

Docker Task 2

1. Install docker compose on EC2 instance on AWS

```
[ec2-user@ip-172-31-22-49 composetest]$ sudo curl -L "https://github.com/docker/compose/releases/download/1.23.2/docker-  
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose  
% 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 11.2M 100 11.2M    0     0 29.0M      0  --:--:-- --:--:-- --:--:-- 29.0M  
[ec2-user@ip-172-31-22-49 composetest]$ sudo chmod +x /usr/local/bin/docker-compose  
[ec2-user@ip-172-31-22-49 composetest]$ docker-compose --version  
docker-compose version 1.23.2, build 1110ad01  
[ec2-user@ip-172-31-22-49 composetest]$
```

2. Docker version on EC2

```
[ec2-user@ip-172-31-22-49 composetest]$ docker-compose --version
```

docker-compose version 1.23.2, build 1110ad01

3. Create composetest directory , and app.py file in it with Python code.

```
[ec2-user@ip-172-31-22-49 ~]$ mkdir composetest  
[ec2-user@ip-172-31-22-49 ~]$ cd composetest/  
[ec2-user@ip-172-31-22-49 composetest]$ touch app.py  
[ec2-user@ip-172-31-22-49 composetest]$ vi app.py  
[ec2-user@ip-172-31-22-49 composetest]$  
[ec2-user@ip-172-31-22-49 composetest]$ cat app.py  
import time  
  
import redis  
from flask import Flask  
  
app = Flask(__name__)  
cache = redis.Redis(host='redis', port=6379)  
  
def get_hit_count():  
    retries = 5  
    while True:  
        try:  
            return cache.incr('hits')  
        except redis.exceptions.ConnectionError as exc:  
            if retries == 0:  
                raise exc  
            retries -= 1  
            time.sleep(0.5)  
  
@app.route('/')
```

b. create requirement file and add following 2 services

```
[ec2-user@ip-172-31-22-49 composetest]$ touch requirements.txt
[ec2-user@ip-172-31-22-49 composetest]$ vi requirements.txt
[ec2-user@ip-172-31-22-49 composetest]$ cat requirements.txt
flask
redis
```

c. Create dockerfile and add changes

```
[ec2-user@ip-172-31-22-49 composetest]$ cat dockerfile
# syntax=docker/dockerfile:1
FROM python:3.10-alpine
WORKDIR /code
ENV FLASK_APP=app.py
ENV FLASK_RUN_HOST=0.0.0.0
RUN apk add --no-cache gcc musl-dev linux-headers
COPY requirements.txt requirements.txt
RUN pip install -r requirements.txt
EXPOSE 5000
COPY . .
CMD ["flask", "run", "--debug"]
```

d. Now create compose.yaml file which can call services and ports

```
[ec2-user@ip-172-31-22-49 composetest]$ cat compose.yaml
services:
  web:
    build: .
    ports:
      - "8000:5000"
  redis:
    image: "redis:alpine"
```

e. After running docker-compose command it gives error

ERROR: client version 1.21 is too old. Minimum supported API version is 1.24, please upgrade your client to a newer version

I tried to upgrade client but it not works.