보여줍니다.
- System Log**  
System log captures output from Java, shell, logging output related to Jenkins.
- 부하 통계**  
Check your resource utilization and see if you need more computers for your builds.
- About Jenkins**  
See the version and license information.

**Troubleshooting**

# JENKINS

## • Jenkins 사용 도구 설정 - Java

The screenshot shows the Jenkins web interface at the 'Global Tool Configuration' page. The browser tabs include 'Install Docker Engine on Ubuntu', 'Global Tool Configuration (jenkins)', and 'localhost:8080/configure?tool=java'. The page title is 'Dashboard > Global Tool Configuration'. Under the 'Default global settings provider' section, the dropdown menu is set to 'Use default maven global settings'. The 'JDK' section is expanded, showing 'JDK installations' and 'Use of JDK installations on this system'. A red rectangular box highlights the 'Add JDK' form. Inside this box, the 'Name' field is labeled 'Java 설정 파일 이름 지정' and contains the text 'java11'. The 'JDK\_HOME' field is labeled 'local Java 설치 경로 지정(개발 당시 사용했던 JDK 입력)' and contains the path 'C:\Program Files\Java\jdk-11.0.13'. The 'Install automatically' checkbox is unchecked. Below the red box, there is another 'Add JDK' button. The 'Git' section is partially visible at the bottom, showing 'Git installations' and a 'Add Git' button. At the very bottom, there are 'Save' and 'Apply' buttons.

Install Docker Engine on Ubuntu Global Tool Configuration (jenkins) +

localhost:8080/configure?tool=java

Dashboard > Global Tool Configuration

Default global settings provider

Use default maven global settings

JDK

JDK installations

Use of JDK installations on this system

Add JDK

JDK Name Java 설정 파일 이름 지정

java11

JDK\_HOME local Java 설치 경로 지정(개발 당시 사용했던 JDK 입력)

C:\Program Files\Java\jdk-11.0.13

☐ Install automatically

Add JDK

Git

Git installations

Add Git

Save Apply

# JENKINS

## • Jenkins 사용 도구 설정 - Git

The image shows the Jenkins 'Global Tool Configuration' page for Git. The 'Name' field is set to 'git'. The 'Path to Git executable' field is set to 'C:\Program Files\Git\cmd\git.exe'. A red box highlights the 'Name' and 'Path to Git executable' fields, with a red arrow pointing from the text 'local Git 실행 파일 경로 지정' to the path field. Another red box highlights the 'Name' field, with a red arrow pointing from the text 'git 설정 파일 이름 지정' to the 'Name' field. A Windows command prompt window is overlaid on the right, showing the command 'C:\Users\Wadein\where git' and the output 'C:\Program Files\Git\cmd\git.exe'. The title of the command prompt window is 'local Git 실행 파일 경로 찾는 방법'.

local Git 실행 파일 경로 찾는 방법

```
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Wadein>where git
C:\Program Files\Git\cmd\git.exe
```

local Git 실행 파일 경로 지정

git 설정 파일 이름 지정

git

Path to Git executable

C:\Program Files\Git\cmd\git.exe

Install automatically

Save Apply



# JENKINS

## Jenkins item 생성

The screenshot shows the Jenkins web interface in a browser window. The address bar indicates the URL is `localhost:8080`. The Jenkins logo and name are at the top left of the dashboard. A search bar and user profile (admin) are at the top right. On the left sidebar, the 'New Item' button (a plus sign in a circle) is highlighted with a red rectangle. Below it are links for '시작' (Start), '빌드 기록' (Build History), 'Jenkins 관리' (Jenkins Management), 'My Views', and '새로운 뷰' (New View). The main area shows a table of recent builds for a job named 'docker'. The table has columns for status (S), workspace (W), name, build number, duration, and a play button icon. The first row shows a successful build (#54) that took 55 seconds. At the bottom, there are links for '아이콘' (Icons), 'S', 'H', 'L', and several 'Atom feed' links.

S	W	Name	최근 성공	최근 실패	최근 소모 시간
✓	cloud	docker	1 day 16 hr #54	1 day 16 hr #53	55 sec



# JENKINS

## • Jenkins item 생성

Install Docker Engine on Ubuntu x New Item (Jenkins) x +

← → ↻ localhost:8080/view/all/newJob

**Jenkins** 🔍 검색 🔔 🔔 🔔 🔔 🔔

Dashboard > All >

Enter an item name

Item 이름 지정

Freestyle project  
이것은 Jenkins의 주요 기능입니다. Jenkins은 어느 빌드 시스템과 어떤 SCM(항상관리)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 workflow) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project  
다양한 환경에서의 테스트, 물려를 특정 빌드, 기타 응용 처럼 다수의 서로다른 환경설정이 필요한 프로젝트에 적절함.

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.

Multibranch Pipeline  
Creates a set of Pipeline projects according to detected branches in one SCM repository.

Organization Folder  
Creates a set of multibranch project subfolders by scanning for repositories.

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

Copy from

OK

# JENKINS

- Jenkins item 생성 - Git

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

Git

Repositories ?

Repository URL ? Git pull repository URL 입력

https://github.com/seunghyeon3/yedamQuiz.git

Credentials ?

- none -

+ Add

고급...

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ? Git pull repository branch 입력

main

Add Branch

Repository browser ?

(자동)

저장 Apply

# JENKINS

- Jenkins item  
생성 - Maven

General 소스 코드 관리 빌드 유형 빌드 환경 **Build** 빌드 후 조치

Build

Invoke top-level Maven targets ?

Maven Version  
mvn

Goals  
install -DskipTests

POM ?  
quiz/pom.xml

Properties ?

JVM Options ?

☐ Inject build variables ?

저장 Apply

Invoke top-level Maven targets 선택

Jenkins에서 설정한 mvn 선택

mvn 제외한 명령어 입력

pom.xml 경로 지정

# JENKINS

- Jenkins item  
생성 - docker

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

Use default maven settings

Global Settings file

Use default maven global settings

**Execute Windows batch command 선택**

Command

See the list of available environment variables

**Docker 이미지 생성 및 전자지갑 복사**

```
cd quize
mkdir wallet_quize
copy /-y C:\Users\hadein\git\vedaa\quize\wallet_quize C:\ProgramData\Jenkins\jenkins\workspace\docker\quize\wallet_quize
docker build -t parkseunghyeon/vedaa_quize .
docker login -u parkseunghyeon -p dckr_pat_8167sr9wC85135J-S1JwzQyM4U
docker push parkseunghyeon/vedaa_quize
```

\* 명령어 다음 페이지 설명

고급...

Add build step

**빌드 후 조치**

Send build artifacts over SSH

SSH Publishers

SSH Server

저장 Apply

# JENKINS

- Jenkins item 생성 - docker

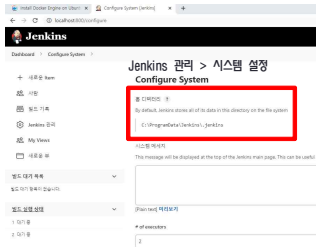
# JENKINS

- Jenkins item 생성 - docker
- cd quiz
- mkdir wallet\_quiz
- copy /-y
- 실제 전자지갑 경로

C:\Users\admin\git\yedamQuiz\quiz\Wallet\_quiz\\*

- 복사될 전자지갑 경로(Jenkins 파일 내 존재)

C:\ProgramData\Jenkins\jenkins\workspace\docker\quiz\wallet\_quiz

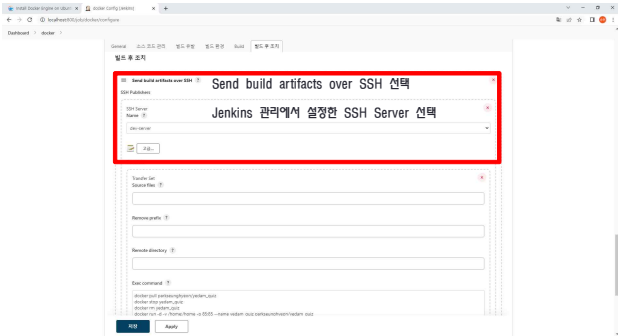


## JENKINS

- Jenkins item 생성 – docker
- `docker build -t parkseunghyeon/yedam_quiz .` (ID/이미지명으로 해야 push 됨)
- `docker login -u parkseunghyeon -p dckr_pat_wlp7srQw0M5135J-SIJxdzQyN4U`  
-> git login처럼 -u 뒤에는 ID, -p 뒤에는 dockerhub token값 필요  
-> 참고 사이트 : <https://teichae.tistory.com/m/entry/Docker-Hub%EC%97%90%EC%84%9C%EC%9D%98-Token-%EB%B0%9C%EA%B8%89-%EB%B0%A9%EB%B2%95>
- `docker push parkseunghyeon/yedam_quiz`

# JENKINS

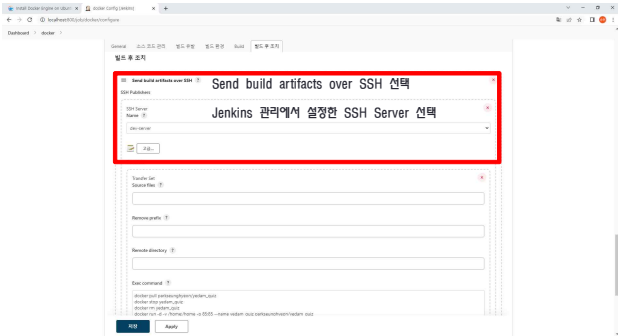
## • Jenkins item 생성 - SSH and Docker 배포





# JENKINS

## • Jenkins item 생성 - SSH and Docker 배포



## • Jenkins item 생성

## - SSH and Docker 배포

```
docker pull parkseunghyeon/yedam_quiz  
docker stop yedam_quiz  
docker rm yedam_quiz  
docker run -d -p 8080:8080 --name yedam_quiz parkseunghyeon/yedam_quiz
```

DockerHub -&gt; EC2 배포 및 저장

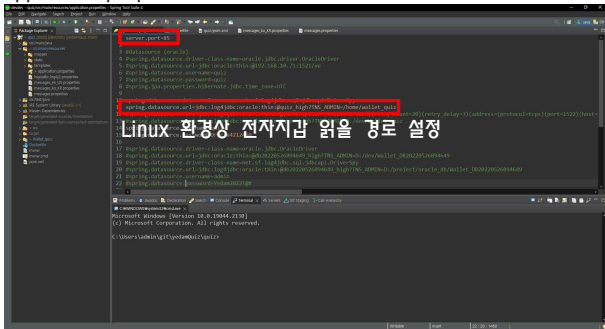
All of the transfer heads (except for Exec timeout) support substitution of [Jenkins environment variables](#).

## JENKINS

- Jenkins item 생성 - SSH and Docker 배포
- `docker pull parkseunghyeon/yedam_quiz`
- `docker stop quiz` (같은 Container 생성 방지)
- `docker rm quiz` (같은 Container 생성 방지)
- `docker run -d`
  - v /home:/home (volume 연결, 외부의 경로와 연결하여 파일 저장 등 용도 사용)
  - p 80:80 (포트 포워딩, 접속 포트:컨테이너 포트)
  - name quiz parkseunghyeon/yedam\_quiz(Container 이름 지정, 컨테이너 재 생성시 stop, rm 하기 위해 추가)

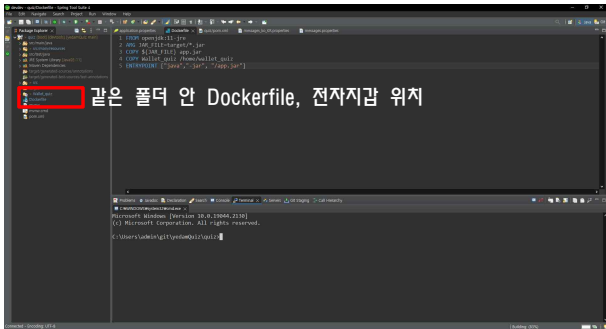
## SpringBoot Setting

- application.properties



# SpringBoot Setting

- Dockerfile



## SpringBoot

- Dockerfile

- 1) FROM - 빌드하는 이미지와 기반 이미지를 지정(베이스 이미지 지정)
- 2) ARG - docker image build 실행할때 사용하는 변수(Dockerfile에서의 변수)
- 3) COPY - 호스트에서 컨테이너로 파일 및 디렉토리를 복사(파일 복사)
- 4) ENTRYPOINT - 컨테이너를 실행 가능 파일로 사용할 때 정의하는 명령

```
FROM openjdk:11-jre
ARG JAR_FILE=target/*.jar
COPY ${JAR_FILE} app.jar
COPY Wallet_quiz /home/wallet_quiz
ENV TZ=Asia/Seoul
ENTRYPOINT ["java","-jar", "/app.jar"]
```