

# PLASS-NIA 인공지능학습데이터구축사업 (2021)

---

4팀(김도현, 전은성)

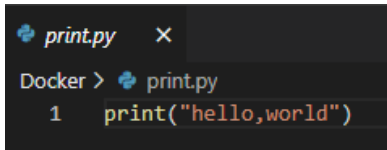
# 목차

1. Docker 이미지 생성 테스트
2. 생성된 이미지 공유
3. Docker 이미지 생성
4. 향후 계획



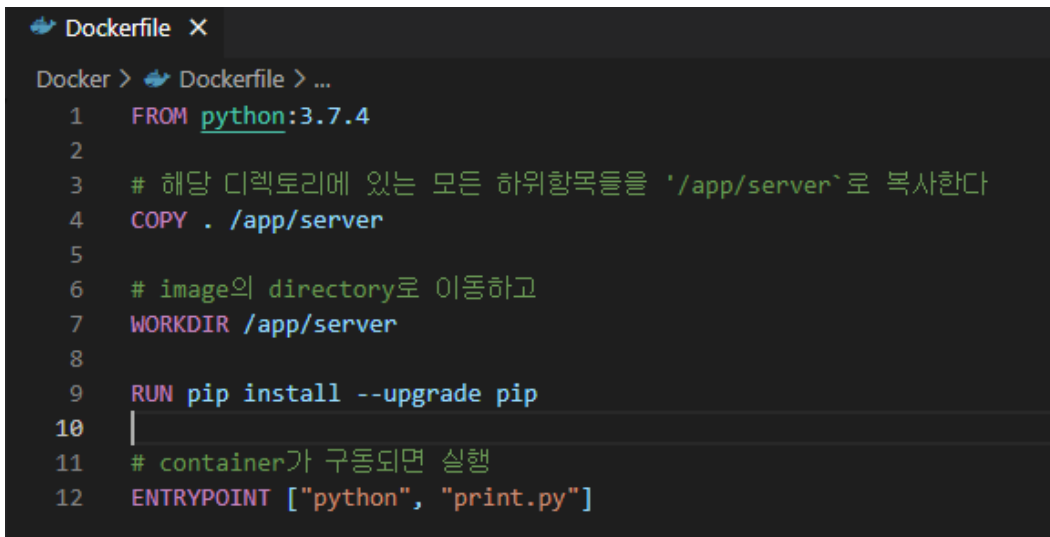
# 1. Docker 이미지 생성 테스트

- Docker 이미지 생성방법을 위해 테스트 진행



```
print.py ×
Docker > print.py
1 print("hello,world")
```

- 테스트를 위해 "Hello,world"를 출력하는 간단한 print.py 파일 생성



```
Dockerfile ×
Docker > Dockerfile > ...
1 FROM python:3.7.4
2
3 # 해당 디렉토리에 있는 모든 하위항목들을 '/app/server'로 복사한다
4 COPY . /app/server
5
6 # image의 directory로 이동하고
7 WORKDIR /app/server
8
9 RUN pip install --upgrade pip
10
11 # container가 구동되면 실행
12 ENTRYPOINT ["python", "print.py"]
```

- 생성한 print.py를 실행하는 Docker 이미지 생성하기 위해 Dockerfile을 작성


# 1. Docker 이미지 생성 테스트

210.94.194.107 - PuTTY

```
NIA_AI_DATASET_2021-C3D@G292-Z40-00:~/Docker$ docker build -t python_print .
Sending build context to Docker daemon  949.2MB
Step 1/5 : FROM python:3.7.0
--> a187104266fb
Step 2/5 : COPY . /app/server
--> b9f8657b2379
Step 3/5 : WORKDIR /app/server
--> Running in 82b0dclb2b95
Removing intermediate container 82b0dclb2b95
--> 1d19619d4c34
Step 4/5 : RUN pip install --upgrade pip
--> Running in 10fb47f85c8c
Collecting pip
  Downloading https://files.pythonhosted.org/packages/a4/6d/6463d49a933f547439d6b5b98b46af8742cc03ae83543e4d7688c2420f8b/pip-21.3.1-py3-none-any.whl (1.7MB)
Installing collected packages: pip
  Found existing installation: pip 18.1
  Uninstalling pip-18.1:
    Successfully uninstalled pip-18.1
  Successfully installed pip-21.3.1
Removing intermediate container 10fb47f85c8c
--> eeaf1e5c9122
Step 5/5 : ENTRYPOINT ["python", "print.py"]
--> Running in 30326b4892fe
Removing intermediate container 30326b4892fe
--> 1f4107459a0d
Successfully built 1f4107459a0d
Successfully tagged python_print:latest
```

- docker build -t [이미지명] 명령어를 사용하여 python\_print라는 이름을 가진 이미지를 생성

# 1. Docker 이미지 생성 테스트

 210.94.194.107 - PuTTY

```
NIA_AI_DATASET_2021-C3D@G292-Z40-00:~/Docker$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
python_print	latest	1f4107459a0d	About a minute ago	1.89GB
python_print	latest	7252b76a467	14 minutes ago	1.89GB

- docker images 명령어를 사용하여 python\_print 이미지가 생성되었는지 확인

 210.94.194.107 - PuTTY

```
NIA_AI_DATASET_2021-C3D@G292-Z40-00:~/Docker$ docker run python_print  
hello,world
```

- docker run [이미지 명] 명령어를 사용하여 docker 실행
- 실행결과 print.py의 실행 결과인 "hello, world" 가 출력

## 2. 생성된 이미지 공유

 210.94.194.107 - PuTTY

```
NIA_AI_DATASET_2021-C3D@G292-Z40-00:~/Docker$ docker save -o docker_test1.tar python_print
```

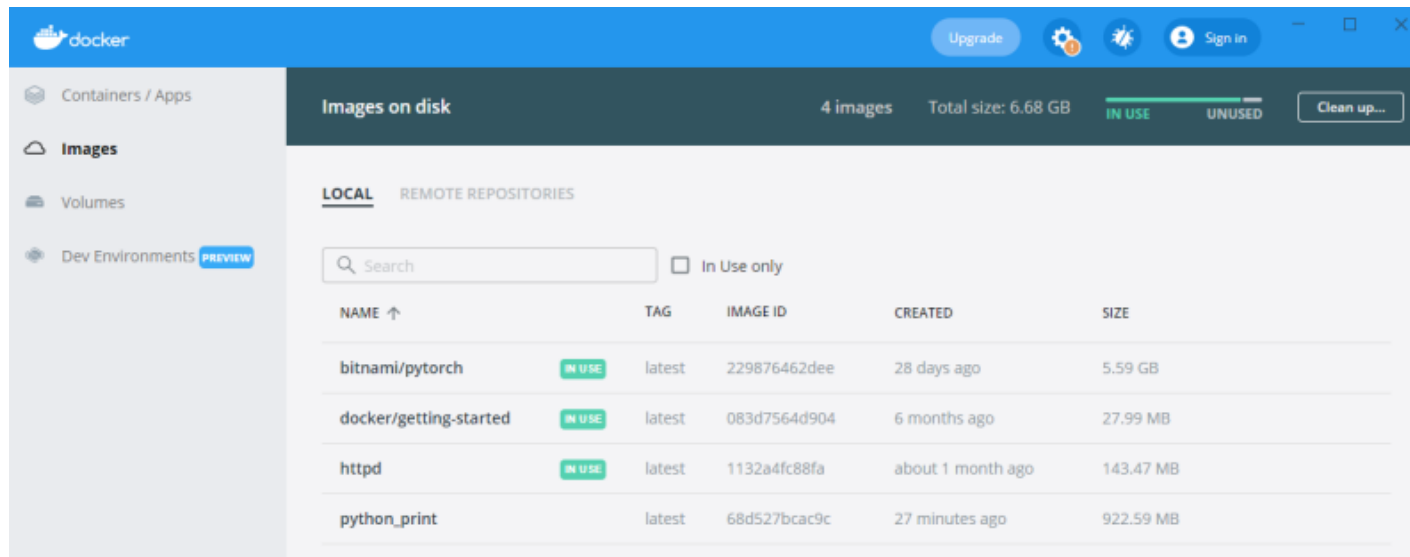
- docker save -o [파일이름.tar] [이미지 명] 명령어를 사용하여 이미지를 tar형식의 파일로 생성

```
(base) C:\Users\Jeon>docker load -i docker_test1.tar
f715ed19c28b: Loading layer [=====>] 105.5MB/105.5MB
8bb25f9cdc41: Loading layer [=====>] 23.99MB/23.99MB
08a01612ffca: Loading layer [=====>] 7.994MB/7.994MB
1191b3f5862a: Loading layer [=====>] 146.4MB/146.4MB
9978d084fd77: Loading layer [=====>] 570.6MB/570.6MB
d62f0ea9a15e: Loading layer [=====>] 17.5MB/17.5MB
8eb8b96ceebb: Loading layer [=====>] 70.76MB/70.76MB
c134b6c064f6: Loading layer [=====>] 4.608kB/4.608kB
4c9ede4ddbda: Loading layer [=====>] 6.342MB/6.342MB
d0223a944527: Loading layer [=====>] 3.584kB/3.584kB
84de8aad9567: Loading layer [=====>] 5.12kB/5.12kB
Loaded image: python_print:latest
```

- 생성된 tar형식 파일의 이미지를 실행하고자 하는 PC로 공유
- 공유된 tar형식 파일의 이미지와 docker load -i [파일이름.tar] 명령어를 사용하여 이미지를 로드

## 2. 생성된 이미지 공유

```
(base) C:\#Users\#Jeon>docker images
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
python_print        latest     68d527bcac9c  27 minutes ago  923MB
bitnami/pytorch     latest     229876462dee  4 weeks ago    5.59GB
httpd               latest     1132a4fc88fa  5 weeks ago    143MB
docker/getting-started latest     083d7564d904  5 months ago   28MB
```



- docker images 명령어와 docker 프로그램을 사용하여 로딩으로 python\_print 이미지가 생성되었는지 확인

### 3. Docker 이미지 생성

```
Dockerfile X
Docker > Dockerfile > RUN
1 FROM python:3.7.0
2
3 # 해당 디렉토리에 있는 모든 하위항목들을 '/app/server'로 복사한다
4 COPY . /app/server
5
6 # image의 directory로 이동하고
7 WORKDIR /app/server
8
9 RUN pip install --upgrade pip
10
11 # 필요한 의존성 file들 설치
12 RUN pip install --requirement requirements.txt
13
14
15 # container가 구동되면 실행
16 ENTRYPOINT ["python", "test(top5).py"]
```

- Print.py와는 달리 다양한 라이브러리를 사용하기 때문에 환경 설치를 위해 requirement.txt가 필요

```
(ENV3D) NIA_AI_DATASET_2021-C3D@G292-740-00:~$ pip freeze > requirement.txt
```

- Pip freeze 명령어를 사용하여 requirement.txt를 생성



# 3. Docker 이미지 생성

```
210.94.194.107 - PuTTY
NIA_AI_DATASET_2021-C3D@G292-240-00:~/Docker$ docker build -t python_test .
Sending build context to Docker daemon  2.862GB
Step 1/6 : FROM python:3.7.0
--> a187104266fb
Step 2/6 : COPY . /app/server
--> d66ef66c41bb
Step 3/6 : WORKDIR /app/server
--> Running in 890cc34d936b
Removing intermediate container 890cc34d936b
--> 545aa7c6a589
Step 4/6 : RUN pip install --upgrade pip
--> Running in 543c4d3fdea9
Collecting pip
  Downloading https://files.pythonhosted.org/packages/a4/6d/6463d49a933f547439d6b5b98b46af8742cc03ae83543e4d7688c2420f8b/pip-21.3.1-py3-none-any.whl (1.7MB)
Installing collected packages: pip
  Found existing installation: pip 18.1
    Uninstalling pip-18.1:
      Successfully uninstalled pip-18.1
Successfully installed pip-21.3.1
Removing intermediate container 543c4d3fdea9
--> 5397b98875ef
Step 5/6 : RUN pip install --requirement requirements.txt
--> Running in e18b58dd5932
Collecting altgraph==0.17.2
  Downloading altgraph-0.17.2-py2.py3-none-any.whl (21 kB)
Collecting astor==0.8.1
  Downloading astor-0.8.1-py2.py3-none-any.whl (27 kB)
Collecting astunparse==1.6.3
  Downloading astunparse-1.6.3-py2.py3-none-any.whl (12 kB)
```

```
ERROR: Cannot install -r requirements.txt (line 41) and h5py==2.10.0 because these package versions have conflicting dependencies.
ERROR: ResolutionImpossible: for help visit https://pip.pypa.io/en/latest/user_guide/#fixing-conflicting-dependencies

The conflict is caused by:
  The user requested h5py==2.10.0
  tensorflow-gpu 2.6.0 depends on h5py~=3.1.0

To fix this you could try to:
1. loosen the range of package versions you've specified
2. remove package versions to allow pip attempt to solve the dependency conflict

The command '/bin/sh -c pip install --requirement requirements.txt' returned a non-zero code: 1
```

- 이미지 생성 중 requirement.txt에 기록된 라이브러리 설치 중 tensorflow-gpu 버전의 문제가 발생
- 환경 재설정 필요

## 4. 향후 계획

---

- 도커 환경 오류 수정
- 새로 제작된 E2ON영상 데이터셋 추가
- 추가된 데이터셋으로 학습 진행
- 테스트 결과 확인

# 감사합니다

---

4팀(김도현, 전은성)