



# Jenkins

## CI/CD ?

**CI(Continuous Intergration)**란 개발자들이 빠른 주기로 작업한 내용을 통합 브랜치에 통합하고 빌드하는 개발 방식을 의미한다.

CI는 개인이 작업한 코드를 master 또는 develop 브랜치에 통합하는 과정에서 발생하는 이슈를 빠르게 발견하기 위해서 필요하다.

**CD**는 **Continuous Delivery** 혹은 **Continuous Depolyment** 두 용어 모두의 축약어이다. Continuous Delivery는 **공유 레포지토리까지 자동으로 Release**하는 것을 의미하며, Continuous Depolyment는 **프로덕션 레벨까지 자동으로 Deploy**하는것을 의미한다.

쉽게 말해서 개발자가 애플리케이션에 변경 사항을 작성한 후 몇 분 이내에 애플리케이션을 자동으로 실행할 수 있는 것을 의미한다.

## Jenkins ?



CI와 CD를 제공하는 툴, CI/CD의 자동화를 지원하는 툴

### Jenkins 설정

#### 1. ec2 ssh 접속

```
ssh -i 키명.pem ubuntu@i9a603.p.ssafy.io
```

#### 2. ec2에 도커 설치(<https://everydayyy.tistory.com/121>)

```
sudo apt-get upgrade
sudo apt-get update
sudo apt update
sudo apt install apt-transport-https ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
sudo apt update
sudo apt install docker-ce
docker --version
```

#### 3. jenkins 도커 이미지 다운

```
docker pull jenkins/jenkins:lts
```

#### 4. 도커 컨테이너 실행

```
docker run -d -t -p 8080:8080 -v /jenkins:/var/jenkins -v /home/ubuntu/.ssh:/root/.ssh -v /var/run/docker.sock:/var/run/docker.sock --
```

## 5. 젠킨스 첫 어드민 키 확인

```
docker exec jenkins cat /var/jenkins_home/secrets/initialAdminPassword
```

## 6. 젠킨스 플러그인 설치

- a. gitlab
- b. ssh agent

## 7. gradle 설정(<https://gksdudrb922.tistory.com/236>)

... ec2 프리티어에서 빌드시, 메모리 부족으로 인한 무한 로딩 이슈

- 싸피에서 주는 ec2를 사용
- swapping 사용하여 메모리 늘리기

### AWS EC2 free에서 발생한 메모리 문제 (jenkins + build 배포) - ec2 초기설정

✓ EC2 Ubuntu 20.04 LTS 스토리지 30port 여러개 메모리 1GB (free 요금으로 ec2를 이용할 경우 메모리 크기가 1GB이다.) local에서 만든 프로젝트를 가상 서버에 배포하기 위해 ec2 환경을 생성했다. 다만, 매번 배포 파일을 만들기 위해

📌 <https://velog.io/@chang626/AWS-EC2-free에서-발생한-메모리-문제-jenkins-build-배포>



# Jenkins