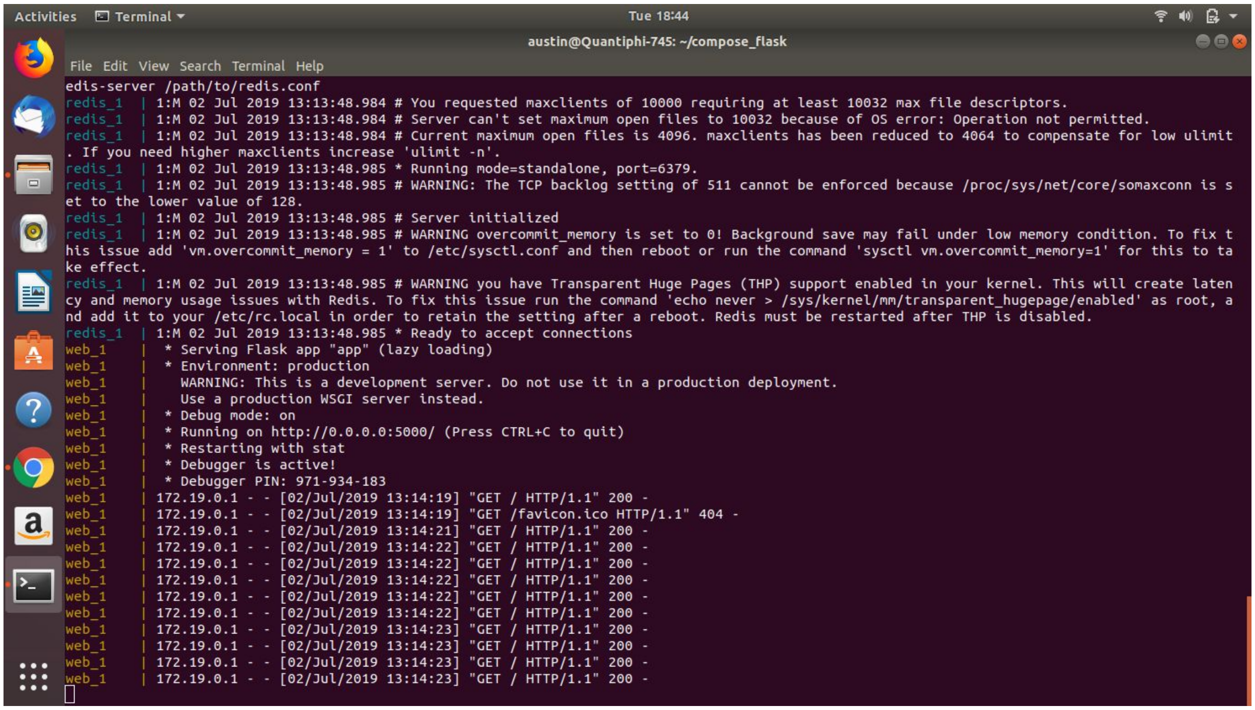
1: Create a web application in Flask which maintains a hit counter in redis, as whenever user hits the page counter increments and gets updated in redis. Build this environment using docker compose

Steps:-

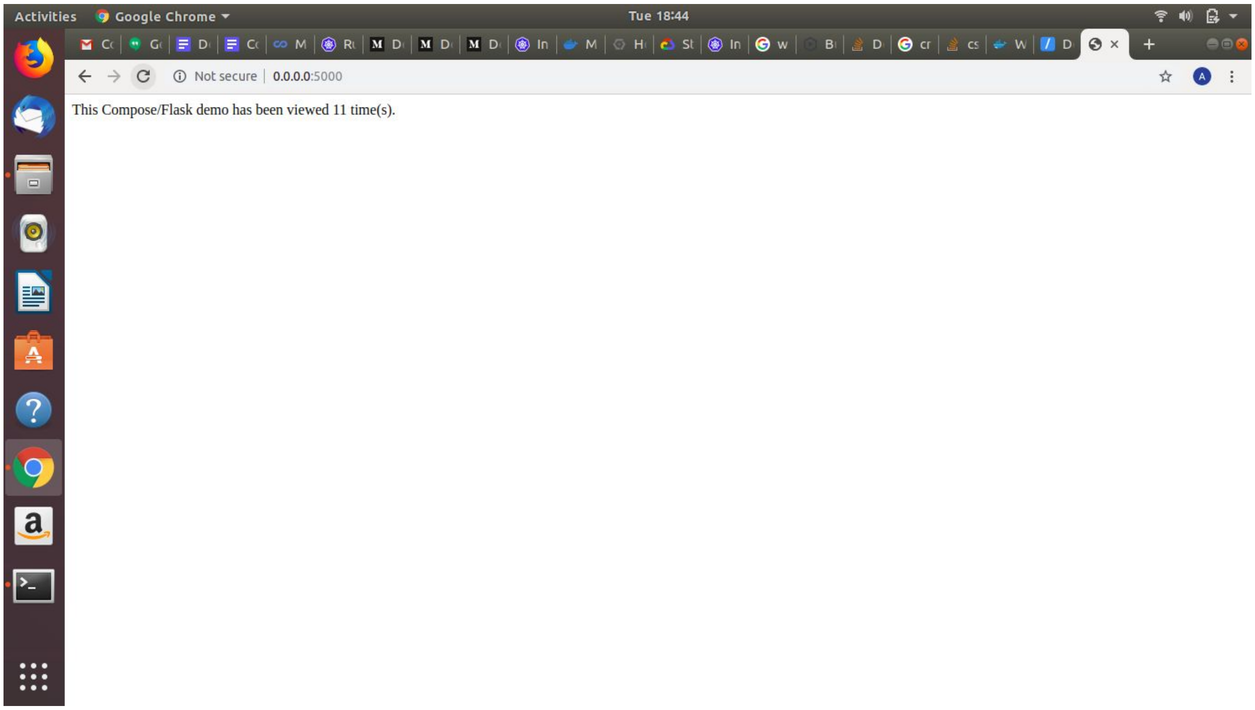
1. Create a directory having the following files - app.py , Dockerfile , requirements.txt , docker-compose.yml
2. Create a requirements.txt with the following content :-
   ```
   flask
   redis
   ```
3. Create a Dockerfile having the following content
   ```
   FROM python:2.7
   ADD . /code
   WORKDIR /code
   RUN pip install -r requirements.txt
   CMD python app.py
   ```
4. Create docker-compose.yml file with the following content:-
   ```
   version: '2'
   services:
     web:
       build: .
       ports:
         - "5000:5000"
     redis:
       image: redis
   ```
5. Create app.py with the following content :-
   ```
   # compose_flask/app.py
   from flask import Flask
   from redis import Redis

   app = Flask(__name__)
   redis = Redis(host='redis', port=6379)

   @app.route('/')
   def hello():
     redis.incr('hits')
     return 'Hey Docker! The file has been viewed %s time(s).' % redis.get('hits')


   if __name__ == "__main__":
     app.run(host="0.0.0.0", debug=True)
   ```
6. Now build your docker with the following command
   ```
   sudo docker build -t compose_flask .
   ```
7. Now run
   ```
   sudo docker-compose up
   ```



Activities
Terminal
Tue 18:44
austin@Quantiphi-745: ~/compose_flask
File Edit View Search Terminal Help

```
edis-server /path/to/redis.conf
redis_1  | 1:M 02 Jul 2019 13:13:48.984 # You requested maxclients of 10000 requiring at least 10032 max file descriptors.
redis_1  | 1:M 02 Jul 2019 13:13:48.984 # Server can't set maximum open files to 10032 because of OS error: Operation not permitted.
redis_1  | 1:M 02 Jul 2019 13:13:48.984 # Current maximum open files is 4096. maxclients has been reduced to 4064 to compensate for low ulimit
. If you need higher maxclients increase 'ulimit -n'.
redis_1  | 1:M 02 Jul 2019 13:13:48.985 * Running mode=standalone, port=6379.
redis_1  | 1:M 02 Jul 2019 13:13:48.985 # WARNING: The TCP backlog setting of 511 cannot be enforced because /proc/sys/net/core/somaxconn is s
et to the lower value of 128.
redis_1  | 1:M 02 Jul 2019 13:13:48.985 # Server initialized
redis_1  | 1:M 02 Jul 2019 13:13:48.985 # WARNING overcommit_memory is set to 0! Background save may fail under low memory condition. To fix t
his issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to ta
ke effect.
redis_1  | 1:M 02 Jul 2019 13:13:48.985 # WARNING you have Transparent Huge Pages (THP) support enabled in your kernel. This will create laten
cy and memory usage issues with Redis. To fix this issue run the command 'echo never > /sys/kernel/mm/transparent_hugepage/enabled' as root, a
nd add it to your /etc/rc.local in order to retain the setting after a reboot. Redis must be restarted after THP is disabled.
redis_1  | 1:M 02 Jul 2019 13:13:48.985 * Ready to accept connections
web_1    |  * Serving Flask app "app" (lazy loading)
web_1    |  * Environment: production
web_1    |    WARNING: This is a development server. Do not use it in a production deployment.
web_1    |    Use a production WSGI server instead.
web_1    |  * Debug mode: on
web_1    |  * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
web_1    |  * Restarting with stat
web_1    |  * Debugger is active!
web_1    |  * Debugger PIN: 971-934-183
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:19] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:19] "GET /favicon.ico HTTP/1.1" 404 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:21] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:22] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:22] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:22] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:22] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:22] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:23] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:23] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:23] "GET / HTTP/1.1" 200 -
web_1    | 172.19.0.1 - - [02/Jul/2019 13:14:23] "GET / HTTP/1.1" 200 -
```

Each refresh is counted as a hit



This Compose/Flask demo has been viewed 11 time(s).