

Practical-7

Deployment of ML project using Streamlit.

Task 1: Ensure that the required libraries are installed

```
streamlit==1.10.0 pandas==1.2.3  
scikit-learn==0.24.1
```

Task 2: Create the docker file using the steps described in theory material.

a) Create a Dockerfile :

```
FROM python:3.8-slim  
WORKDIR /app  
COPY . /app  
RUN pip install --no-cache-dir -r requirements.txt  
EXPOSE 80  
ENV NAME World  
CMD ["gunicorn", "--bind", "0.0.0.0:80", "app:app"]
```

b) Create a requirement.txt file :

```
scikit-learn==0.24.2  
pandas==1.3.3  
numpy==1.21.2 flask==2.1.0  
gunicorn==20.1.0
```

c) Create a Streamlit file :



d) Create a Docker Image :

```

PS D:\Desktop\stream> docker build -t stream .
[+] Building 3.1s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 577B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [auth] library/python:pull token for registry-1.docker.io
=> [1/4] FROM docker.io/library/python:3.8-slim@sha256:19e07fa24813e88b04e606772213bd03ba044637cc939a211e28ccf997a9162a
=> => transferring context: 93B
=> CACHED [2/4] WORKDIR /app
=> CACHED [4/4] RUN pip install --no-cache-dir -r requirements.txt
=> exporting to image
=> => exporting layers
=> => writing image sha256:e56ed293e3b764515644f7bb676072f8e666754267516a1758d42045027a5b2f

```

Check the image is created or not :

```

PS D:\Desktop\stream> docker images

```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
stream	latest	e56ed293e3b7	16 minutes ago	495MB

Task 4: Run the docker container to execute the docker image and host the machine learning model using streamlit app server.

```

PS D:\Desktop\stream> docker run -p 8080:8501 stream

```

Collecting usage statistics. To deactivate, set browser.gatherUsageStats to False.

You can now view your Streamlit app in your browser.

Network URL: <http://172.17.0.4:8501>
 External URL: <http://103.238.106.204:8501>