

# GitHub Actions 실습 - Build

주신영 bit1010@live.com

GitHub Actions로 Python앱을 빌드하기 위해 Workflow file을 생성하는 방법을 배웁니다.

## Github Actions 를 활용하여 Build

- Github 에 접속하여 새로운 Repository (github-actions-project) 생성  
Add a README file 체크(  
**파일이 하나도 없으면 커밋시 브랜치 선택이 안됨**)
- Actions 클릭
  - 매우 다양한 툴 들을 통합하기 쉽게 되어 있음
- Python Application 선택
- ▼ Workflow file 생성
  - YAML 파일
  - Github event 알아보기
    - <https://docs.github.com/en/actions/learn-github-actions/events-that-trigger-workflows>

```
name [optional]
on [required] events
  : workflow 를 시작하게 할 수 있는 Github event 의 이름
  :   on 조건(push, pull_request)

jobs [required]  jobs.<job_id>
  : 한개 또는 여러개 가능(ex build)

  : steps 하나씩 진행되는 작업
  : steps 1) can run commands, 2) setup tasks 3) run
```

- uses : selects an action (actions/ 다음에는 재사
- run : runs a command-line command

- actions 알아보기

- <https://github.com/actions>
- setup-python 검색  
CI 서버에 파이썬 환경 구성을 위한 많은 과정을 간단하게 설정 가능
- Python Application의 코드를 아래 코드로 변경
- checkout  
GitHub의 코드 저장소에 올려둔 코드를 CI 서버로 내려받은 후에 특정 브랜치로 전환

```
name: Python application

on:
  push:
    branches: [ python-ci-workflow ]
  pull_request:
    branches: [ python-ci-workflow ]

jobs:
  build:
    steps:
      - uses: actions/checkout@v2
      - name: Set up Python
        uses: actions/setup-python@v2
        with:
          python-version: "3.8"
      - name: Display Python version
        run: python -c "import sys; print(sys.version)"
```

- yml 파일 이름을 ci.yml 으로 변경
- 새로운 **브랜치 생성**해서 커밋  
Start commit → Create a new branch.. → 이름을 '**python-ci-workflow**' 로 변경 → Create pull request
- Details 클릭 → build (코드가 없어서 짧게 지나가버림)

- Actions → 에러 발생
- python-ci-workflow브랜치의 ci.yml파일 수정  
build 밑에 runs-on 추가 Commit

```
build:
    runs-on: ubuntu-latest
```

- Actions에서 build 내용 확인

▼ 파일 수정만 했는데 Action이 실행된 이유는?

GitHub사이트에서 commit하면 원격저장소에 바로 저장 되므로 push가 된 상태  
코드에서

push:

```
branches: [ python-ci-workflow ]
```

▼ 이 코드들은 어디서 실행되는 걸까?

- Github 에 의해 관리된다
  - Workflow 의 각 jobs 은 새로운 가상 환경에서 실행된다.
  - 당연히 Azure 어딘가..
- runs-on 은 실행되는 서버의 운영체제를 나타낸다.(설명만..)
    - Ubuntu, Windows, Mac

```
jobs:
  build:
    runs-on: ubuntu-latest
    strategy:
      matrix:
        os: [ubuntu-latest, windows-latest, macOS-...
```

- ci.yml 을 업데이트 한다.
  - 세 가지 운영 체제 모두에서 세 개의 빌드가 병렬로 실행된다.

```
name: Python application

on:
  push:
    branches: [ python-ci-workflow ]
  pull_request:
    branches: [ python-ci-workflow ]

jobs:
  build:
    runs-on: ${{ matrix.os }}
    strategy:
      matrix:
        os: [ubuntu-latest, macos-latest, windows-latest]
        python-version: ['3.6', '3.8']
        exclude:
          - os: macos-latest
            python-version: '3.8'
          - os: windows-latest
            python-version: '3.6'

    steps:
      - uses: actions/checkout@v2
      - name: Set up Python
        uses: actions/setup-python@v2
        with:
          python-version: ${{ matrix.python-version }}
      - name: Display Python version
        run: python -c "import sys; print(sys.version)"
```