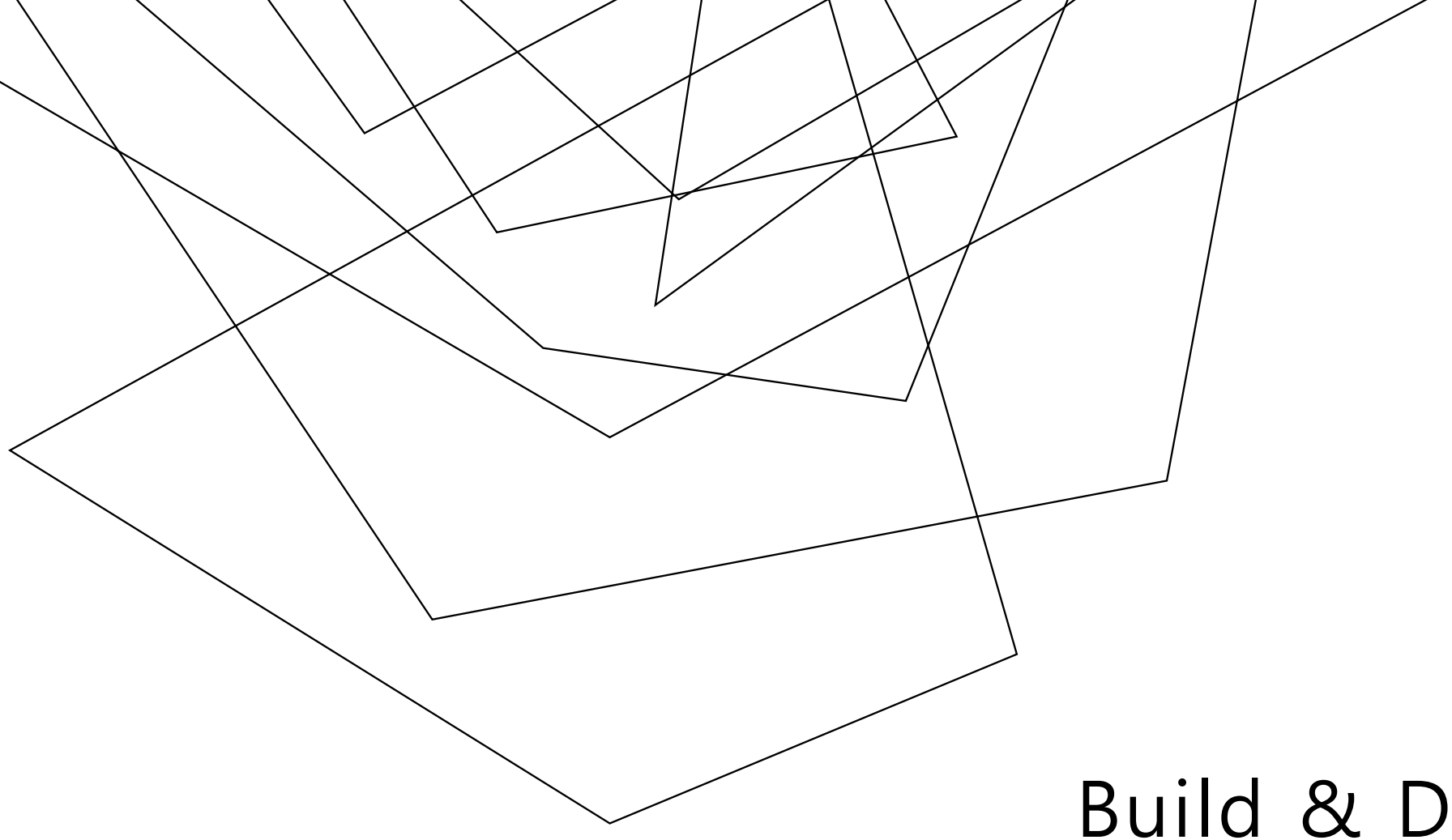
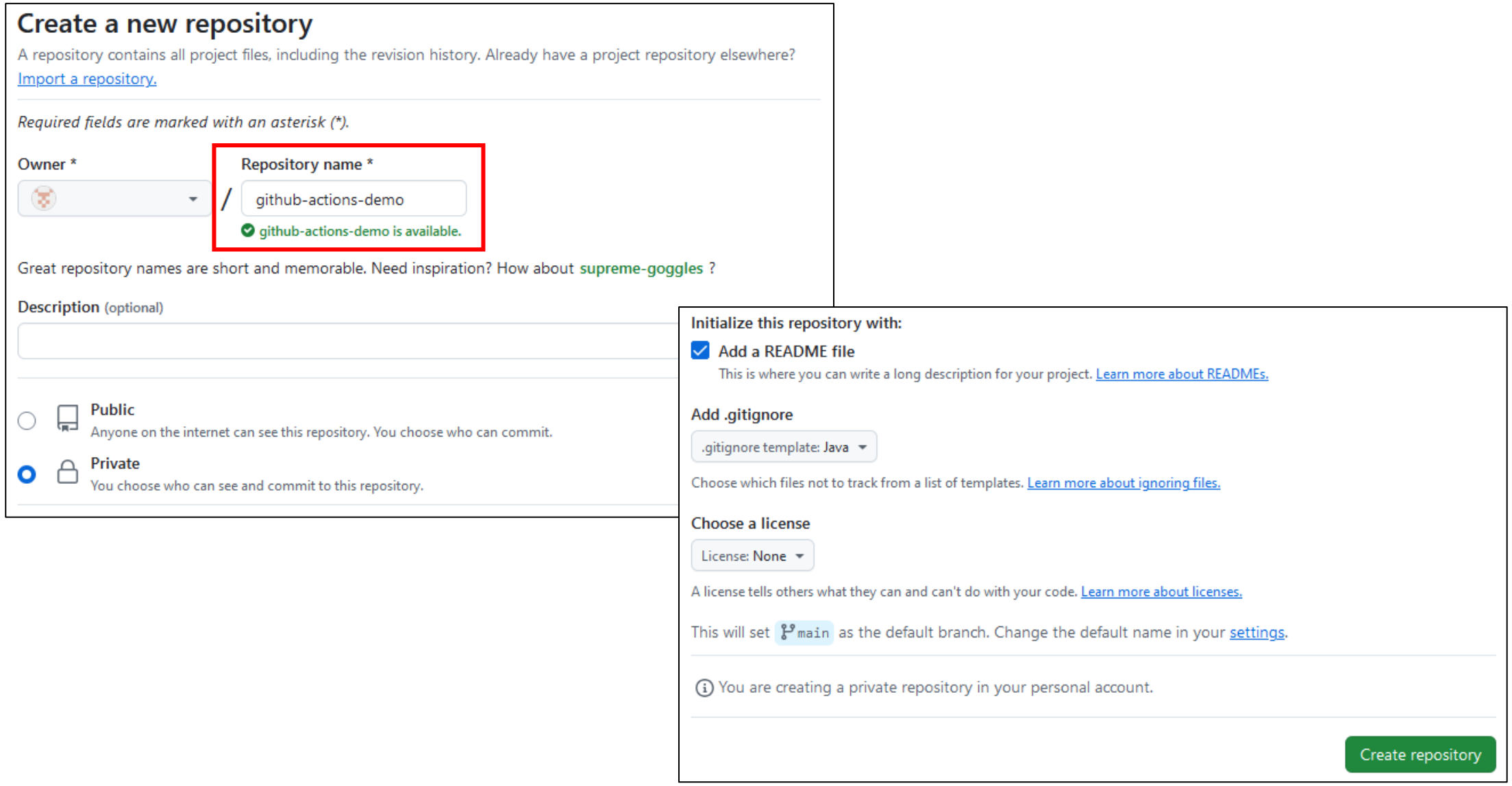
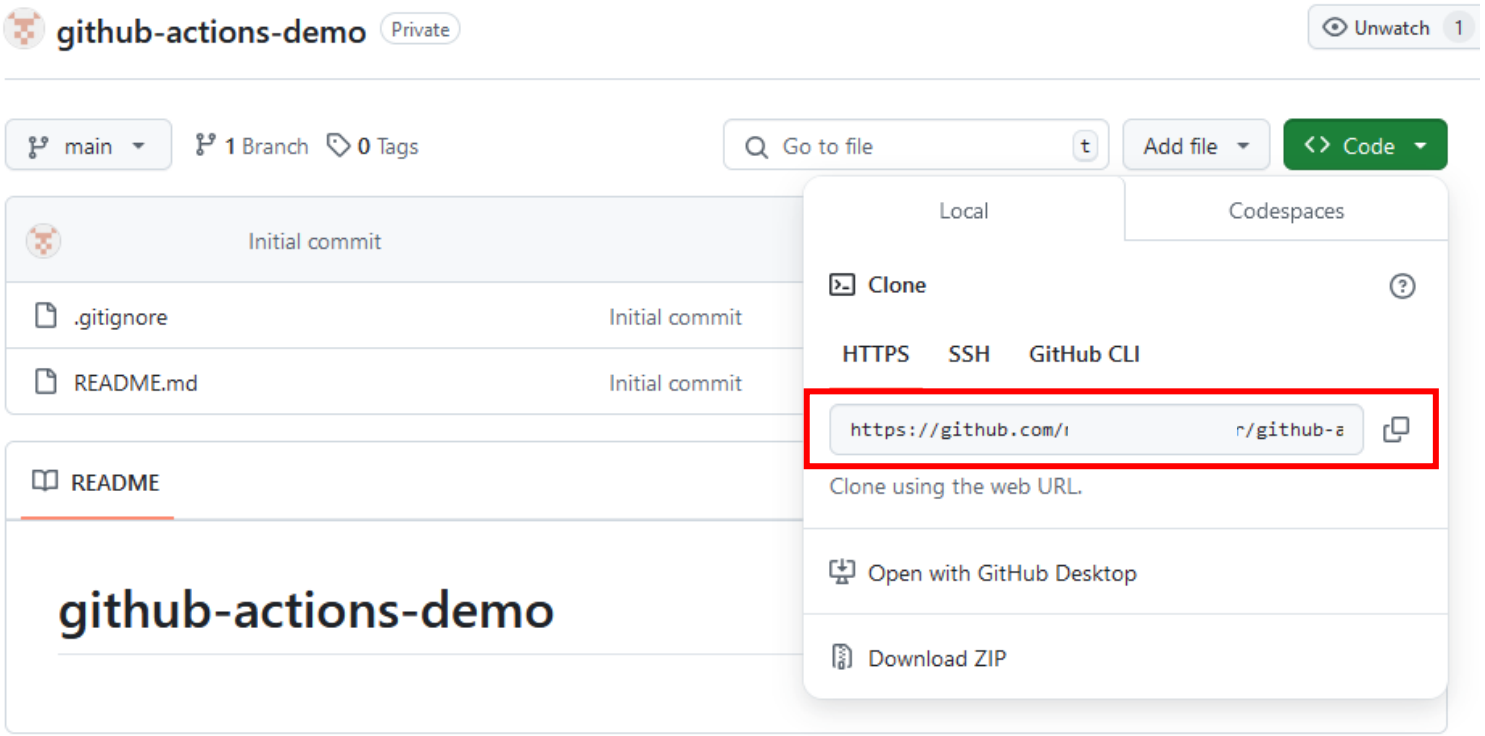


Build & Deploy with Github Actions

Github 저장소 만들기

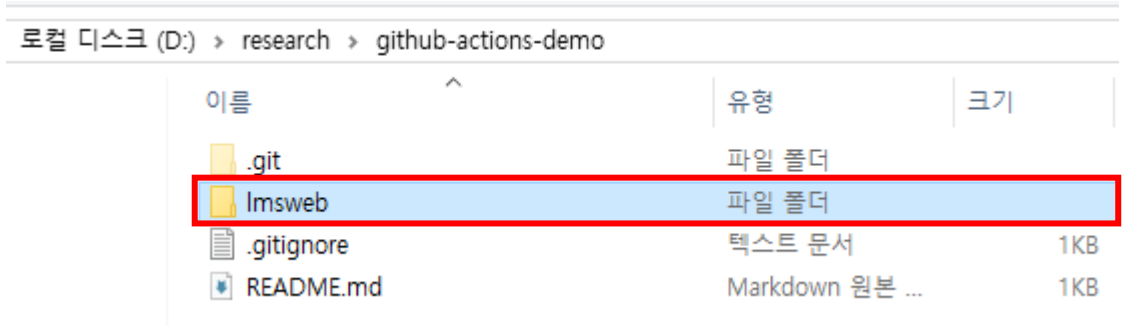


로컬 저장소 만들기




```
관리자: 명령 프롬프트
D:₩research>git clone https://github.com/                    /github-actions-demo.git
Cloning into 'github-actions-demo'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (4/4), done.
```

로컬 저장소에 샘플 프로젝트 복사 및 원격 저장소에 적용




```
관리자: 명령 프롬프트
D:Wresearch>cd github-actions-demo

D:WresearchWgithub-actions-demo>git add .
warning: in the working copy of 'lmsweb/.gitignore',
```

```
D:WresearchWgithub-actions-demo>git commit -m "add sample project"
[main e0bee6f] add sample project
 67 files changed, 34606 insertions(+)
 create mode 100644 lmsweb/.gitignore
```

```
D:WresearchWgithub-actions-demo>git push origin main
Enumerating objects: 94, done.
Counting objects: 100% (94/94), done.
Delta compression using up to 8 threads
Compressing objects: 100% (81/81), done
```

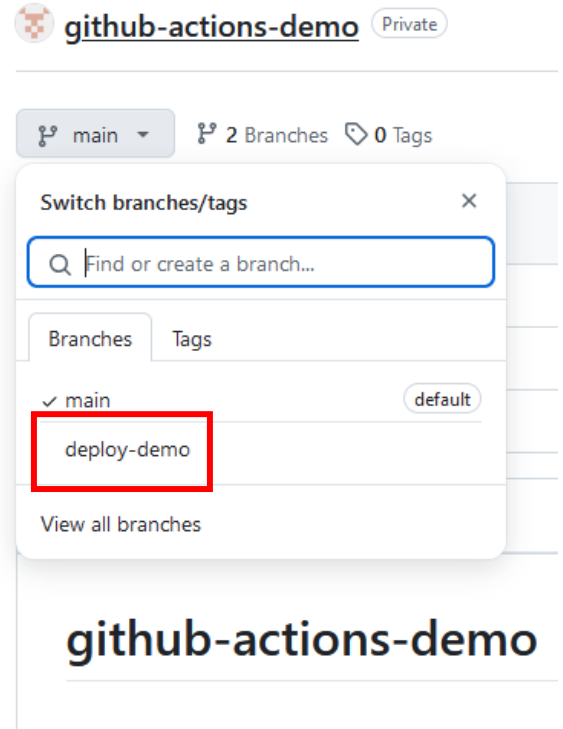
branch 만들기

main branch를 기반으로 deploy-demo branch 생성

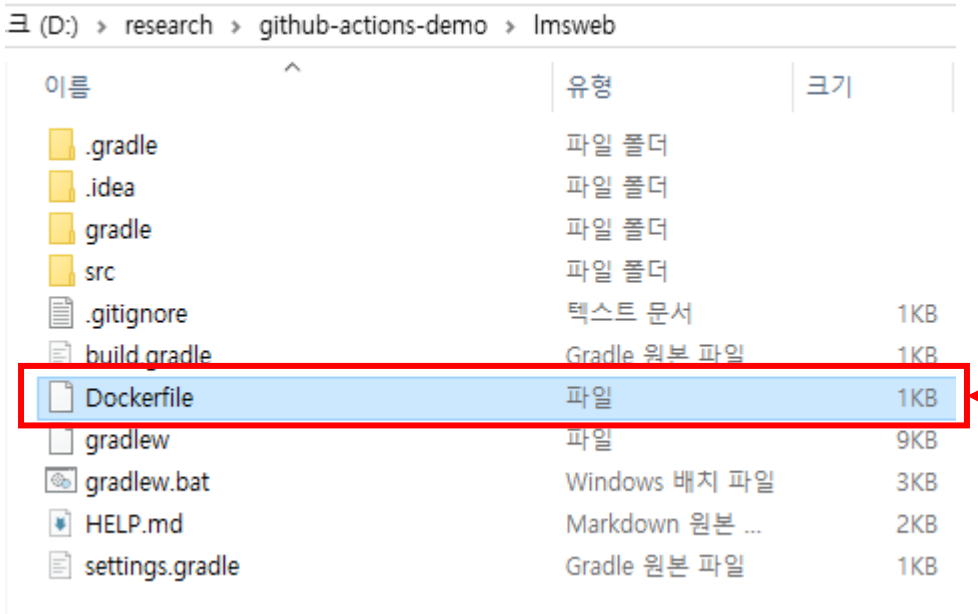
관리자: 명령 프롬프트

```
D:₩research₩github-actions-demo>git branch --list
* main

D:₩research₩github-actions-demo>git branch deploy-demo main

D:₩research₩github-actions-demo>git branch --list
  deploy-demo
* main

D:₩research₩github-actions-demo>git checkout deploy-demo
Switched to branch 'deploy-demo'

D:₩research₩github-actions-demo>git branch --list
* deploy-demo
  main
```

```
D:₩research₩github-actions-demo>git push origin deploy-demo
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'deploy-demo' on GitHub by visiting:
remote:      https://github.com/                    /github-actions-demo/pull/new/deploy-demo
remote:
To https://github.com/                    /github-actions-demo.git
 * [new branch]      deploy-demo -> deploy-demo
```

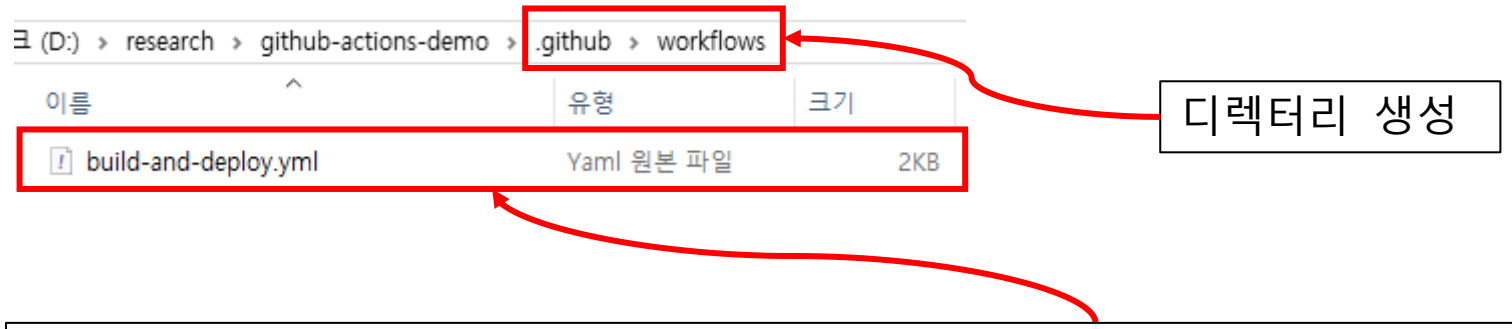


프로젝트에 Dockerfile 추가




```
FROM eclipse-temurin:17-jdk-alpine
WORKDIR /app
COPY ./build/libs/lmsweb-0.0.1-SNAPSHOT.jar lmsweb.jar
EXPOSE 8080
CMD ["java", "-jar", "lmsweb.jar"]
```

Github Actions Workflow 만들기




```yaml
name: Build and Deploy

on:
  push:
    branches: [deploy-demo]

jobs:
  build:
    runs-on: [ubuntu-latest]
    steps:
      - name: Checkout source
        uses: actions/checkout@v3
      - name: Setup Java
        uses: actions/setup-java@v3
        with:
          distribution: 'temurin'
          java-version: '17'
```

Github Actions Workflow 만들기 (계속)

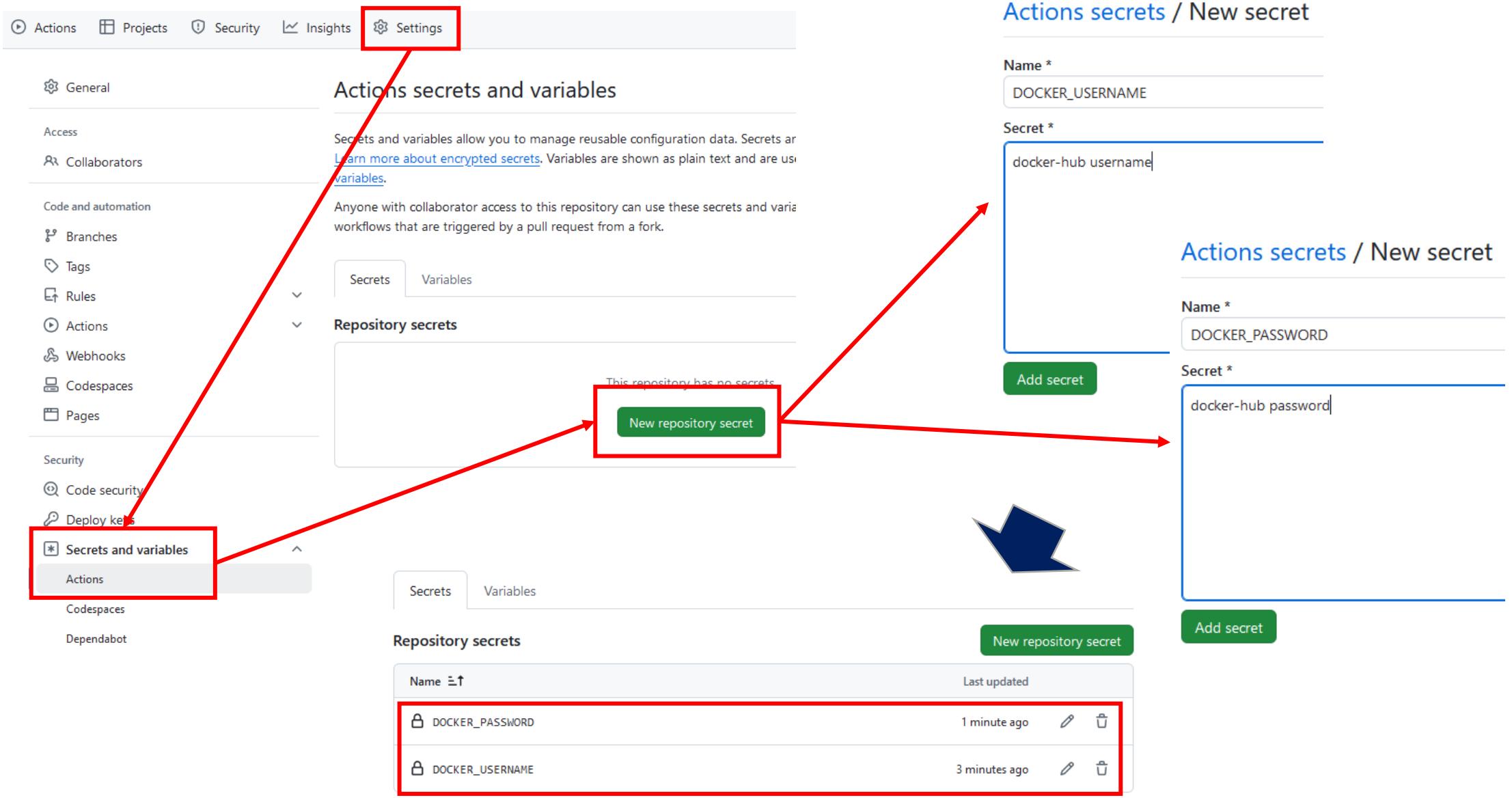
- workflow 파일 (계속)

```
      - name: Build Project
        run: lmsweb/gradlew -p lmsweb bootJar
      - name: Login to docker hub
        run: docker login -u ${{secrets.DOCKER_USERNAME}} -p ${{secrets.DOCKER_PASSWORD}}
      - name: Build docker image
        run: docker build -t shared1repo1z/lmsweb lmsweb
      - name: Publish image to docker hub
        run: docker push shared1repo1z/lmsweb:latest

  deploy:
    needs: build
    runs-on: [aws-ec2]
    steps:
      - name: Pull Image from docker hub
        run: docker pull shared1repo1z/lmsweb:latest
      - name: Delete old container
        run: docker rm -f lmsweb-container
      - name: Run docker container
        run: docker run -d -p 8080:8080 --name lmsweb-container shared1repo1z/lmsweb
```

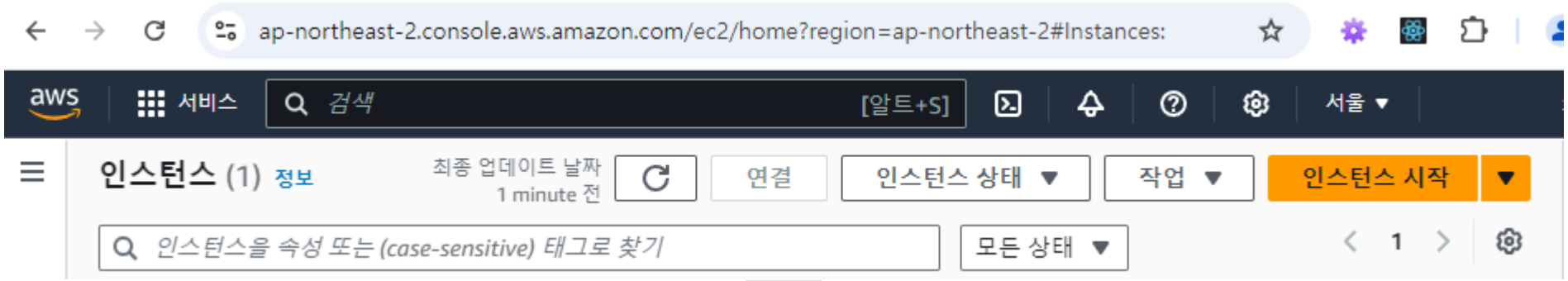
Github Settings → Security에 등록

각 사용자 계정으로 변경

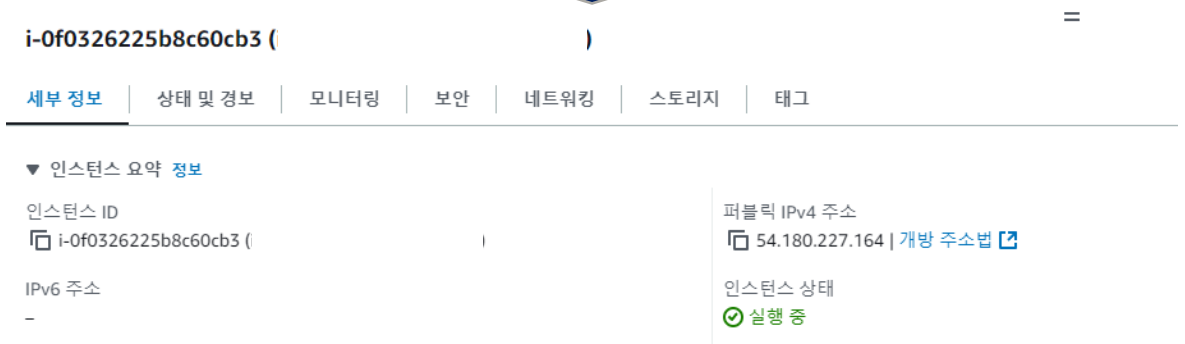
Github Secrets 환경 변수에 Docker Hub 계정 정보 등록



AWS EC2 Machine 만들기



이름 : your-machine-name
Amazon Machine Image (AMI) : Amazon Linux 2003 AMI
인스턴스 유형 : t3.medium
키 페어 이름 : (기존 키 페어 선택 또는 생성)
보안 그룹 : (기존 보안그룹 선택 또는 생성)



EC2 Machine에 Docker 등 유틸리티 설치

- Docker 설치

```
[ec2-user@i                ]$ sudo yum update -y
Last metadata expiration check: 0:09:57 ago on Tue Oct  1 06:25:08 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-5-53 ~]$ sudo yum install -y docker
Last metadata expiration check: 0:10:03 ago on Tue Oct  1 06:25:08 2024.
Dependencies resolved.
```

- Docker 서비스 시작

```
Complete!
[ec2-user@i                ]$ sudo systemctl status docker
○ docker.service - Docker Application Container Engine
     Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; preset: disabled)
     Active: inactive (dead)
TriggeredBy: ○ docker.socket
       Docs: https://docs.docker.com
[ec2-user@ip-172-31-5-53 ~]$ sudo systemctl start docker
[ec2-user@ip-172-31-5-53 ~]$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; preset: disabled)
     Active: active (running) since Tue 2024-10-01 06:37:51 UTC; 6s ago
TriggeredBy: ● docker.socket
```

EC2 Machine에 Docker 등 유틸리티 설치

- ec2-user에 docker 실행 권한 부여

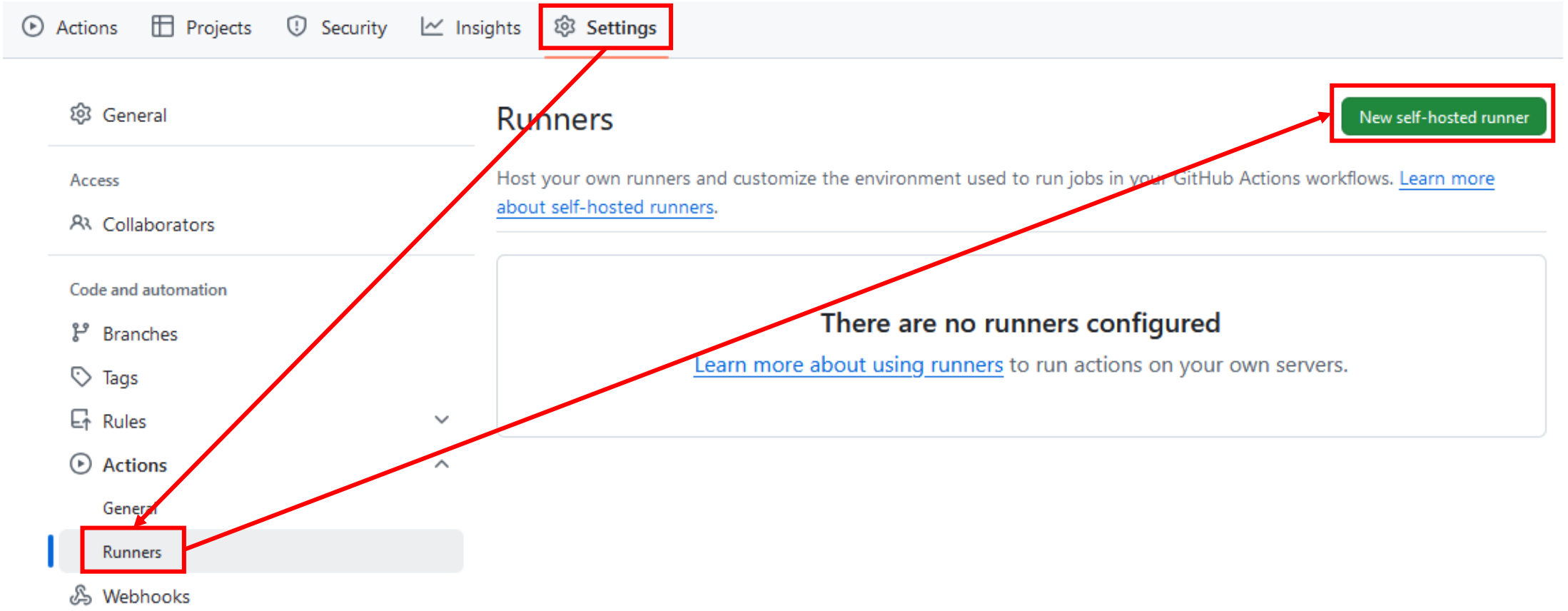
```
[ec2-user@ip-172-31-5-53 ~]$
[ec2-user@ip-172-31-5-53 ~]$
[ec2-user@ip-172-31-5-53 ~]$
[ec2-user@ip-172-31-5-53 ~]$ sudo usermod -a -G docker ec2-user
```

- ec2 machine에서 exit 실행 후 다시 접속

- dotnet-sdk 설치

```
[ec2-user@i                    ]$ sudo yum install -y dotnet-sdk-6.0
Last metadata expiration check: 0:27:16 ago on Tue Oct  1 06:25:08 2024.
Dependencies resolved.
================================================================================
```

Github Actions Runner 등록



Github Actions Runner 등록 (계속)

Runners / Add new self-hosted runner · ml-dl-instructor/github-actions-demo

Adding a self-hosted runner requires that you download, configure, and execute the GitHub Actions Runner. By downloading and configuring the GitHub Actions Runner, you agree to the GitHub Terms of Service or GitHub Corporate Terms of Service, as applicable.

Runner image

macOS | Linux | Windows

Architecture

x64

Download

```
# Create a folder
$ mkdir actions-runner && cd actions-runner
# Download the latest runner package
$ curl -o actions-runner-linux-x64-2.319.1.tar.gz -L https://github.com/actions/runner/releases/download/v2.319.1/actions-runner-linux-x64-2.319.1.tar.gz
# Optional: Validate the hash
$ echo "3f6efb7488a183e291fc2c62876e14c9ee732864173734facc85a1bfb1744464  actions-runner-linux-x64-2.319.1.tar.gz" | shasum -a 256 -c
# Extract the installer
$ tar xzf ./actions-runner-linux-x64-2.319.1.tar.gz
```

Configure

```
# Create the runner and start the configuration experience
$ ./config.sh --url https://github.com/ml-dl-instructor/github-actions-demo --token A6BXU6PKXMSCXUTQHKBEVTLG7OVZI
# Last step, run it!
$ ./run.sh
```

```
[ec2-user@ip-172-31-5-53 ~]$ mkdir actions-runner && cd actions-runner
[ec2-user@ip-172-31-5-53 actions-runner]$
[ec2-user@ip-172-31-5-53 actions-runner]$ curl -o actions-runner-linux-x64-2.319.1.tar.gz -L https://github.com/actions/runner/releases/download/v2.319.1/actions-runner-linux-x64-2.319.1.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100  208M  100  208M    0     0   120M      0  0:00:01  0:00:01 --:--:--  311M
[ec2-user@ip-172-31-5-53 actions-runner]$ tar xzf ./actions-runner-linux-x64-2.319.1.tar.gz
```

Github Actions Runner 등록 (계속)

Runners / Add new self-hosted runner · ml-dl-instructor/github-actions-demo

Adding a self-hosted runner requires that you download, configure, and execute the GitHub Actions Runner. By downloading and configuring the GitHub Actions Runner, you agree to the GitHub Terms of Service or GitHub Corporate Terms of Service, as applicable.

Runner image

○ macOS ◉ Linux ○ Windows

Architecture

x64

Download

```
# Create a folder
$ mkdir actions-runner && cd actions-runner
# Download the latest runner package
$ curl -o actions-runner-linux-x64-2.319.1.tar.gz -L https://github.com/actions/runner/releases/
download/v2.319.1/actions-runner-linux-x64-2.319.1.tar.gz
# Optional: Validate the hash
$ echo "3f6efb7488a183e291fc2c62876e14c9ee732864173734facc85a1bfb1744464  actions-runner-linux-
x64-2.319.1.tar.gz" | shasum -a 256 -c
# Extract the installer
$ tar xzf ./actions-runner-linux-x64-2.319.1.tar.gz
```

Configure

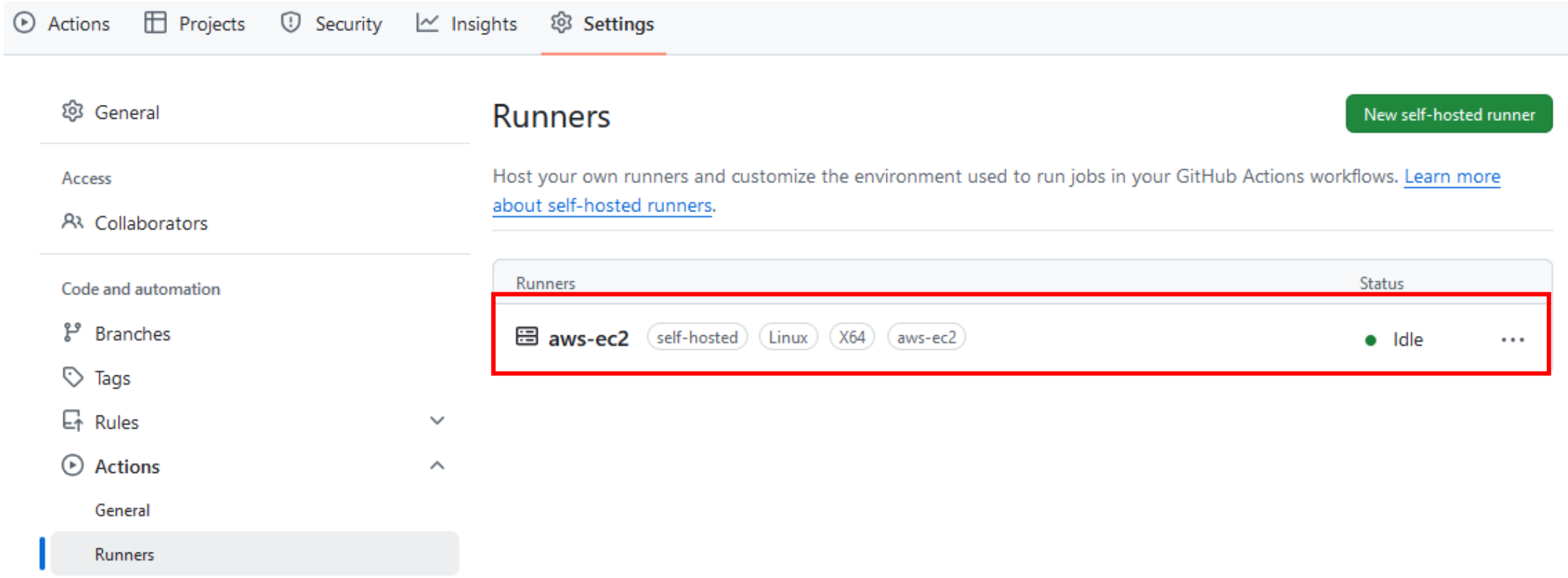
```
# Create the runner and start the configuration experience
$ ./config.sh --url https://github.com/ml-dl-instructor/github-actions-demo --token
A6BXU6PKXMSCXUTQHKBEVTLG7OVZI
# Last step, run it!
$ ./run.sh
```

```
[ec2-user@ip-172-31-5-53 actions-runner]$ ./config.sh --url https://github.com/ml-dl

--------------------------------------------------------------------------------
|        ____ _ _   _   _       _          _     _ _(_) ___  _ __  ___         |
|       / ___(_) |_| | | |_   _| |__      / \   ___| |_(_) / _ \| '_ \/ __|        |
|      | |  _| | __| |_| | | | | '_ \    / _ \ / __| __| |/ _ \| | | \__ \        |
|      | |_| | | |_|  _  | |_| | |_) |  / ___ \ (__| |_| | (_) | | | \__ \        |
|       \____|_|\__|_| |_|\__,_|_.__/  /_/   \_\___|\__|_|\___/|_| |_|___/        |
|                                                                              |
|                       Self-hosted runner registration                        |
|                                                                              |
--------------------------------------------------------------------------------

# Authentication


√ Connected to GitHub

# Runner Registration

Enter the name of the runner group to add this runner to: [press Enter for Default]

Enter the name of runner: [press Enter for ip-172-31-5-53] aws-ec2

This runner will have the following labels: 'self-hosted', 'Linux', 'X64'
Enter any additional labels (ex. label-1,label-2): [press Enter to skip] aws-ec2

√ Runner successfully added
√ Runner connection is good

# Runner settings

Enter name of work folder: [press Enter for _work]

√ Settings Saved.
```

```
[ec2-user@ip-172-31-5-53 actions-runner]$ ./run.sh

√ Connected to GitHub

Current runner version: '2.319.1'
2024-10-01 07:29:16Z: Listening for Jobs
```

Github Actions Runner 등록 (계속)

- 등록 확인



실행 확인

- add → commit → push 실행

- github 저장소에서 workflow 실행 확인