

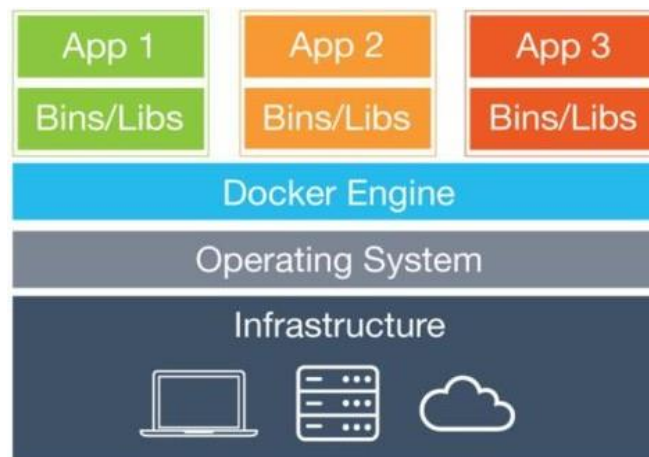
Experiment 4

Aim: Installation and configuration of Docker

Theory:

Docker is a set of platform as a service (PaaS) products that use OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries and configuration files; they can communicate with each other through well-defined channels. Because all of the containers share the services of a single operating system kernel, they use fewer resources than virtual machines.

Containers isolate applications' execution environments from one another, but share the underlying OS kernel. They're typically measured in megabytes, use far fewer resources than VMs, and start up almost immediately. They can be packed far more densely on the same hardware and spun up and down en masse with far less effort and overhead. Containers provide a highly efficient and highly granular mechanism for combining software components into the kinds of application and service stacks needed in a modern enterprise, and for keeping those software components updated and maintained.



Installing Docker:

1. Update your existing list of packages:

```
$ sudo apt update
```

2. Install a few prerequisite packages which let apt use packages over HTTPS:

```
$ sudo apt install apt-transport-https ca-certificates curl  
software-properties-common
```

3. Add the GPG key for the official Docker repository to your system:

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

4. Add the Docker repository to APT sources:

```
$ sudo add-apt-repository "deb [arch=amd64]  
https://download.docker.com/linux/ubuntu bionic stable"
```

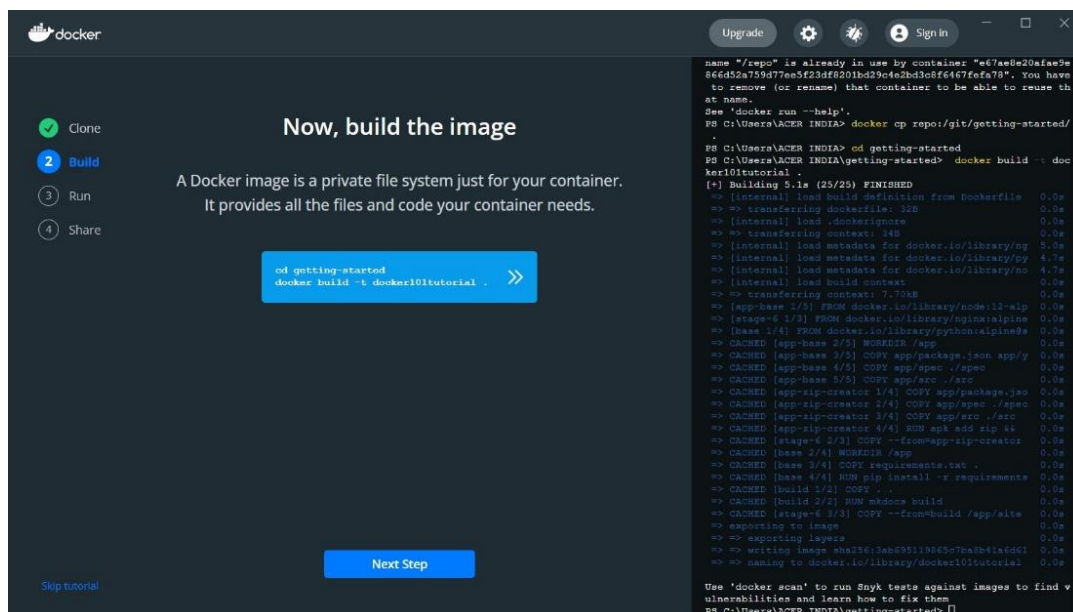
- ```
$ sudo apt update
```

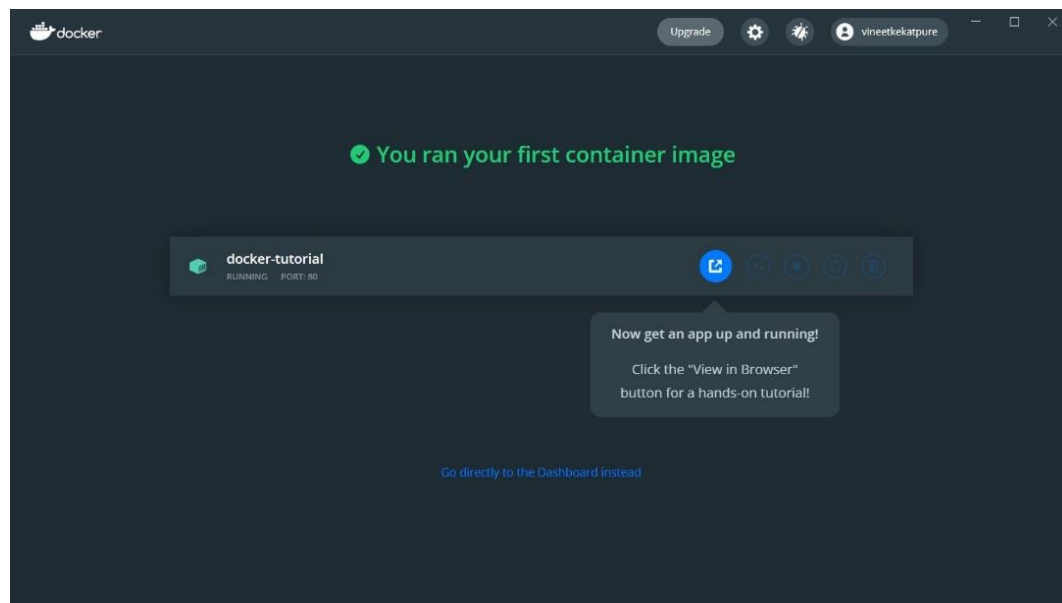
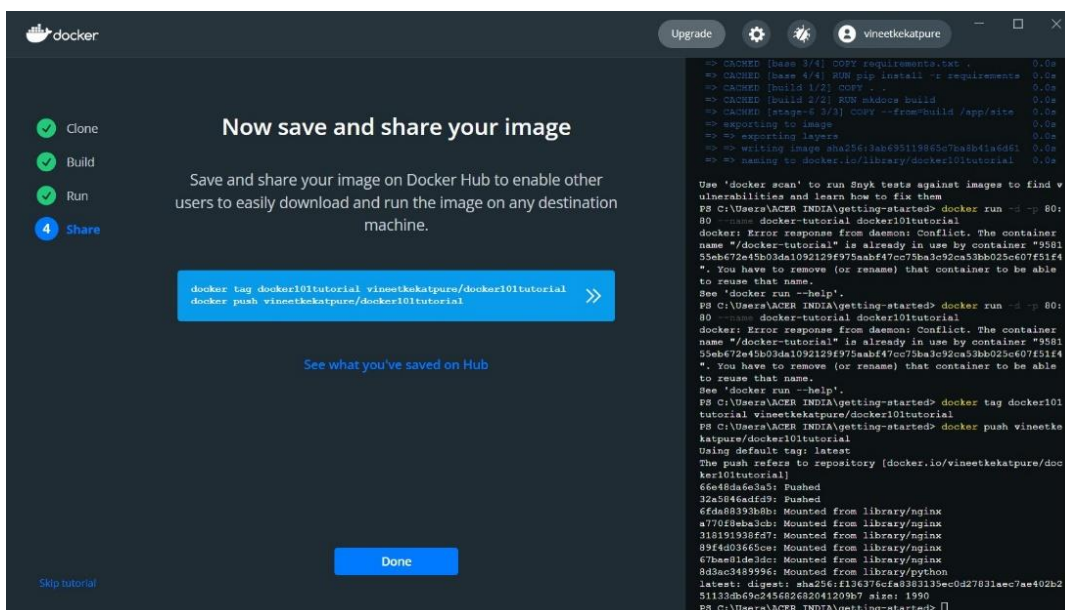
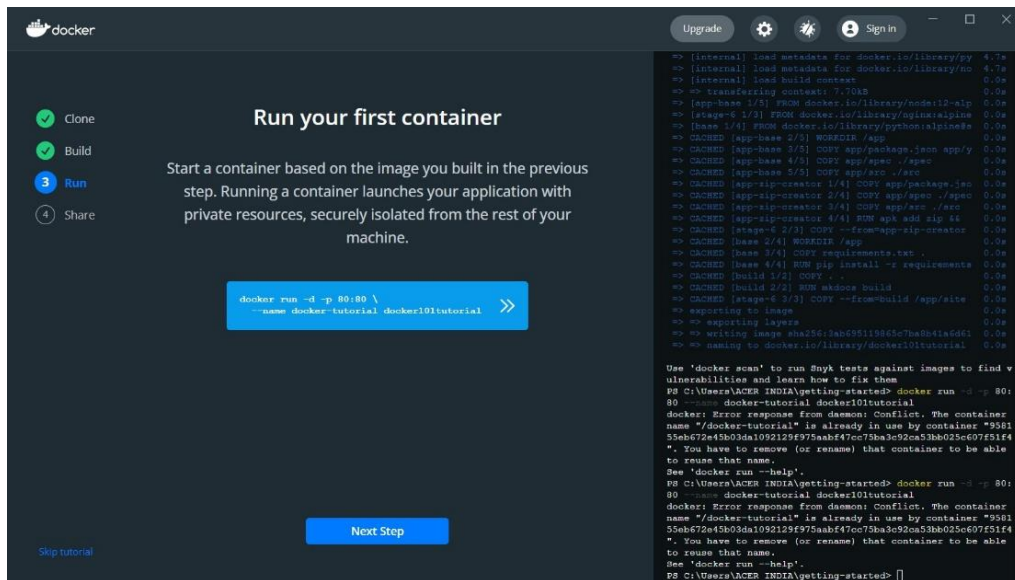
- ```
$ apt-cache policy docker-ce

docker-ce:
  Installed: (none)
  Candidate: 18.03.1~ce~3-0~ubuntu
  Version table:
     18.03.1~ce~3-0~ubuntu 500
        500 https://download.docker.com/linux/ubuntu bionic/stable amd64
Packages
```

- ```
$ sudo apt install docker-ce
```

## Using Docker:





C:\Users\ACER INDIA>docker --version  
Docker version 20.10.12, build e91ed57

This command shows the version of docker installed

```
C:\Users\ACER INDIA>docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to
https://hub.docker.com to create one.
Username: vineetkekatpure
Password:
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/access-tokens/
```

This command logs you into docker

```
C:\Users\ACER INDIA>docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
08c01a0ec47e: Pull complete
Digest: sha256:669e010b58baf5beb2836b253c1fd5768333f0d1dbcb834f7c07a4dc93f474be
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

This command pulls images from the central Docker Repository

```
C:\Users\ACER INDIA>docker images
```

| REPOSITORY                        | TAG    | IMAGE ID     | CREATED        | SIZE   |
|-----------------------------------|--------|--------------|----------------|--------|
| docker101tutorial                 | latest | 3ab695119865 | 21 minutes ago | 28.8MB |
| vineetkekatpure/docker101tutorial | latest | 3ab695119865 | 21 minutes ago | 28.8MB |
| ubuntu                            | latest | 54c9d81cbb44 | 2 weeks ago    | 72.8MB |
| alpine/git                        | latest | c6b70534b534 | 3 months ago   | 27.4MB |

This command shows all the Docker images downloaded on the user's system.

```
C:\Users\ACER INDIA>docker run -it -d ubuntu
9cfee894116275eaca4b84e57977821f9d2ec9528fa6fc7b66e79aadbe09d8ea
```

This command runs containers from their image name.

```
C:\Users\ACER INDIA>docker ps
```

| CONTAINER ID | IMAGE             | COMMAND                 | CREATED        | STATUS        | PORTS              | NAMES           |
|--------------|-------------------|-------------------------|----------------|---------------|--------------------|-----------------|
| 9cfee8941162 | ubuntu            | "bash"                  | 21 seconds ago | Up 20 seconds |                    | funny_solomon   |
| 958155eb672e | docker101tutorial | "/docker-entrypoint..." | 24 minutes ago | Up 24 minutes | 0.0.0.0:80->80/tcp | docker-tutorial |

This command lists all the containers currently running on the system

```
C:\Users\ACER INDIA>docker ps -a
```

| CONTAINER ID | IMAGE             | COMMAND                  | CREATED        | STATUS                    | PORTS              | NAMES           |
|--------------|-------------------|--------------------------|----------------|---------------------------|--------------------|-----------------|
| 9cfee8941162 | ubuntu            | "bash"                   | 42 seconds ago | Up 40 seconds             |                    | funny_solomon   |
| 958155eb672e | docker101tutorial | "/docker-entrypoint..."  | 25 minutes ago | Up 25 minutes             | 0.0.0.0:80->80/tcp | docker-tutorial |
| e67ae8e20afa | alpine/git        | "git clone https://g..." | 29 minutes ago | Exited (0) 29 minutes ago |                    | repo            |

Adding "-a" to the above command shows if there are any stopped containers/

```
C:\Users\ACER INDIA>docker exec -it 9cfee8941162 bash
root@9cfee8941162:/#
```

For logging in a container, "exec" command is used.

```
C:\Users\ACER INDIA>docker stop b110e1c732eb6546a6c1a0425bb99671524b27adc5df7894d899080cb698edc2
b110e1c732eb6546a6c1a0425bb99671524b27adc5df7894d899080cb698edc2
```

This command stops the running container.

```
C:\Users\ACER INDIA>docker kill ba817fba58f35cdf0d8afa2fd4f8a87daebf366d7fd093dc83e8ddf74e40be5e
ba817fba58f35cdf0d8afa2fd4f8a87daebf366d7fd093dc83e8ddf74e40be5e
```

This command kills the container by stopping its execution immediately. The difference between 'docker kill' and 'docker stop'. 'docker stop' gives the container time to shutdown gracefully, in situations when it is taking too much time for getting the container to stop, one can opt to kill it

**Conclusion:** Thus we have successfully installed and configured Docker.