

CI/CD Pipeline : GitHub Actions

Jae Hyeon Kim

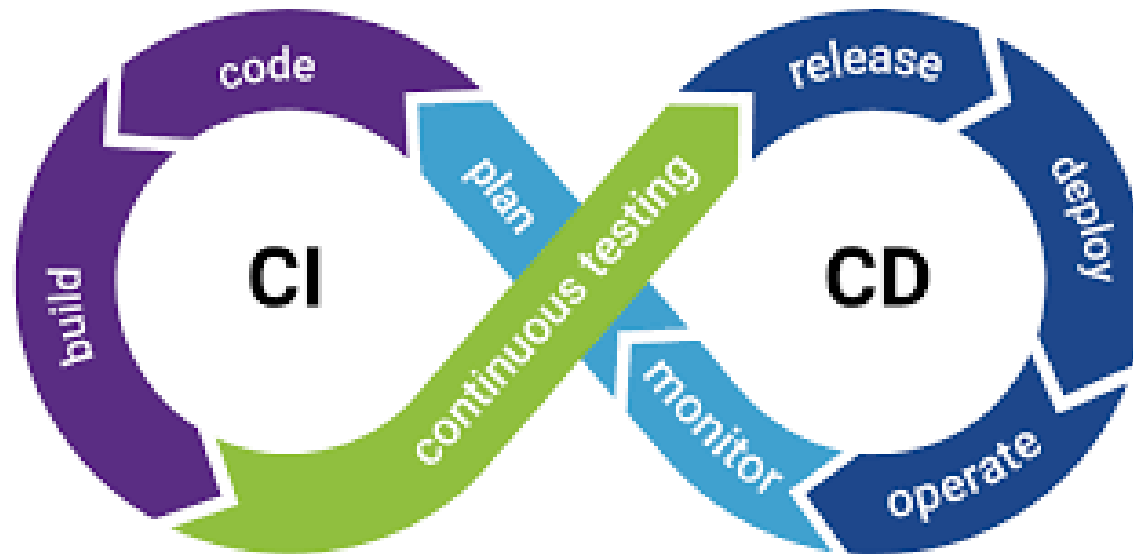
— Contents

1. CI/CD Pipeline
2. GitHub Actions

CI/CD Pipeline

— CI/CD Pipeline

- 코드를 빌드, 테스트, 배포하는 과정을 거쳐 최종적으로 출시하여 사용자에게 전달하기 위해 코드가 거치는 일련의 단계
- 파이프라인 자동화를 통해 더욱 빠른 속도로 더욱 안전하게 고품질 코드를 개발할 수 있다.



— Continuous Integration

- CI는 지속적인 통합을 의미
- CI를 구현하면 애플리케이션에 대한 새로운 코드 변경사항이 정기적으로 빌드 및 테스트되어 공유 레포지토리에 병합된다
- 여러 명의 개발자가 동시에 애플리케이션 개발과 관련된 코드 작업을 하 경우 서로 충돌하는 문제를 해결할 수 있다

— Continuous Delivery

- CD는 지속적인 제공을 의미
- 개발자들이 애플리케이션에 적용한 변경 사항이 버그 테스트를 거쳐 배포 레포지토리에 자동으로 업로드 되는것
- 운영팀은 이 레포지토리에서 애플리케이션을 실시간 프로덕션 환경으로 배포할 수 있다

- Continuous Deployment

- CD는 지속적인 배포를 의미하기도 한다
- 레포지토리에서 고객이 사용 가능한 프로덕션 환경까지 자동으로 릴리즈하는 것을 의미
- 애플리케이션 제공 속도를 저해하는 수동 프로세스로 인한 운영팀의 프로세스 과부하 문제를 해결



- CI/CD Tool

- CI/CD 툴은 팀이 개발, 배포, 테스트를 자동화하도록 지원
- Jenkins
- Circle CI
- Travis CI
- GitHub Actions



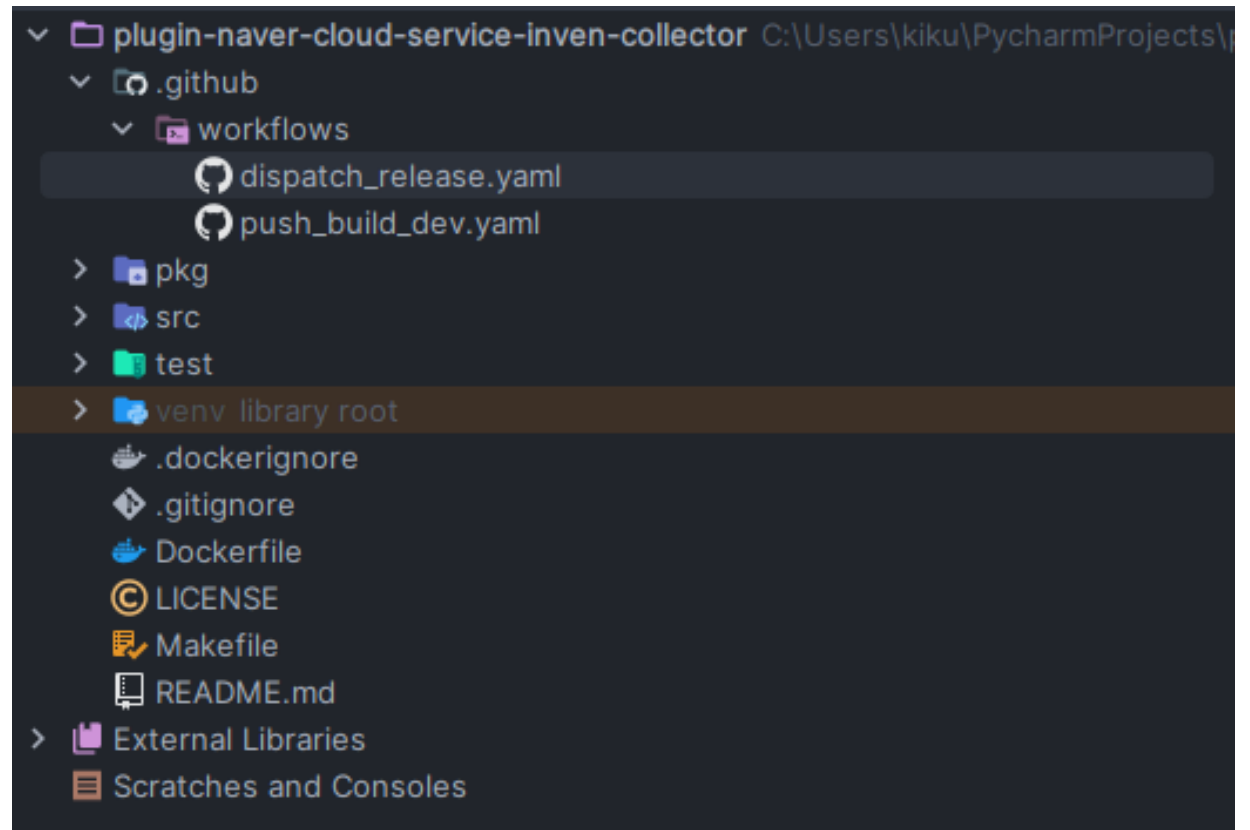
GitHub Actions

GitHub Actions

- GitHub Actions는 GitHub에서 제공하는 CI/CD를 위한 서비스
- GitHub 레포지토리에서 어떤 이벤트가 발생했을 때 특정 작업이 일어나게 하거나 주기적으로 어떤 작업들을 반복해서 실행시킬 수 있다
- 기존 CI/CD 서비스 대비 간편한 설정과 높은 접근성이 장점

– GitHub Actions - Workflows

- 가장 상위 개념으로 자동화 작업 과정
- Yaml 파일로 작성



- GitHub Actions - Workflows

- 가장 상위 개념으로 자동화 작업 과정
- YAML 파일로 작성

```
name: "[Push] Build dev"

on:
  push:
    branches:
      - main
      - release
    paths-ignore:
      - '.github/**'
      - 'src/VERSION'
  workflow_dispatch:

jobs:
  versioning:
    runs-on: ubuntu-latest
    outputs:
      version: ${ steps.versioning.outputs.VERSION }
    steps:
      - uses: actions/checkout@v2
      - name: get current date
        run: |
          sudo ln -sf /usr/share/zoneinfo/Asia/Seoul /etc/localtime
          echo "TIME=$(date +%Y%m%d.%H%M%S)" >> $GITHUB_ENV

      - name: set version with current date
        id: versioning
        run: |
          echo "VERSION=$(cat src/VERSION | cut -c 2-).${ env.TIME }" >> $GITHUB_OUTPUT
          echo "::set-output name=VERSION::$(cat src/VERSION | cut -c 2-).${ env.TIME }"
```

— GitHub Actions - Jobs

- 독립된 가상 머신 또는 컨테이너에서 돌아가는 하나의 처리 단위
- Workflows에 작성
- 모든 jobs는 동시에 실행되며 필요 시 작업 간에 의존 관계를 설정하여 작업이 실행되는 순서를 제어할 수도 있다

– GitHub Actions - Jobs

```
jobs:
  versioning:
    runs-on: ubuntu-latest
    outputs:
      version: ${ steps.versioning.outputs.VERSION }
    steps:
      - uses: actions/checkout@v2
      - name: get current date
        run: |
          sudo ln -sf /usr/share/zoneinfo/Asia/Seoul /etc/localtime
          echo "TIME=$(date +%Y%m%d.%H%M%S)" >> $GITHUB_ENV

      - name: set version with current date
        id: versioning
        run: |
          echo "VERSION=$(cat src/VERSION | cut -c 2-).${ env.TIME }" >> $GITHUB_OUTPUT
          echo "::set-output name=VERSION::$(cat src/VERSION | cut -c 2-).${ env.TIME }"
```

– GitHub Actions - Jobs

```
docker:
  if: github.repository_owner == 'kiku99'
  needs: versioning
  runs-on: ubuntu-latest
  env:
    VERSION: ${ needs.versioning.outputs.version }
  steps:
    - uses: actions/checkout@v2
      name: get service name
      run: |
        echo "SERVICE=$(echo ${ github.repository } | cut -d '/' -f2)" >> $GITHUB_ENV

    - name: Upload docker
      uses: docker/build-push-action@v1
      with:
        path: .
        repository: kiku99/${ env.SERVICE }
        username: ${ secrets.DOCKER_USERNAME }
        password: ${ secrets.DOCKER_PASSWORD }
        tags: ${ env.VERSION }
```

– GitHub Actions – Steps

- 작업(jobs)의 각 단계를 의미
- Steps는 단순한 커맨드나 스크립트가 될 수도 있고 action이라고 하는 좀 더 복잡한 명령일 수도 있다
- 커맨드나 스크립트를 실행할 때는 run 속성을 사용
- Action을 사용할 때는 uses 속성을 사용

– GitHub Actions – Steps

```
jobs:
  versioning:
    runs-on: ubuntu-latest
    outputs:
      version: ${ steps.versioning.outputs.VERSION }
    steps:
      - uses: actions/checkout@v2
      - name: get current date
        run: |
          sudo ln -sf /usr/share/zoneinfo/Asia/Seoul /etc/localtime
          echo "TIME=$(date +%Y%m%d.%H%M%S)" >> $GITHUB_ENV

      - name: set version with current date
        id: versioning
        run: |
          echo "VERSION=$(cat src/VERSION | cut -c 2-).${ env.TIME }" >> $GITHUB_OUTPUT
          echo "::set-output name=VERSION::$(cat src/VERSION | cut -c 2-).${ env.TIME }"
```

– GitHub Actions – Actions

- GitHub Actions에서 빈번하게 필요한 반복 단계를 재사용하기 용이하도록 제공되는 일종의 메커니즘
- 하나의 레포지토리 범위 안에서 여러 workflow 간에서 공유
- 공개 레포지토리를 사용하면 모든 레포지토리에서 공유 가능

– GitHub Actions – Actions

```
docker:
  if: github.repository_owner == 'kiku99'
  needs: versioning
  runs-on: ubuntu-latest
  env:
    VERSION: ${ needs.versioning.outputs.version }
  steps:
    - uses: actions/checkout@v2
      name: get service name
      run: |
        echo "SERVICE=$(echo ${ github.repository } | cut -d '/' -f2)" >> $GITHUB_ENV

    - name: Upload docker
      uses: docker/build-push-action@v1
      with:
        path: .
        repository: kiku99/${ env.SERVICE }
        username: ${ secrets.DOCKER_USERNAME }
        password: ${ secrets.DOCKER_PASSWORD }
        tags: ${ env.VERSION }
```

GitHub Actions

<> Code

Issues 2

Pull requests

Discussions

Actions

Projects 1

Wiki

Security

Insights

Settings

Actions

New workflow

All workflows

[Dispatch] Release

[Push] Build dev

Management

Caches

Runners

Beta

All workflows

Showing runs from all workflows

56 workflow runs

Event

Status

Branch

Actor

✓ Merge remote-tracking branch 'origin/release'

[Dispatch] Release #25: Commit [06e2f4b](#) pushed by kiku99

main

2 days ago

17s

...

✓ Merge remote-tracking branch 'origin/release'

[Push] Build dev #31: Commit [06e2f4b](#) pushed by kiku99

main

2 days ago

1m 25s

...

✓ Merge pull request #117 from kiku99/dev

[Push] Build dev #30: Commit [cf23932](#) pushed by kiku99

release

3 days ago

1m 58s

...

✓ Merge pull request #115 from kiku99/dev

[Push] Build dev #29: Commit [485b646](#) pushed by skduddd

release

3 days ago

1m 25s

...

✓ Merge pull request #113 from kiku99/TASK-0072-front-end-수정

[Push] Build dev #28: Commit [2f0b498](#) pushed by kiku99

release

3 days ago

1m 34s

...

✓ Merge pull request #112 from kiku99/dev

[Push] Build dev #27: Commit [97b5040](#) pushed by kiku99

release

3 days ago

1m 57s

...

✓ Merge pull request #110 from kiku99/dev

[Push] Build dev #26: Commit [86f85c1](#) pushed by kiku99

release

3 days ago

1m 53s

...

✓ Merge pull request #108 from kiku99/dev

[Push] Build dev #25: Commit [15682b5](#) pushed by skduddd

release

4 days ago

1m 23s

...

✓ Merge pull request #106 from kiku99/dev

[Push] Build dev #24: Commit [ac23267](#) pushed by skduddd

release

4 days ago

1m 47s

...

— GitHub Actions

← [Push] Build dev

✓ Merge remote-tracking branch 'origin/release' #31

Re-run all jobs

...

Summary

Jobs

✓ versioning

✓ docker

Run details

Usage

Workflow file

Triggered via push 2 days ago

kiku99 pushed → 06e2f4b main

Status

Success

Total duration

1m 25s

Artifacts

—

push_build_dev.yaml

on: push

✓ versioning

4s

✓ docker

1m 2s

Annotations

4 warnings

⚠ versioning

The following actions uses node12 which is deprecated and will be forced to run on node16: actions/checkout@v2. For more info: <https://github.blog/changelog/2023-06-13-github-actions-all-actions-will-run-on-node16-inste...>

[Show more](#)

⚠ versioning

The `set-output` command is deprecated and will be disabled soon. Please upgrade to using Environment Files. For more information see: <https://github.blog/changelog/2022-10-11-github-actions-deprecating-save-state-and-s...>

[Show more](#)

⚠ docker

The following actions uses node12 which is deprecated and will be forced to run on node16: actions/checkout@v2. For more info: <https://github.blog/changelog/2023-06-13-github-actions-all-actions-will-run-on-node16-inste...>

[Show more](#)

— GitHub Actions

← [Push] Build dev

✓ Merge remote-tracking branch 'origin/release' #31

Re-run all jobs

...

Summary

Jobs

✓ versioning

✓ docker

Run details

Usage

Workflow file

docker

succeeded 2 days ago in 1m 2s

Search logs



>	✓ Set up job	1s
>	✓ Pull docker/github-actions:v1	3s
>	✓ Run actions/checkout@v2	0s
>	✓ get service name	0s
>	✓ Upload docker	55s
>	✓ Post Run actions/checkout@v2	0s
>	✓ Complete job	0s



100

Job

Ru


do


GitHub Actions



 Search Docker Hub ctrl+K Explore Repositories Organizations Help Upgrade  kiku99

kiku99 / [Repositories](#) / [plugin-ncloud-service-inven-collector](#) / [General](#) Using 0 of 1 private repositories. [Get more](#)

[General](#) [Tags](#) [Builds](#) [Collaborators](#) [Webhooks](#) [Settings](#)

 Add a short description for this repository
The short description is used to index your content on Docker Hub and in search engines. It's visible to users in search results. Update











 **kiku99 / plugin-ncloud-service-inven-collector**

Description
This repository does not have a description 
 Last pushed: 2 days ago


Docker commands Public View
To push a new tag to this repository:

```
docker push kiku99/plugin-ncloud-service-inven-collector:tagname
```

Tags
This repository contains 22 tag(s).

Tag	OS	Type	Pulled	Pushed
 1.0.20231107.225058		Image	---	2 days ago
 1.0.20231106.215826		Image	3 days ago	3 days ago
 1.0.20231106.214737		Image	3 days ago	3 days ago
 1.0.20231106.191746		Image	3 days ago	3 days ago
 1.0.20231106.190444		Image	3 days ago	3 days ago

[See all](#) [Go to Advanced Image Management](#)

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. [Read more about automated builds](#) 
Upgrade