

Experiment 2 - Docker on Linux/Windows/Mac OS

Name:	Chaitanya Madhav R
SRN:	PES1UG20CS634
Section:	K

Screenshots: 1a.

```
C:\Users\chait\Desktop\PES1UG20CS634>docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

C:\Users\chait\Desktop\PES1UG20CS634>
```

2a.

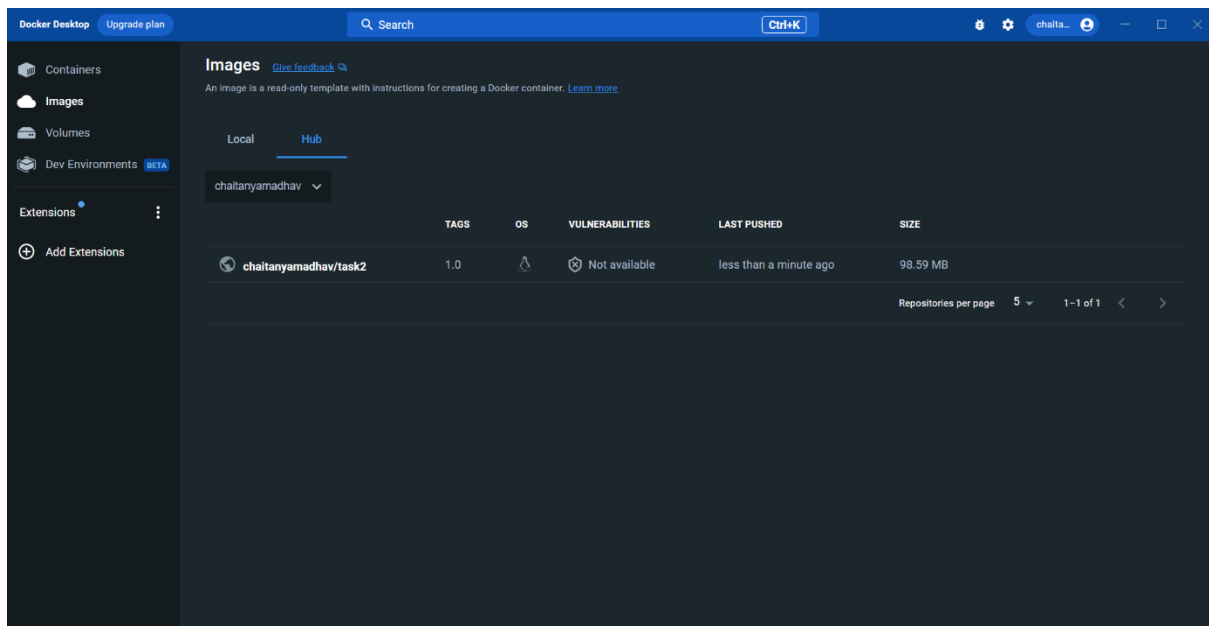
```
C:\Users\chait\Desktop\PES1UG20CS634>docker pull python
Using default tag: latest
latest: Pulling from library/python
Digest: sha256:4d25a30d2943e5feb9c571b763936c843d7a6f45216f508bfe99741067f4ca06
[+] Building 40.4s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 31B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/ubuntu:18.04
=> [auth] library/ubuntu:pull token for registry-1.docker.io
=> CACHED [1/5] FROM docker.io/library/ubuntu:18.04@sha256:a3765b4d74747b5e9bdd03205b3fbc4fa19a02781c185f97f24c8f4f84ed7bbf
=> [internal] load build context
=> => transferring context: 30B
=> [2/5] RUN apt-get update
=> [3/5] RUN apt-get install gcc -y
=> [4/5] COPY program.c program.c
=> [5/5] RUN gcc program.c
=> exporting to image
=> => exporting layers
=> => writing image sha256:18b45f5b5619c477dbb383093c8be3a973067031f42f9339400758b044e89822
=> => naming to docker.io/library/task2

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

C:\Users\chait\Desktop\PES1UG20CS634>docker run task2
Running this inside a container !
My SRN is <PES1UG20CS634>

C:\Users\chait\Desktop\PES1UG20CS634>
```

2b.



```
C:\Users\chait\Desktop\PES1UG20CS634>docker login -u chaitanyamadhav -p RCM@108108 docker.io
WARNING! Using --password via the CLI is insecure. Use --password-stdin.
Login Succeeded

C:\Users\chait\Desktop\PES1UG20CS634>docker tag task2 chaitanyamadhav/task2:1.0

C:\Users\chait\Desktop\PES1UG20CS634>docker push chaitanyamadhav/task2:1.0
The push refers to repository [docker.io/chaitanyamadhav/task2]
45a4b9cd9f23: Pushed
26c874ceda0b: Pushed
815823fc00c3: Pushed
f8a825114f60: Pushed
475a54c2a93d: Mounted from library/ubuntu
1.0: digest: sha256:05b2025b8b0242c91520295ae5ac32fbc97ceda30d07c4d21452eb0db9023dfd size: 1368

C:\Users\chait\Desktop\PES1UG20CS634>
```

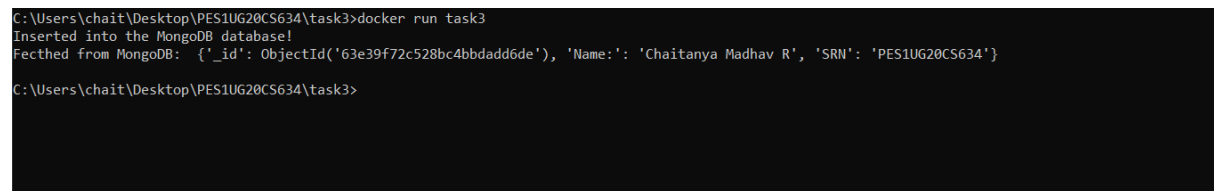
3a.

```
C:\Users\chait\Desktop\PES1UG20CS634\task3>docker run -p 80:80 task3
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2023/02/08 12:57:11 [notice] 181: using the "epoll" event method
2023/02/08 12:57:11 [notice] 181: nginx/1.23.3
2023/02/08 12:57:11 [notice] 181: built by gcc 10.2.1 20210110 (Debian 10.2.1-6)
2023/02/08 12:57:11 [notice] 181: OS: Linux 5.10.16-3-microsoft-standard-WSL2
2023/02/08 12:57:11 [notice] 181: getrlimit(RLIMIT_NFILE): 1048576:1048576
2023/02/08 12:57:11 [notice] 181: start worker processes
2023/02/08 12:57:11 [notice] 181: start worker process 29
2023/02/08 12:57:11 [notice] 181: start worker process 30
2023/02/08 12:57:11 [notice] 181: start worker process 31
2023/02/08 12:57:11 [notice] 181: start worker process 32
2023/02/08 12:57:11 [notice] 181: start worker process 33
2023/02/08 12:57:11 [notice] 181: start worker process 34
2023/02/08 12:57:11 [notice] 181: start worker process 35
2023/02/08 12:57:11 [notice] 181: start worker process 36
172.17.0.1 - - [08/Feb/2023:12:57:25 +0000] "GET /my.html HTTP/1.1" 200 114 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36 Edg/107.0.1418.56"
2023/02/08 12:57:25 [error] 30830: *2 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 172.17.0.1, server: localhost, request: "GET /favicon.ico HTTP/1.1", host: "localhost", referer: "http://localhost/my.html"
172.17.0.1 - - [08/Feb/2023:12:57:25 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost/my.html" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36 Edg/107.0.1418.56" - -
```

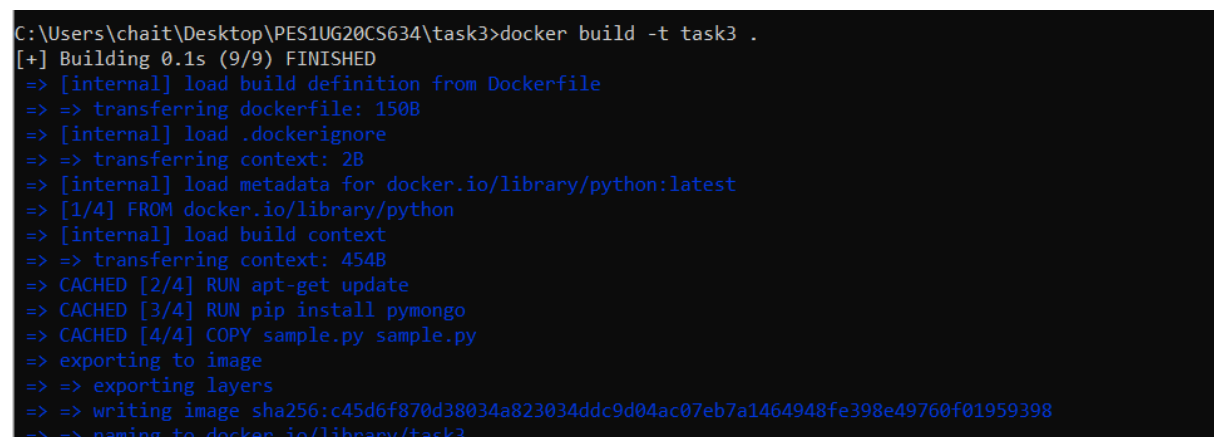
3b.



3c.



3e.



```

[+] Building 0.2s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> transferring dockerfile: 150B
=> [internal] load .dockerignore
=> transferring context: 2B
=> [internal] load metadata for docker.io/library/python:latest
=> [1/4] FROM docker.io/library/python
=> [internal] load build context
=> transferring context: 454B
=> CACHED [2/4] RUN apt-get update
=> CACHED [3/4] RUN pip install pymongo
=> CACHED [4/4] COPY sample.py sample.py
=> exporting to image
=> exporting layers
=> writing image sha256:b9566de3647146733044d6d33b781b48f45c8923d3dbb8537623915
=> naming to docker.io/library/task3-pycode
[+] Building 0.1s (9/9) FINISHED
[+] Running 3/3
  -- Network task3_default Created
  -- Container task3-mongodb-1 Created
  -- Container task3-pycode-1 Created
  Attaching to task3-mongodb-1, task3-pycode-1
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.676+00:00"},"s":"I", "c":"NETWORK", "id":"4915701", "ctx":"","msg":"Initialized wire specification","attr":{"spec":{"IncomingExternalClient":{"wiredireVersio
n":"0","maxWireVersion":"17"},"incomingInternalClient":{"minWireVersion":"0","maxWireVersion":"17"},"outgoing":{"minWireVersion":"6","maxWireVersion":"17"},"isInternalClient":true}}}}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.686+00:00"},"s":"I", "c":"CONTROL", "id":"23285", "ctx":"","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.703+00:00"},"s":"I", "c":"NETWORK", "id":"4648801", "ctx":"main","msg":"Implicit TCP FastOpen unavailable. If TCP FastOpen is required, set tcpFastOpenServer
, tcpFastOpenClient, and tcpFastOpenQueueSize."}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.729+00:00"},"s":"I", "c":"REPL", "id":"5123008", "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"TenantMigrationDonorSe
vice","namespace":"config.tenantMigrationDonors"}}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.729+00:00"},"s":"I", "c":"REPL", "id":"5123008", "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"TenantMigrationRecipie
ntService","namespace":"config.tenantMigrationRecipients"}}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.729+00:00"},"s":"I", "c":"REPL", "id":"5123008", "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"ShardSplitDonorService
","namespace":"config.tenantSplitDonors"}}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.729+00:00"},"s":"I", "c":"CONTROL", "id":"5945603", "ctx":"main","msg":"Multi threading initialized"}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.732+00:00"},"s":"I", "c":"CONTROL", "id":"4615611", "ctx":"main","msg":"MongoDB starting","attr":{"pid":1,"port":27017,"dbPath":"/data/db","architec
ture":"64-bit","host":"69c4fc33f141"}]}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.732+00:00"},"s":"I", "c":"CONTROL", "id":"23403", "ctx":"initandlisten","msg":"Build Info","attr":{"buildInfo":{"version":"6.0.4","gitVersion":"d4ff5946c1f
35368ba172f0f385a566cd2f547","openSSLVersion":"OpenSSL 3.0.2 15 Mar 2022","modules":[""],"allocator":"tcmalloc","environment":{"distmod":"Ubuntu2204","distarch":"x86_64","target_arch":"x86_64"}}}}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.732+00:00"},"s":"I", "c":"CONTROL", "id":"51765", "ctx":"initandlisten","msg":"Operating System","attr":{"os":{"name":"Ubuntu","version":"22.04"}}}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.732+00:00"},"s":"I", "c":"CONTROL", "id":"219151", "ctx":"initandlisten","msg":"Options set by command line","attr":{"options":{"net":{"bindIp":"*"}}}}]
task3-mongodb-1 | [{"t":{"$date":"2023-02-08T12:21:28.740+00:00"},"s":"I", "c":"STORAGE", "id":"22297", "ctx":"initandlisten","msg":"Using the XFS filesystem is strongly recommended with the WiredTiger storage
[+] Building 0.2s (9/9) FINISHED
=> [internal] load build definition from Dockerfile

```

[illegible]

