

## 1. Docker install on wsl.

```
root@vishalpc:~# exit
exit
root@vishalpc:/home/vishal# exit
exit
vishal@vishalpc:~$ su -
Password:
root@vishalpc:~# docker --version
Docker version 28.5.2, build ecc6942
root@vishalpc:~# mkdir Docker_tasks
```

- 19.** Run four HTTPD Docker containers with distinct, meaningful names, and apply restart policies (NO, On-Failure, Always, and Unless-Stopped) to each of the four containers, respectively. Demonstrate that the restart policies function as expected.

### Container1 with restart\_no policy

```
root@vishalpc:~/Docker_tasks# docker run -d --name web01_no -p 8080:80 --restart no httpd
Unable to find image 'httpd:latest' locally
latest: Pulling from library/httpd
d7ecded7702a: Already exists
fab98c44430d: Pull complete
4f4fb700ef54: Pull complete
13c22d886563: Pull complete
4870f70a8556: Pull complete
233dec01418c: Pull complete
Digest: sha256:ecfd5ca1bfe1fc5e44a5836c5188bde7f397b50c7a5bb603a017543e29948a01
Status: Downloaded newer image for httpd:latest
1a3f0ba24450afdbc12eb6aa2b571823bf815b1fe91a3f2b0b26bec9022a5acd
root@vishalpc:~/Docker_tasks# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
1a3f0ba24450 httpd "httpd-foreground" 10 seconds ago Up 9 seconds 0.0.0.0:8080->80/tcp, [::]:8080->80/tcp web01_no
root@vishalpc:~/Docker_tasks# docker inspect web01_no
[
    {
        "Id": "1a3f0ba24450afdbc12eb6aa2b571823bf815b1fe91a3f2b0b26bec9022a5acd",
        "Created": "2025-11-06T05:12:28.993742014Z",
        "Path": "httpd-foreground",
        "Args": [],
        "State": {
            "Status": "running",
            "Running": true,
            "Paused": false,
            "Restarting": false,
            "OOMKilled": false,
            "Dead": false,
            "Pid": 10888,
            "ExitCode": 0,
            "Error": ""
        }
    }
]
```

### Container1 with restart onfailuer policy

```
        "Config": {},
    },
    "NetworkMode": "bridge",
    "PortBindings": {
        "80/tcp": [
            {
                "HostIp": "",
                "HostPort": "8081"
            }
        ]
    },
    "RestartPolicy": {
        "Name": "on-failure",
        "MaximumRetryCount": 0
    },
    "AutoRemove": false,
    "VolumeDriver": "",
    "VolumesFrom": null,
    "ConsoleSize": [
        41,
        156
    ],
    "Labels": {
        "com.docker.compose.project": "myapp",
        "com.docker.compose.service": "web"
    }
}
```

```

root@vishalpc:~/Docker_tasks#
root@vishalpc:~/Docker_tasks#
root@vishalpc:~/Docker_tasks# docker run -d --name appserver_onfailure -p 8081:80 --restart on-failure httpd
f40f37fdae7904686d1391f64e4a90d6c761337fe16137893e86d455728faef8
root@vishalpc:~/Docker_tasks# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
f40f37fdae79 httpd "httpd-foreground" 5 seconds ago Up 4 seconds 0.0.0.0:8081->80/tcp, [::]:8081->80/tcp appserver_onfailure
1a3f0ba24450 httpd "httpd-foreground" 4 minutes ago Up 4 minutes 0.0.0.0:8080->80/tcp, [::]:8080->80/tcp web01_no
root@vishalpc:~/Docker_tasks# docker inspect appserver_onfailure
[
  {
    "Id": "f40f37fdae7904686d1391f64e4a90d6c761337fe16137893e86d455728faef8",
    "Created": "2025-11-06T05:16:49.760682656Z",
    "Path": "httpd-foreground",
    "Args": [],
    "State": {
      "Status": "running",
      "Running": true,
      "Paused": false,
      "Restarting": false,
      "OOMKilled": false,
      "Dead": false,
      "Pid": 11222
    }
  }
]

```

```

root@vishalpc:~/Docker_tasks# docker run -d --name db_always -p 8083:80 --restart always httpd
920b3bc4ed7e0d908b8cde97092aec923c93dc9a75f4896193502bd860f5
root@vishalpc:~/Docker_tasks# docker run -d --name local_unless_stopped -p 8084:80 --restart unless-stopped httpd
a3919ab44ddeda9629014cef3b42ab9d7a865703216cc6a7b3f8e477bdbee193
root@vishalpc:~/Docker_tasks# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
a3919ab44dde httpd "httpd-foreground" 6 seconds ago Up 5 seconds 0.0.0.0:8084->80/tcp, [::]:8084->80/tcp local_unless-stopped
920b3bc4ed7 httpd "httpd-foreground" 50 seconds ago Up 50 seconds 0.0.0.0:8083->80/tcp, [::]:8083->80/tcp db_always
f40f37fdae79 httpd "httpd-foreground" 9 minutes ago Up 9 minutes 0.0.0.0:8081->80/tcp, [::]:8081->80/tcp appserver_onfailure
1a3f0ba24450 httpd "httpd-foreground" 13 minutes ago Up 13 minutes 0.0.0.0:8080->80/tcp, [::]:8080->80/tcp web01_no
root@vishalpc:~/Docker_tasks#
root@vishalpc:~/Docker_tasks#
root@vishalpc:~/Docker_tasks#

```

## 20. Change the restart policy of above running container from the default to a custom policy using the docker update command. e.g. docker update --help

```

root@vishalpc:~/Docker_tasks# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
a3919ab44dde httpd "httpd-foreground" 8 minutes ago Up 8 minutes 0.0.0.0:8084->80/tcp, [::]:8084->80/tcp local_unless-stopped
920b3bc4ed7 httpd "httpd-foreground" 8 minutes ago Up 8 minutes 0.0.0.0:8083->80/tcp, [::]:8083->80/tcp db_always
f40f37fdae79 httpd "httpd-foreground" 17 minutes ago Up 17 minutes 0.0.0.0:8081->80/tcp, [::]:8081->80/tcp appserver_onfailure
1a3f0ba24450 httpd "httpd-foreground" 21 minutes ago Up 21 minutes 0.0.0.0:8080->80/tcp, [::]:8080->80/tcp web01_no
root@vishalpc:~/Docker_tasks# docker update --restart=always web01_no
web01_no
root@vishalpc:~/Docker_tasks# docker inspect web01_no
[
  {
    "Id": "1a3f0ba24450afdbc12eb6aa2b571823bf815b1fe91a3f2b0b26bec9022a5acd",
    "Created": "2025-11-06T05:12:28.993742014Z",
    "Path": "httpd-foreground",
    "Args": [],
    "State": {
      "Status": "running",
      "Running": true,
      "Paused": false,
      "Restarting": false,
      "OOMKilled": false,
      "Dead": false,
      "Pid": 10888,
      "ExitCode": 0,
      "Error": "",
      "StartedAt": "2025-11-06T05:12:29.289909647Z",
      "FinishedAt": "0001-01-01T00:00:00Z"
    },
    "Image": "sha256:6a4fe18d08d26a61b32d6aa5d17b2417c4f4de63e8460abf4cd824b911b9de88",
    "ResolvConfPath": "/var/lib/docker/containers/1a3f0ba24450afdbc12eb6aa2b571823bf815b1fe91a3f2b0b26bec9022a5acd/resolv.conf",
    "HostnamePath": "/var/lib/docker/containers/1a3f0ba24450afdbc12eb6aa2b571823bf815b1fe91a3f2b0b26bec9022a5acd/hostname",
    "HostsPath": "/var/lib/docker/containers/1a3f0ba24450afdbc12eb6aa2b571823bf815b1fe91a3f2b0b26bec9022a5acd/hosts",
    "LogPath": "/var/lib/docker/containers/1a3f0ba24450afdbc12eb6aa2b571823bf815b1fe91a3f2b0b26bec9022a5acd/1a3f0ba24450afdbc12eb6aa2b571823bf815b1fe91a3f2b0b26bec9022a5acd-log",
    "Name": "/web01_no",
    "RestartCount": 0,
    "Driver": "overlay2",
    "Platform": "linux",
    "MountLabel": ""
  }
]

```

```

"AppArmorProfile": null,
"ExecIDs": null,
"HostConfig": {
    "Binds": null,
    "ContainerIDFile": "",
    "LogConfig": {
        "Type": "json-file",
        "Config": {}
    },
    "NetworkMode": "bridge",
    "PortBindings": {
        "80/tcp": [
            {
                "HostIp": "",
                "HostPort": "8080"
            }
        ]
    },
    "RestartPolicy": {
        "Name": "always",
        "MaximumRetryCount": 0
    },
    "AutoRemove": false,
    "VolumeDriver": "",
    "VolumesFrom": null,
    "ConsoleSize": [
        41,
        156
    ],
    "CapAdd": null,
    "CapDrop": null,
    "CgroupnsMode": "private",
    "Dns": [],
    "DnsOptions": [],
    "DnsSearch": [],
    "ExtraHosts": null,
    "GroupAdd": null,
    "IpcMode": "private",
    "Cgroup": "",
    "Links": null
}

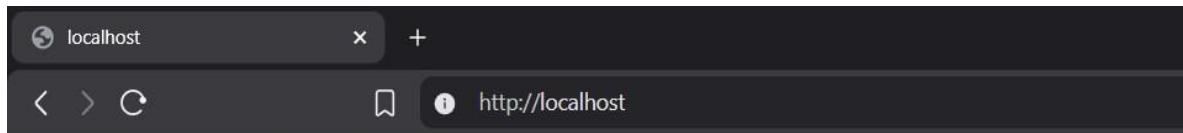
```

- 21.** Launch an NGINX container with a meaningful name and expose it on the host's port 80. Create an "index.html" file containing the text "Hello there, Let's be the Team CloudEthiX," and copy the file to the container's "/usr/share/nginx/html/" location. Access the container in a browser to verify that the webpage displays correctly.

```

webserver
root@vishalpc:~/Docker_tasks$ docker container run -dit --name webserver -p 80:80 nginx
332a8c5dbfc9b84ae1eaf3a779654256d153c3acd0f1ec6b61d6e2d08d4ac55b
root@vishalpc:~/Docker_tasks$ docker exec -it webserver /bin/bash
root@332a8c5dbfc9:/# echo "Hello there, Let's be the Team CloudEthiX" > index.html
root@332a8c5dbfc9:/# mv index.html /usr/share/nginx/html/
root@332a8c5dbfc9:/# cd /usr/share/nginx/html/
root@332a8c5dbfc9:/usr/share/nginx/html# ls
50x.html index.html
root@332a8c5dbfc9:/usr/share/nginx/html#
root@332a8c5dbfc9:/usr/share/nginx/html#
root@332a8c5dbfc9:/usr/share/nginx/html# |

```



- 22.** Run a docker container with CPU and Memory limit. docker container run --help

```
root@vishalpc:~/Docker_tasks# docker run -d -m 512m --memory-reservation=256m --cpus=2 --name newaap nginx
31357881496ee997dd111ef9bfb0a64ad804061341b37562027621242bc0dc19
root@vishalpc:~/Docker_tasks# docker inspect memcont1 | grep -i memory
    "Memory": 536870912,
    "MemoryReservation": 268435456,
    "MemorySwap": 1073741824,
    "MemorySwappiness": null,
root@vishalpc:~/Docker_tasks# docker inspect memcont1 | grep -i cpu
    "CpuShares": 0,
    "NanoCpus": 0,
    "CpuPeriod": 0,
    "CpuQuota": 0,
    "CpuRealtimePeriod": 0,
    "CpuRealtimeRuntime": 0,
    "CpusetCpus": "",
    "CpusetMems": "",
    "CpuCount": 0,
    "CpuPercent": 0,
```

## 23. Update CUP and Memory of docker container using docker update.

ref links :-

```
    "CpuCount": 0,
    "CpuPercent": 0,
root@vishalpc:~/Docker_tasks# docker update -m 1024m --cpus=1 newaap
newaap
root@vishalpc:~/Docker_tasks# docker inspect newaap | grep -i cpu
    "CpuShares": 0,
    "NanoCpus": 1000000000,
    "CpuPeriod": 0,
    "CpuQuota": 0,
    "CpuRealtimePeriod": 0,
    "CpuRealtimeRuntime": 0,
    "CpusetCpus": "",
    "CpusetMems": "",
    "CpuCount": 0,
    "CpuPercent": 0,
root@vishalpc:~/Docker_tasks# docker inspect newaap | grep -i memory
    "Memory": 1073741824,
    "MemoryReservation": 268435456,
    "MemorySwap": 1073741824,
    "MemorySwappiness": null,
root@vishalpc:~/Docker_tasks# |
```

## 24. Pull the Busy-box image to your local system, tag it, and push it to the Docker Hub repository "yourname\_cloudeithix\_busybox."

```

root@vishalpc:~/Docker_tasks# docker pull busybox
Using default tag: latest
latest: Pulling from library/busybox
e59838ecfec5: Pull complete
Digest: sha256:e3652a00a2fabd16ce889f0aa32c38eec347b997e73bd09e69c962ec7f8732ee
Status: Downloaded newer image for busybox:latest
docker.io/library/busybox:latest
root@vishalpc:~/Docker_tasks# docker image ls
REPOSITORY TAG IMAGE ID CREATED SIZE
httpd latest 6a4fe18d08d2 2 days ago 117MB
nginx latest d261fd19cb63 2 days ago 152MB
busybox latest 08ef35a1c3f0 13 months ago 4.43MB
root@vishalpc:~/Docker_tasks# docker tag busybox:latest vishalbobhate/cloudethix:v1
root@vishalpc:~/Docker_tasks# docker login -u vishalbobhate

Info → A Personal Access Token (PAT) can be used instead.
To create a PAT, visit https://app.docker.com/settings

Password:

WARNING! Your credentials are stored unencrypted in '/root/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
root@vishalpc:~/Docker_tasks# docker push vishalbobhate/cloudethix:v1
The push refers to repository [docker.io/vishalbobhate/cloudethix]
e14542cc0629: Mounted from library/busybox
v1: digest: sha256:755d9ce09782b2f60fd3321878e611f871817aa8a726e8c605f2f67ae6ffa9e2 size: 527
root@vishalpc:~/Docker_tasks#

```

Allow important notifications from Docker. You will get important notifications from Docker. Tool when do not select this app in settings. Allow

Repositories / cloudethix / Tags

vishalbobhate/cloudethix

Last pushed 1 minute ago · Repository size: 2.1 MB · ⭐0 · ⚡0

work area

Add a category

General Tags Image Management ⚡ Collaborators Webhooks Settings

Sort by Newest Filter tags Delete

TAG	Digest	OS/ARCH	Last pull	Compressed size
v1	755d9ce09782	linux/amd64	less than 1 day	2.11 MB

Using 1 of 1 private repositories.

Docker commands

To push a new tag to this repository:

`docker push vishalbobhate/cloudethix:v1`

25. In your project directory, create a Dockerfile (named Dockerfile) with the following content.

FROM nginx:latest

COPY custom-index.html /usr/share/nginx/html/index.html

Ensure you also have a file named custom-index.html in the same directory. Build the Docker image using the Dockerfile you created and push it to your repository. Delete the local image. Start a new

container using the custom Nginx image that you just pushed. Map port 8080 on your host to port 80 in the container. Check the container page in browser.

```
root@vishalpc:~/Docker_tasks# docker run -d --name web_custom_nginx vishalbobhate/cloudeithix:latest
Unable to find image 'vishalbobhate/cloudeithix:latest' locally
latest: Pulling from vishalbobhate/cloudeithix
d7ecded7702a: Already exists
266626526d42: Already exists
320b0949be89: Already exists
d921c57c6a81: Already exists
9def903993e4: Already exists
52bc359bcd7: Already exists
e2f8e296d9df: Already exists
c9e2adf41f1f: Already exists
Digest: sha256:7ba86a332bd8c8b06d4856cedaa225d293b6ed77c7a286587e6a2518284b35aa
Status: Downloaded newer image for vishalbobhate/cloudeithix:latest
95ab2d57654f759dbf54c928d269bb62f760b712c2bf1066df124d94c32a03bc
root@vishalpc:~/Docker_tasks# docker ps
CONTAINER ID   IMAGE                  COMMAND       CREATED      STATUS      PORTS     NAMES
95ab2d57654f   vishalbobhate/cloudeithix:latest   "/docker-entrypoint..."   22 seconds ago   Up 20 seconds   80/tcp    web_custom_nginx
```



**26.** Push the Redis images tagged as version 1 and 3 to your Docker Hub repository, named "yourname\_cloudeithix\_redis."

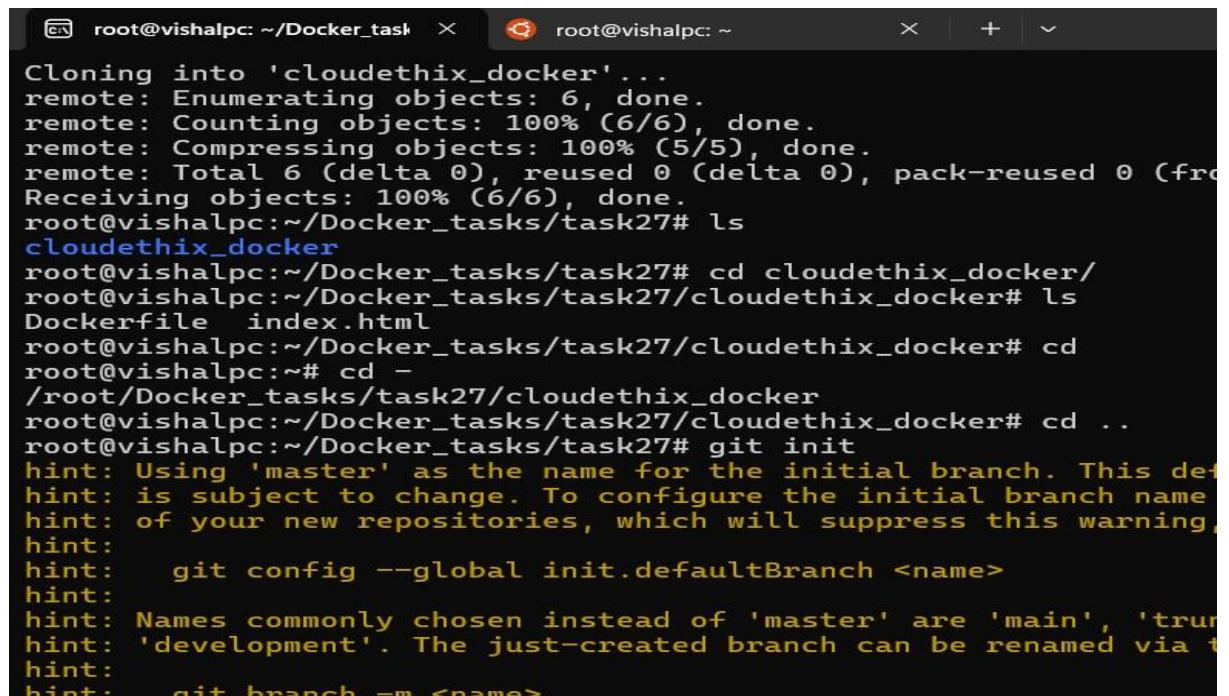
```
root@vishalpc:~/Docker_tasks# docker run -d --name web_custom_nginx vishalbobhate/cloudeithix:latest
Unable to find image 'vishalbobhate/cloudeithix:latest' locally
latest: Pulling from vishalbobhate/cloudeithix
d7ecded7702a: Already exists
266626526d42: Already exists
320b0949be89: Already exists
d921c57c6a81: Already exists
9def903993e4: Already exists
52bc359bcd7: Already exists
e2f8e296d9df: Already exists
c9e2adf41f1f: Already exists
Digest: sha256:7ba86a332bd8c8b06d4856cedaa225d293b6ed77c7a286587e6a2518284b35aa
Status: Downloaded newer image for vishalbobhate/cloudeithix:latest
95ab2d57654f759dbf54c928d269bb62f760b712c2bf1066df124d94c32a03bc
root@vishalpc:~/Docker_tasks# docker ps
CONTAINER ID   IMAGE                  COMMAND       CREATED      STATUS      PORTS     NAMES
95ab2d57654f   vishalbobhate/cloudeithix:latest   "/docker-entrypoint..."   22 seconds ago   Up 20 seconds   80/tcp    web_custom_nginx
root@vishalpc:~/Docker_tasks#
root@vishalpc:~/Docker_tasks# docker pull redis
Using default tag: latest
latest: Pulling from library/redis
1adaabd6b0d6b: Pull complete
22506777a996: Pull complete
5dec27664782: Pull complete
2e60b18d70d8: Pull complete
729797e636a7: Pull complete
4f4fb700ef54: Pull complete
oc981fc7c621: Pull complete
Digest: sha256:5c7c0445ed86918cb9efb96d95a6bf0c03ed2059fe2c5f02b4d74f477ffe47915
Status: Downloaded newer image for redis:latest
docker.io/library/redis:latest
root@vishalpc:~/Docker_tasks# docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
vishalbobhate/cloudeithix  latest   2b7f61ec9a0d  21 minutes ago  152MB
httpd               latest   6a4fe18d08d2  2 days ago   117MB
nginx               latest   d261fd19cb63  2 days ago   152MB
redis               latest   017b1c12abf9  2 days ago   137MB
busybox              latest   08ef35a1c3f0  13 months ago  4.43MB
vishalbobhate/cloudeithix  v1      08ef35a1c3f0  13 months ago  4.43MB
root@vishalpc:~/Docker_tasks# docker push vishalbobhate/vishal_cloudeithix_redis:version1
The push refers to repository [docker.io/vishalbobhate/vishal_cloudeithix_redis]
An image does not exist locally with the tag: vishalbobhate/vishal_cloudeithix_redis
root@vishalpc:~/Docker_tasks# docker redis:latest vishalbobhate/vishal_cloudeithix_redis:version1
```

```

root@vishalpc:~/Docker_tasks$ docker run -d --name web_custom_nginx vishalbobhate/cloudeithix:latest
Unable to find image 'vishalbobhate/cloudeithix:latest' locally
Latest: Pulling from vishalbobhate/cloudeithix
d7ecded7702a: Already exists
266626526d42: Already exists
329b0949be89: Already exists
d921c57c6a81: Already exists
9def983993e4: Already exists
52bc359bcd7: Already exists
e2f8e296d9df: Already exists
c9e2adff41f1f: Already exists
Digest: sha256:7ba86a32bd8c8b06d4856cedaa225d293b6ed77c7a286587e6a2518284b35aa
Status: Downloaded newer image for vishalbobhate/cloudeithix:latest
95ab2d57654f759dbf54c928d269bb62f760b712c2bf1066df124d94c32a03bc
root@vishalpc:~/Docker_tasks$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS     NAMES
95ab2d57654f        vishalbobhate/cloudeithix:latest   "/docker-entrypoint..."   22 seconds ago    Up 20 seconds   80/tcp   web_custom_nginx

```

27. Create a remote Git repository and add the Dockerfile and index.html files. Clone the repository locally and create a release branch. Build the Docker image from the release branch with a meaningful tag, then run a container from the image and expose it on host port 8383. Check the webpage in a browser, and upon success, push the image to your Docker Hub repository named "yourusername\_cloudeithix\_nginx." Finally, push the release branch to the remote Git repository and merge it by creating a pull request.



```

Cloning into 'cloudeithix_docker'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from...)
Receiving objects: 100% (6/6), done.
root@vishalpc:~/Docker_tasks/task27# ls
cloudeithix_docker
root@vishalpc:~/Docker_tasks/task27# cd cloudeithix_docker/
root@vishalpc:~/Docker_tasks/task27/cloudeithix_docker# ls
Dockerfile index.html
root@vishalpc:~/Docker_tasks/task27/cloudeithix_docker# cd ..
root@vishalpc:~# cd -
/root/Docker_tasks/task27/cloudeithix_docker
root@vishalpc:~/Docker_tasks/task27/cloudeithix_docker# cd ..
root@vishalpc:~/Docker_tasks/task27# git init
hint: Using 'master' as the name for the initial branch. This default
hint: is subject to change. To configure the initial branch name
hint: of your new repositories, which will suppress this warning,
hint:
hint:     git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk'
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:     git branch -m <name>

```

```

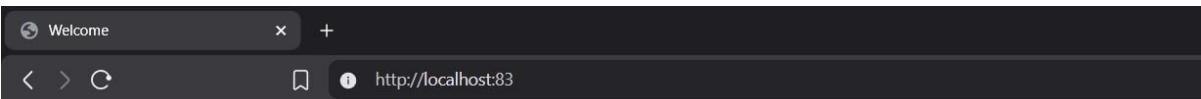
/root/Docker_tasks/task27/cloudeithix_docker
root@vishalpc:~/Docker_tasks/task27/cloudeithix_docker# cd ..
root@vishalpc:~/Docker_tasks/task27# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:     git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:     git branch -m <name>
Initialized empty Git repository in /root/Docker_tasks/task27/.git/
root@vishalpc:~/Docker_tasks/task27# git add

```

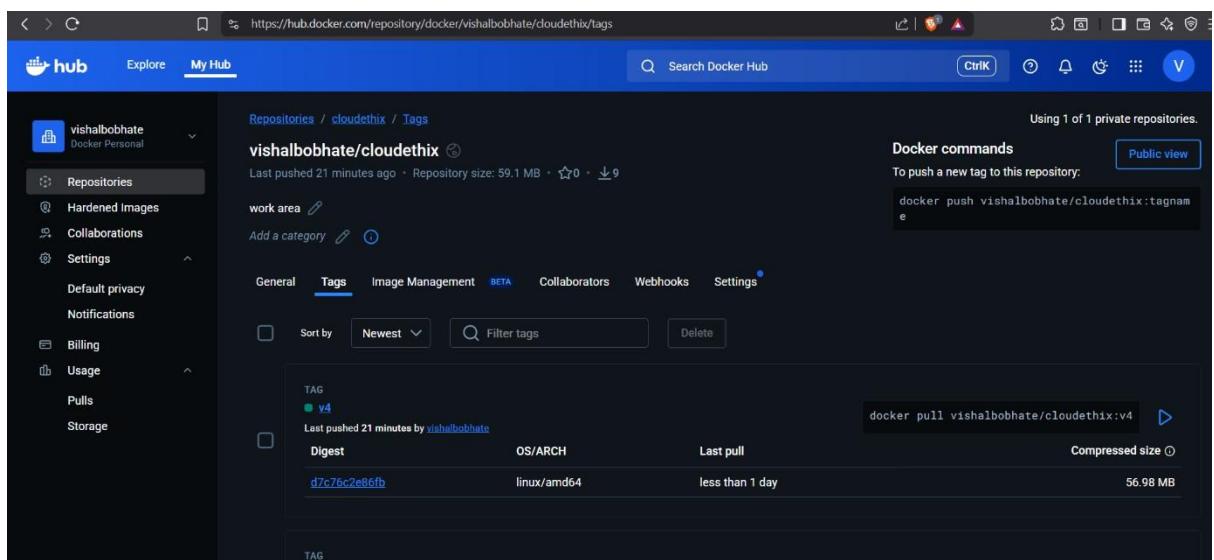
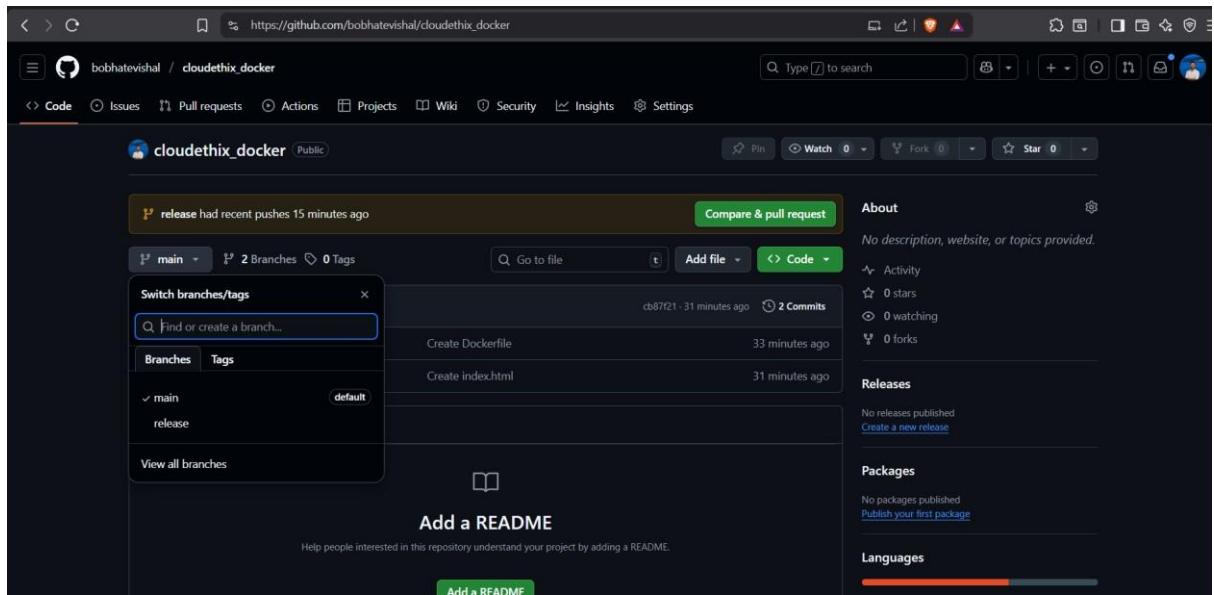
```

root@vishalpc:~/Docker_tasks/task27# git branch release
root@vishalpc:~/Docker_tasks/task27# git branch
* master
  release
root@vishalpc:~/Docker_tasks/task27# git checkout release
Switched to branch 'release'
root@vishalpc:~/Docker_tasks/task27# ls
cloudehix_docker
root@vishalpc:~/Docker_tasks/task27# cd cloudehix_docker/
root@vishalpc:~/Docker_tasks/task27/cloudehix_docker# ls
Dockerfile index.html
root@vishalpc:~/Docker_tasks/task27/cloudehix_docker# docker build -t mydocker_img
=>=> naming to docker.io/library/mydocker_img
root@vishalpc:~/Docker_tasks/task27/cloudehix_docker# docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
mydocker_img         latest   b5768552ea59  12 seconds ago  152MB
vishalbobhate/cloudehix
httpd               latest   2b7f61ec9a0d  2 hours ago   152MB
nginx               latest   d261fd19cb63  2 days ago    117MB
redis               latest   017b1c12abf9  2 days ago    152MB
vishalbobhate/vishal_cloudehix_redis version1 017b1c12abf9  2 days ago    137MB
vishalbobhate/vishal_cloudehix_redis version3 017b1c12abf9  2 days ago    137MB
busybox              latest   08ef35a1c3f0  13 months ago  4.43MB
vishalbobhate/cloudehix           v1      08ef35a1c3f0  13 months ago  4.43MB
root@vishalpc:~/Docker_tasks/task27/cloudehix_docker# docker tag mydocker_img:latest vishalbobhate/cloudehix:v4
root@vishalpc:~/Docker_tasks/task27/cloudehix_docker# docker push vishalbobhate/cloudehix:v4
The push refers to repository [docker.io/vishalbobhate/cloudehix]
dc24ed7e0824: Pushed
d7217c60dca4: Layer already exists
d81df94f8d07: Layer already exists
99cd1b1b6a43: Layer already exists
2ced4cd78a7b: Layer already exists
8feb164cd673: Layer already exists
6e19587ac541: Layer already exists
36d06fe0cbc6: Layer already exists
v4: digest: sha256:d7c76c2e86fbcc700c34c98e0e30a1f5c2cda4cbcaf1b59c53cf98a47c3c38c53 size: 1985
root@vishalpc:~/Docker_tasks/task27/cloudehix_docker# git push origin
Username for 'https://github.com': bobhatevishal
Password for 'https://bobhatevishal@github.com':
Everything up-to-date
root@vishalpc:~/Docker_tasks/task27/cloudehix_docker# git checkout master
error: pathspec 'master' did not match any file(s) known to git
root@vishalpc:~/Docker_tasks/task27/cloudehix_docker# git branch
* master
Enumerating objects: 2, done.
Counting objects: 100% (2/2), done.
Writing objects: 100% (2/2), 206 bytes | 206.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'release' on GitHub by visiting:
remote:     https://github.com/bobhatevishal/cloudehix_docker/pull/new/release
remote:
To https://github.com/bobhatevishal/cloudehix_docker.git
 * [new branch]      release -> release
branch 'release' set up to track 'origin/release'.
root@vishalpc:~/Docker_tasks/task27# docker run -d --name webserver_27 -p 83:83 mydocker_img:latest
30ef20a2784d90e95d8e3260b24fd7b525d4c76cede18d37ff6ccce9eda05ela87
root@vishalpc:~/Docker_tasks/task27# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS                 NAMES
ES
30ef20a2784d        mydocker_img:latest   "/docker-entrypoint..."   44 seconds ago   Up 43 seconds   80/tcp, 0.0.0.0:83->83/tcp, [::]:83->83/tcp   web
server_27
f66e5720e3f2        vishalbobhate/cloudehix:latest   "/docker-entrypoint..."   2 hours ago      Up 2 hours       0.0.0.0:80->80/tcp, [::]:80->80/tcp   web
_custom_nginx
root@vishalpc:~/Docker_tasks/task27# docker run -d --name webserver_27 -p 83:80 mydocker_img:latest
docker: Error response from daemon: Conflict. The container name "webserver_27" is already in use by container "30ef20a2784d90e95d8e3260b24fd7b525d4c76cede18d37ff6ccce9eda05ela87". You have to remove (or rename) that container to be able to reuse that name.
Run 'docker run --help' for more information
root@vishalpc:~/Docker_tasks/task27# docker run -d --name webserver_27 -p 83:80 mydocker_img:latest
c1c07bc3a33343d7a4cdf8af96032afeaca5c71b43c68a942c2a0619703577020f

```



Hello from CloudEthix WebApp



**28.** Save all local Redis images as a .tar file in the master branch of your local repository. Delete all Redis images from your local system and push the master branch to the remote repository. Load the Redis images from the tar file to your local system, and verify that all images have been loaded correctly.

```

root@vishalpc:~/Docker_tasks# ls
Dockerfile custom-index.html task27
root@vishalpc:~/Docker_tasks# docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
mydocker_img        latest   b5768552ea59  33 minutes ago  152MB
vishalbobhate/cloudeithix    v4      b5768552ea59  33 minutes ago  152MB
vishalbobhate/cloudeithix    latest   2b7f61ec9a0d  3 hours ago   152MB
httpd               latest   6a4fe18d08d2  2 days ago   117MB
nginx               latest   d261fd19cb63  2 days ago   152MB
redis               latest   017b1c12abf9  2 days ago   137MB
vishalbobhate/vishal_cloudeithix_redis version1 017b1c12abf9  2 days ago   137MB
vishalbobhate/vishal_cloudeithix_redis version3 017b1c12abf9  2 days ago   137MB
busybox              latest   08ef35a1c3f0  13 months ago  4.43MB
vishalbobhate/cloudeithix    v1      08ef35a1c3f0  13 months ago  4.43MB
root@vishalpc:~/Docker_tasks# docker image ls | grep -i "redis"
vishalbobhate/vishal_cloudeithix_redis version1 017b1c12abf9  2 days ago   137MB
vishalbobhate/vishal_cloudeithix_redis version3 017b1c12abf9  2 days ago   137MB
redis               latest   017b1c12abf9  2 days ago   137MB
root@vishalpc:~/Docker_tasks# git branch
fatal: not a git repository (or any of the parent directories): .git
root@vishalpc:~/Docker_tasks# cd task27
root@vishalpc:~/Docker_tasks/task27# git branch
  master
* release
root@vishalpc:~/Docker_tasks/task27# git checkout master
Switched to branch 'master'

```

```

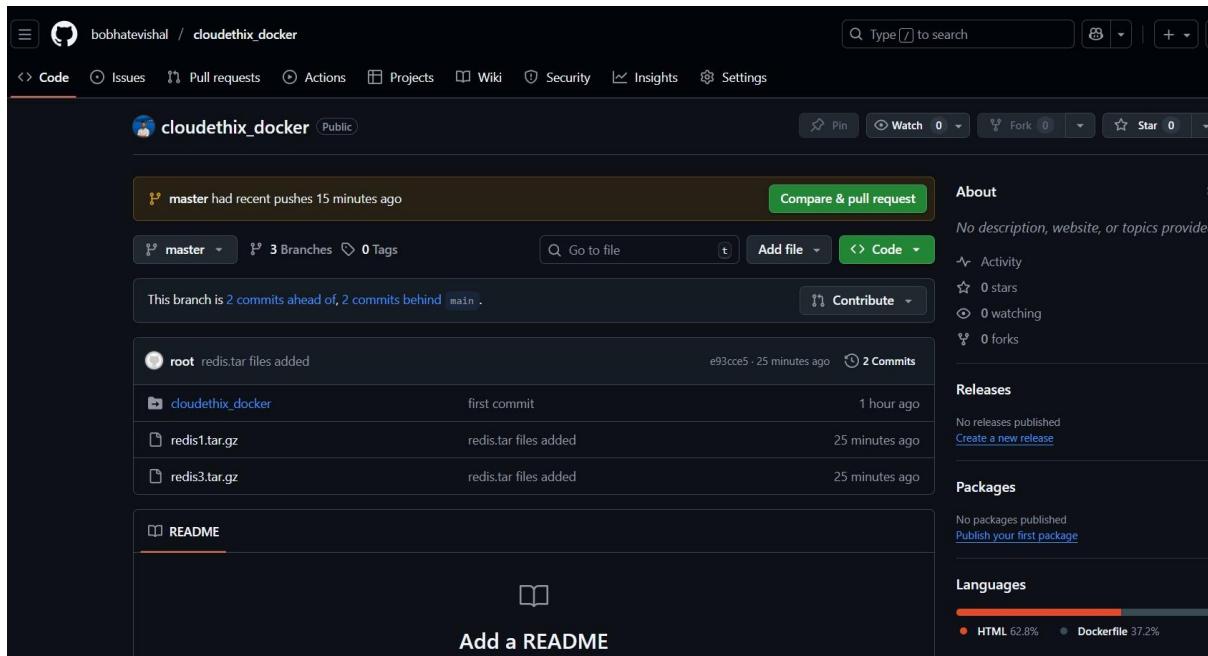
root@vishalpc:~/Docker_tasks/task27# git checkout master
Switched to branch 'master'
root@vishalpc:~/Docker_tasks/task27# docker save -o redis1.tar > visha
-bash: vishalbobhate/vishal_cloudeithix_redis:version1: No such file or
root@vishalpc:~/Docker_tasks/task27# docker save -o redis1.tar vishalb
root@vishalpc:~/Docker_tasks/task27# docker save -o redis3.tar vishalb
root@vishalpc:~/Docker_tasks/task27# ls
cloudeithix_docker  redis1.tar  redis3.tar
root@vishalpc:~/Docker_tasks/task27# git add .
root@vishalpc:~/Docker_tasks/task27# git commit -m "redis.tar files ad
[master f7c515a] redis.tar files added
Committer: root <root@vishalpc.localdomain>

```

```

root@vishalpc:~/Docker_tasks/task27# git checkout master
Switched to branch 'master'
root@vishalpc:~/Docker_tasks/task27# docker save -o redis1.tar > vishalbobhate/vishal_cloudeithix_redis:version1
-bash: vishalbobhate/vishal_cloudeithix_redis:version1: No such file or directory
root@vishalpc:~/Docker_tasks/task27# docker save -o redis1.tar vishalbobhate/vishal_cloudeithix_redis:version1
root@vishalpc:~/Docker_tasks/task27# docker save -o redis3.tar vishalbobhate/vishal_cloudeithix_redis:version3
root@vishalpc:~/Docker_tasks/task27# ls
cloudeithix_docker  redis1.tar  redis3.tar
root@vishalpc:~/Docker_tasks/task27# git add .
root@vishalpc:~/Docker_tasks/task27# git commit -m "redis.tar files added"
[master f7c515a] redis.tar files added
Committer: root <root@vishalpc.localdomain>
root@vishalpc:~/Docker_tasks/task27# gunzip redis1.tar
redis1.tar.gz  redis3.tar.gz
root@vishalpc:~/Docker_tasks/task27# gunzip redis1.tar.gz  redis3.tar.gz
root@vishalpc:~/Docker_tasks/task27# ls
cloudeithix_docker  redis1.tar  redis3.tar
root@vishalpc:~/Docker_tasks/task27# docker load -i redis1.tar
Loaded image: vishalbobhate/vishal_cloudeithix_redis:version1
root@vishalpc:~/Docker_tasks/task27# docker load -i redis3.tar
Loaded image: vishalbobhate/vishal_cloudeithix_redis:version3
root@vishalpc:~/Docker_tasks/task27# docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
mydocker_img        latest   b5768552ea59  About an hour ago  152MB
vishalbobhate/cloudeithix    v4      b5768552ea59  About an hour ago  152MB
vishalbobhate/cloudeithix    latest   2b7f61ec9a0d  3 hours ago   152MB
httpd               latest   6a4fe18d08d2  2 days ago   117MB
nginx               latest   d261fd19cb63  2 days ago   152MB
redis               latest   017b1c12abf9  2 days ago   137MB
vishalbobhate/vishal_cloudeithix_redis version1 017b1c12abf9  2 days ago   137MB
vishalbobhate/vishal_cloudeithix_redis version3 017b1c12abf9  2 days ago   137MB
busybox              latest   08ef35a1c3f0  13 months ago  4.43MB
vishalbobhate/cloudeithix    v1      08ef35a1c3f0  13 months ago  4.43MB
root@vishalpc:~/Docker_tasks/task27# docker image ls | grep -i redis
redis               latest   017b1c12abf9  2 days ago   137MB
vishalbobhate/vishal_cloudeithix_redis version1 017b1c12abf9  2 days ago   137MB
vishalbobhate/vishal_cloudeithix_redis version3 017b1c12abf9  2 days ago   137MB
root@vishalpc:~/Docker_tasks/task27# |

```



- 29.** Pull the Busy-box image to your local system, tag it, and push it to the Docker Hub repository "yourname\_cloudethix\_busybox." Export the Docker image from the NGINX container, create a .tar file, and import the tar file to create a Docker image with a meaningful name. After importing the image, tag it and push it to the "yourname\_cloudethix\_busybox" Docker Hub repository.

```

vishalbobhate/vishal_cloudethix_redis    latest      017b1c12abf9   2 days ago   137MB
vishalbobhate/vishal_cloudethix_redis    version1   017b1c12abf9   2 days ago   137MB
busybox                                latest      08ef35a1c3f0   13 months ago  4.43MB
vishalbobhate/cloudethix                v1         08ef35a1c3f0   13 months ago  4.43MB
vishalbobhate/cloudethix_busybox        v5         08ef35a1c3f0   13 months ago  4.43MB
root@vishalpc:~/Docker_tasks# docker push bobhatevishal/cloudethix_busybox:v5
The push refers to repository [docker.io/bobhatevishal/cloudethix_busybox]
An image does not exist locally with the tag: bobhatevishal/cloudethix_busybox
root@vishalpc:~/Docker_tasks# docker tag busybox:latest vishalbobhate/cloudethix_busybox:v5
root@vishalpc:~/Docker_tasks# docker push bobhatevishal/cloudethix_busybox:v5
The push refers to repository [docker.io/bobhatevishal/cloudethix_busybox]
An image does not exist locally with the tag: bobhatevishal/cloudethix_busybox
root@vishalpc:~/Docker_tasks# docker tag vishalbobhate/cloudethix_busybox:v5 vishalbobhate/cloudethix_busybox:v5
Error response from daemon: No such image: vishalbobhate/cloudethix_busybox:latest
root@vishalpc:~/Docker_tasks# docker tag vishalbobhate/cloudethix_busybox:v5 vishalbobhate/cloudethix_busybox:v5
root@vishalpc:~/Docker_tasks# docker push bobhatevishal/cloudethix_busybox:v5
The push refers to repository [docker.io/bobhatevishal/cloudethix_busybox]
An image does not exist locally with the tag: bobhatevishal/cloudethix_busybox
root@vishalpc:~/Docker_tasks# docker login
Authenticating with existing credentials... [Username: vishalbobhate]

Info : To login with a different account, run 'docker logout' followed by 'docker login'

Login Succeeded

```

```

root@vishalpc:~/Docker_tasks# docker push vishalbobhate/cloudethix_busybox:v5
The push refers to repository [docker.io/vishalbobhate/cloudethix_busybox]
e14542cc0629: Mounted from vishalbobhate/cloudethix
v5: digest: sha256:755d9ce09782b2f60fd3321878e611f871817aa8a726e8c605f2f67ae6ffa9e2 size: 527
root@vishalpc:~/Docker_tasks# docker run -d --name nginx_container nginx
669ae08cb00cb15e8f8a624c3e5ccb9b94722d9b2785cef47ae42e054c17a1c
root@vishalpc:~/Docker_tasks# docker export nginx_container -o nginx_cont.tar
root@vishalpc:~/Docker_tasks# ls
Dockerfile custom-index.html nginx_cont.tar task27
root@vishalpc:~/Docker_tasks# docker import nginx_cont.tar vishalbobhate/cloudethix_busybox:v5
sha256:4d01f95718d4655f87434a653f816860114dd69779a5ef1fa44df44c216849e
root@vishalpc:~/Docker_tasks# docker image ls | grep -i busybox
vishalbobhate/cloudethix_busybox          v5         4d01f95718d4   2 minutes ago  150MB
busybox                                latest      08ef35a1c3f0   13 months ago  4.43MB
vishalbobhate/cloudethix_busybox        <none>     08ef35a1c3f0   13 months ago  4.43MB
root@vishalpc:~/Docker_tasks# docker import nginx_cont.tar vishalbobhate/cloudethix_busybox:imported

```

```

root@vishalpc:~/Docker_tasks# docker import nginx_cont.tar vishalbobhate/cloudeithix_busybox:imported
sha256:b96ccca240c4b9729dd260c1fcc9551a6d2de52d21595968b54841fe883f060
root@vishalpc:~/Docker_tasks# docker image ls | grep -i busybox
vishalbobhate/cloudeithix_busybox imported b96ccca240c4 4 seconds ago 150MB
vishalbobhate/cloudeithix_busybox v5 4d01f95718d4 4 minutes ago 150MB
vishalbobhate/cloudeithix_busybox latest 08ef35a1c3f0 13 months ago 4.43MB
vishalbobhate/cloudeithix_busybox <none> 08ef35a1c3f0 13 months ago 4.43MB
root@vishalpc:~/Docker_tasks# docker tag vishalbobhate/cloudeithix_busybox:imported vishalbobhate/cloudeithix_busybox:imported
root@vishalpc:~/Docker_tasks# docker pu
pull (Download an image from a registry) push (Upload an image to a registry)
root@vishalpc:~/Docker_tasks# docker push vishalbobhate/cloudeithix_busybox:imported
The push refers to repository [docker.io/vishalbobhate/cloudeithix_busybox]
1bc9712f895: Pushed
imported: digest: sha256:b6aaa9c155106eac66b84954a51a19a7992f03c6ee3b6e4b0aec59e38023c796 size: 528
root@vishalpc:~/Docker_tasks#

```

The screenshot shows the Docker Hub interface for the repository `vishalbobhate/cloudeithix_busybox`. The repository was last pushed 2 minutes ago and has a size of 59.6 MB. It has 0 stars and 1 pull request. The 'General' tab is selected, showing a table of tags. There is one tag named 'imported' which was pulled less than 1 day ago and pushed 2 minutes ago.

### 30. Dockerfile creation: Create a simple Dockerfile to build a custom image with the following specifications:

Base image: Ubuntu

Install packages: curl, vim, and git

Set an environment variable: MY\_NAME=Your\_Name

Build the custom image using docker build and list all available images using docker images.

```

root@vishalpc:~/Docker_tasks# mkdir Docker_task30
root@vishalpc:~/Docker_tasks# cd Docker_task30/
root@vishalpc:~/Docker_tasks/Docker_task30# nano Dockerfile
root@vishalpc:~/Docker_tasks/Docker_task30# docker build -t task30_img .
[+] Building 52.8s (7/7) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 150B
=> [internal] load metadata for docker.io/library/ubuntu:latest
=> [auth] library/ubuntu:pull token for registry-1.docker.io
=> [internal] load .dockerrcignore
=> => transferring config: 2B
=> [1/2] FROM docker.io/library/ubuntu:latest@sha256:66460d557b25769b102175144d538d8219c077c678a49af4afca6fbfc1b5252
=> => resolve docker.io/library/ubuntu:latest@sha256:66460d557b25769b102175144d538d8219c077c678a49af4afca6fbfc1b5252
=> => sha256:97bed23a3#971024aa8d254abbe67b7168772340d1f494034773bc464e8dd5b6 2.30kB / 2.30kB
=> => sha256:4b3ffdbcc5201a0fc03585952effb4ed2d1ea5e704d2e7330212fb8b10c86a3 29.72MB / 29.72MB
=> => sha256:66460d557b25769b102175144d538d88219c077c678a49af4afca6fbfc1b5252 6.69kB / 6.69kB
=> => sha256:d22e4fb389065ef4a4a61bb364167588698ef6d955fe8a7e0cdb1cd6de80fa7eec 424B / 424B
=> => extracting sha256:4b3ffdbcc5201a0fc03585952effb4ed2d1ea5e704d2e7330212fb8b10c86a3
=> [2/2] RUN apt-get update && apt-get install -y curl vim git
=> => exporting to image
=> => exporting layers
=> => writing image sha256:95b554af674a2c4901b1f77c1eb1731e62debc8aca89a5c76bc20a97da233682
=> => naming to docker.io/library/task30_img
root@vishalpc:~/Docker_tasks/Docker_task30# docker image ls
REPOSITORY TAG IMAGE ID CREATED SIZE
task30_img latest 95b554af674a 19 seconds ago 289MB
vishalbobhate/cloudeithix_busybox imported b96ccca240c4 11 minutes ago 150MB
vishalbobhate/cloudeithix_busybox v5 4d01f95718d4 15 minutes ago 150MB

```

```

=> => naming to docker.io/library/task30_img
root@vishalpc:~/Docker_tasks/Docker_task30# docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
task30_img          latest   95b554af674a  19 seconds ago  289MB
vishalbobhate/cloudeithix_busybox  imported  b96ccca240c4  11 minutes ago  150MB
vishalbobhate/cloudeithix_busybox  v5       4d01f95718d4  15 minutes ago  150MB
mydocker_img         latest   b5768552ea59  2 hours ago   152MB
vishalbobhate/cloudeithix        v4       b5768552ea59  2 hours ago   152MB
vishalbobhate/cloudeithix        latest   2b7f61ec9a0d  4 hours ago   152MB
httpd               latest   6a4fe18d08d2  2 days ago    117MB
nginx              latest   d261fd19cb63  2 days ago    152MB
redis               latest   017b1c12abf9  2 days ago    137MB
vishalbobhate/vishal_cloudeithix_redis  version1  017b1c12abf9  2 days ago    137MB
vishalbobhate/vishal_cloudeithix_redis  version3  017b1c12abf9  2 days ago    137MB
busybox             latest   08ef35alc3f0  13 months ago   4.43MB
vishalbobhate/cloudeithix        v1       08ef35alc3f0  13 months ago   4.43MB
vishalbobhate/cloudeithix_busybox <none>   08ef35alc3f0  13 months ago   4.43MB
root@vishalpc:~/Docker_tasks/Docker_task30# docker run -it task30_img:latest
root@0b7198096658:/# hostname
0b7198096658
root@0b7198096658:/# echo "$MY_NAME"
Vishal_Bobhate
root@0b7198096658:/# |

```

**31. Create Directories:** Establish two directories named "DHUB" and "AWSECR."

```

root@vishalpc:~/Docker_tasks/Docker_task30# mkdir DHUB AWSECR
root@vishalpc:~/Docker_tasks/Docker_task30# ls -ltr
total 12
-rw-r--r-- 1 root root 113 Nov  6 10:56 Dockerfile
drwxr-xr-x 2 root root 4096 Nov  6 11:04 DHUB
drwxr-xr-x 2 root root 4096 Nov  6 11:04 AWSECR
root@vishalpc:~/Docker_tasks/Docker_task30# |

```

**32. Dockerfile Creation:** Develop two Dockerfiles to construct custom images with the following specifications:

- Base image: Ubuntu
- Install packages: httpd
- Add "I am from Docker Hub" to the index.html file for DHUB directory and "I am from ECR" for AWSECR directory.
- Set environment variable ENV\_NAME=DHUB for the DHUB directory and ENV\_NAME=AWSECR for the AWSECR directory. • Start http service using ENTRYPOINT
- Expose port 80.

```

root@vishalpc:~/Docker_tasks/Docker_task30# cd DHUB/
root@vishalpc:~/Docker_tasks/Docker_task30/DHUB# ls
root@vishalpc:~/Docker_tasks/Docker_task30/DHUB# nano Dockerfile
root@vishalpc:~/Docker_tasks/Docker_task30/DHUB# docker build -t Dhub_img .
[+] Building 0.0s (0/0)
ERROR: failed to build: invalid tag "Dhub_img": repository name must be lowercase
root@vishalpc:~/Docker_tasks/Docker_task30/DHUB# docker build -t dhub_img .
[+] Building 37.1s (8/8) FINISHED
    => [internal] load build definition from Dockerfile
    => transferring dockerfile: 255B
    => [internal] load metadata for docker.io/library/ubuntu:latest
    => [auth] library/ubuntu:pull token for registry-1.docker.io
    => [internal] load .dockerrcignore
    => => transferring context: 2B
    => CACHED [1/3] FROM docker.io/library/ubuntu:latest@sha256:66460d557b25769b102175144d538d88219c077c678a49af4afca6fbfc1b5252
    => [2/3] RUN apt-get update && apt-get install -y apache2
    => [3/3] RUN echo "I am from Docker Hub" > /var/www/html/index.html
    => exporting to image
    => => exporting layers
    => => writing image sha256:81521e6f9dfaee71024b5495d2d98b3d47118792164666ec386808fd8bfc9b3b
    => => naming to docker.io/library/dhub_img
root@vishalpc:~/Docker_tasks/Docker_task30/DHUB# docker image ls | grep -i dhub
dhub_img          latest      81521e6f9dfa   About a minute ago   241MB
root@vishalpc:~/Docker_tasks/Docker_task30/DHUB# docker run -d --name dhubapp -p 8090:80 dhub_img:latest
ead6808bd832533a32759d79203871013c138c5a9e91e893ca5b9e42730c547d
root@vishalpc:~/Docker_tasks/Docker_task30/DHUB#

```

GNU nano 7.2 Dockerfile

```

FROM ubuntu:latest

RUN apt-get update && apt-get install -y apache2
RUN echo "I am from Docker Hub" > /var/www/html/index.html
ENV ENV_NAME=DHUB
EXPOSE 80
ENTRYPOINT ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]

```

The screenshot shows a web browser window with the URL <http://localhost:8090>. The page content is "I am from Docker Hub".

I am from Docker Hub

```

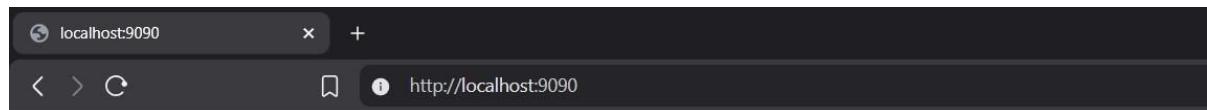
root@vishalpc:~/Docker_tasks/Docker_task30/DHUB# cd ..
root@vishalpc:~/Docker_tasks/Docker_task30# cd AWSECR/
root@vishalpc:~/Docker_tasks/Docker_task30/AWSECR# nano Dockerfile
root@vishalpc:~/Docker_tasks/Docker_task30/AWSECR# docker build -t awsecr_app .
[+] Building 2.9s (8/8) FINISHED
      docker:default
=> [internal] load build definition from Dockerfile          0.1s
=> => transferring dockerfile: 250B                         0.0s
=> [internal] load metadata for docker.io/library/ubuntu:latest   1.9s
=> [auth] library/ubuntu:pull token for registry-1.docker.io    0.0s
=> [internal] load .dockerignore                            0.1s
=> => transferring context: 2B                           0.0s
=> [1/3] FROM docker.io/library/ubuntu:latest@sha256:66460d557b25769b102175144d538d88219c077c678a49af4afca6fbfc1b5252  0.0s
=> CACHED [2/3] RUN apt-get update && apt-get install -y apache2           0.0s
=> [3/3] RUN echo "I am from ECR" > /var/www/html/index.html        0.6s
=> exporting to image                                     0.1s
=> => exporting layers                                0.1s
=> => writing image sha256:b79ad855aef77639a22959dfa98cff0ba80bc4b368f77ab631c84bf6192bf53  0.0s
=> => naming to docker.io/library/awsecr_app            0.0s
root@vishalpc:~/Docker_tasks/Docker_task30/AWSECR# docker image ls | grep -i awse
awsecr_app          latest      b79ad855aef7  14 seconds ago  241MB
root@vishalpc:~/Docker_tasks/Docker_task30/AWSECR# docker run -d --name awsecr_svr -p 9090:80 awsecr_app:latest
55a430ea4197ee0a62e396e6df2c9358faed80582f7202abb2d21d2917f48730

```

```

GNU nano 7.2
FROM ubuntu:latest
RUN apt-get update && apt-get install -y apache2
RUN echo "I am from ECR" > /var/www/html/index.html
ENV ENV_NAME=AWSECR
EXPOSE 80
ENTRYPOINT ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]

```



**33. Build Custom Images:** Utilize the docker build command to build the custom images. List all available images using docker images.

```

root@vishalpc:~/Docker_tasks/Docker_task30/AWSECR# docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
awsecr_app          latest   b79ad855aef7  5 minutes ago  241MB
dhub_img            latest   81521e6f9dfa  14 minutes ago  241MB

```

**34. Push Images to Repositories:** Push the Docker image to Docker Hub. Push the Docker image to AWS ECR.

```

root@vishalpc:~/Docker_tasks/Docker_task30# docker tag awsecr_app:latest vishalbobhate/cloudbethix:v10
root@vishalpc:~/Docker_tasks/Docker_task30# docker tag dhub_img:latest vishalbobhate/cloudbethix:v11
root@vishalpc:~/Docker_tasks/Docker_task30# docker push vishalbobhate/cloudbethix
Using default tag: latest
The push refers to repository [docker.io/vishalbobhate/cloudbethix]
3cef8777468f: Layer already exists
d7217c60dca4: Layer already exists
d81df94f8d07: Layer already exists
99cd1b1b6a43: Layer already exists
2ced4cd78a7b: Layer already exists
8feb164cd673: Layer already exists
6e19587ac541: Layer already exists
36d06fe0cbc6: Layer already exists
latest: digest: sha256:7ba86a332bd8c8b06d4856cedaa225d293b6ed77c7a286587e6a2518284b35aa size: 1985
root@vishalpc:~/Docker_tasks/Docker_task30#

```

```

Default region name [None]:
Default output format [None]:
root@vishalpc:~# aws ecr-public get-login-password --region us-east-1 | docker login --username AWS --password-stdin public.ecr.aws/n6g3p4z8

WARNING! Your credentials are stored unencrypted in '/root/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
root@vishalpc:~# cd -
/root/Docker_tasks/Docker_task30
root@vishalpc:~/Docker_tasks/Docker_task30# docker build -t vishal/apache2_01 .
[+] Building 4.2s (7/7) FINISHED
  => [internal] load build definition from Dockerfile
  => => transferring Dockerfile: 150B
  => [internal] load metadata for docker.io/library/ubuntu:latest
  => [auth] library/ubuntu:pull token for registry-1.docker.io
  => [internal] load .dockerrcignore
  => => transferring context: 2B
  => [1/2] FROM docker.io/library/ubuntu:latest@sha256:66460d557b25769b102175144d538d88219c077c678a49af4afca6fbfc1b5252
  => CACHED [2/2] RUN apt-get update && apt-get install -y curl vim
  => exporting to image
  => => exporting layers
  => => writing image sha256:95b554af674a2c4901b1f77c1eb1731e62debc8aca89a5c76bc20a97da233682
  => => naming to docker.io/vishal/apache2_01
root@vishalpc:~/Docker_tasks/Docker_task30# docker tag vishal/apache2_01:latest public.ecr.aws/n6g3p4z8/vishal/apache2_01:latest
root@vishalpc:~/Docker_tasks/Docker_task30# docker push public.ecr.aws/n6g3p4z8/vishal/apache2_01:latest
The push refers to repository [public.ecr.aws/n6g3p4z8/vishal/apache2_01]
fc1989447a26: Pushed
073ec47a8c22: Pushed
latest: digest: sha256:d13b6cb5922991023c49713a6f09ffe902c291ebb0c6c6925d386fb9db220657 size: 741
root@vishalpc:~/Docker_tasks/Docker_task30#

```

### 35. Run Containers:

```
docker run -d -p 8081:80 --name I_AM_FROM_DHUB your_docker_hub_image
```

```
docker run -d -p 8082:80 --name I_AM_FROM_ECR your_aws_ecr_image
```

```

Run 'docker run --help' for more information
root@vishalpc:~/Docker_tasks/Docker_task30# docker run -d -p 8020:80 --name I_AM_FROM_DHUB vishalbobhate/cloudbethix:latest
a485817e078eed6d23d76b7eb66265d2632a931e248a1098e43c06d303277cf6
root@vishalpc:~/Docker_tasks/Docker_task30# docker run -d -p 8030:80 --name I_AM_FROM_ECR public.ecr.aws/n6g3p4z8/vishal/apache2_01
c4eb666e913fc71291a0f739b4842827fbcc30810bb0e469c37bc398365b7fb5
root@vishalpc:~/Docker_tasks/Docker_task30#
root@vishalpc:~/Docker_tasks/Docker_task30#

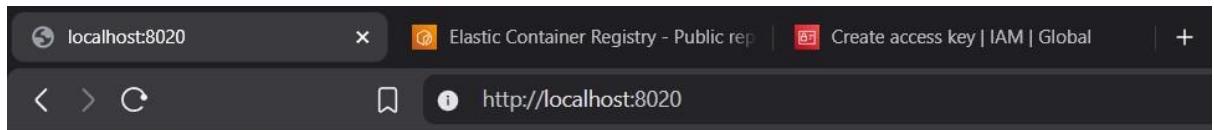
```

### 36. Access Pages from Browser:

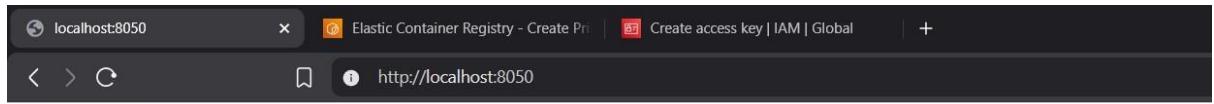
Open a web browser and access the pages:

For Docker Hub: <http://localhost:8081>

For AWS ECR: <http://localhost:8082>



hello everyone this is my web1



I am from ECR

**37. Dockerfile creation:** Create a simple Dockerfile to build a custom image with the following specifications:

- Base image: Ubuntu
- Install packages: curl, vim, and git
- Set an environment variable: MY\_NAME=Your\_Name
- Build the custom image using docker build and list all available images using docker images.

```
root@vishalpc:~/11/task1# nano Dockerfile
root@vishalpc:~/11/task1# docker build -t task1_img .
[+] Building 38.6s (6/6) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 240B
=> [internal] load metadata for docker.io/library/ubuntu:latest
=> [internal] load .dockerignore
=> => transferring context: 2B
=> CACHED [1/2] FROM docker.io/library/ubuntu:latest@sha256:66460d557b25769b102175144d538d88219c077c678a49af4afca6fbfc1b5252
=> [2/2] RUN apt update && apt install -y curl vim git
=> exporting to image
=> => exporting layers
=> => writing image sha256:886904700ee37d92f9b9b1201377b3b3a8b2b3e69a1900539b5218dd1aa1577
=> => naming to docker.io/library/task1_img
root@vishalpc:~/11/task1# docker images
REPOSITORY          TAG           IMAGE ID        CREATED         SIZE
task1_img           latest        886904700ee3   15 seconds ago  289MB
mariadb             latest        7418634b3691   15 hours ago   330MB
nginx               latest        d261fd19cb63    7 days ago     152MB
mysql               latest        0e642d18a936   2 weeks ago    783MB
root@vishalpc:~/11/task1# cat Dockerfile
#Base image: Ubuntu
FROM ubuntu:latest
# Install packages: curl, vim, and git
RUN apt update && apt install -y curl vim git
#Set an environment variable: MY_NAME=Your_Name
ENV MY_NAME=VISHAL_BOBHATE
root@vishalpc:~/11/task1# |
```

**38. Docker networking:** Create two Docker containers with different names, and add them to the same custom network. Verify that the containers can communicate with each other using their names as hostnames.

```
]
root@vishalpc:~/11/task1# docker exec -it mycont22 /bin/bash
bash-5.1# curl newname
this is the app running on my localhost
bash-5.1#
bash-5.1#
bash-5.1#
bash-5.1# curl 172.18.0.3
this is the app running on my localhost
bash-5.1#
bash-5.1#
```

**39. Data persistence with volumes:** Run a Docker container with an attached volume, create a file inside the volume, and verify that the file persists after the container is stopped and removed.

```
root@vishalpc:~/11/task3# ls
root@vishalpc:~/11/task3# docker run -dit --name vol_cont -v myvolume:/data -p81:80 /bin/bash
docker: invalid reference format

Run 'docker run --help' for more information
root@vishalpc:~/11/task3# docker run -dit --name vol_cont -v myvolume:/data -p81:80 nginx /bin/bash
3f02c0d3a2936e7bd77bc752ab2717b5f823908847181193f9aff1bb67697221
root@vishalpc:~/11/task3# docker exec -it 3f02c0d3a2936 /bin/bash
root@3f02c0d3a293:#
bin  boot  data  dev  docker-entrypoint.d  docker-entrypoint.sh  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp
sr  var
root@3f02c0d3a293:/# cd data
root@3f02c0d3a293:/data# ls
root@3f02c0d3a293:/data# touch 1.txt 2.txt
root@3f02c0d3a293:/data# echo "hello vishal" > 3.txt
root@3f02c0d3a293:/data# ls
1.txt 2.txt 3.txt
root@3f02c0d3a293:/data# exit
exit
root@vishalpc:~/11/task3# docker stop vol_cont
vol_cont
root@vishalpc:~/11/task3# docker rm vol_cont
vol_cont
root@vishalpc:~/11/task3# docker run -dit --name new_cont2_volum -v myvolume:/data -p81:80 nginx /bin/bash
dd5893091330a7fa6bd4857a3931833434695a1781b8830b414a47177b5d9e45
root@vishalpc:~/11/task3# docker exec -it dd5893091330a /bin/bash
root@dd5893091330:#
bin  boot  data  dev  docker-entrypoint.d  docker-entrypoint.sh  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp
sr  var
root@dd5893091330:/# cd data
root@dd5893091330:/data# ls
1.txt 2.txt 3.txt
root@dd5893091330:/data# cat 3.txt
hello vishal
root@dd5893091330:/data# |
```

**40. Docker Compose:** Create a docker-compose.yml file to run a multi-container application consisting of a web server (e.g., Nginx) and a database server (e.g., MySQL). Ensure that both containers are connected to the same network.

```
root@vishalpc:~/11/task4# docker compose up -d
WARN[0000] /root/11/task4/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential configuration
[+] Running 4/4
  ✓ Network task4_app_network  Created                               0.1s
  ✓ Volume task4_db_data      Created                               0.05s
  ✓ Container mysql_db        Created                               0.2s
  ✘ Container webcont         Error response from daemon: invalid volume specification: '/root/11/task4/html:/usr/share/nginx/html:rw': invalid mount config for type "bind": i
Error response from daemon: invalid volume specification: '/root/11/task4/html:/usr/share/nginx/html' mount path must be absolute
root@vishalpc:~/11/task4# nano docker-compose.yml
root@vishalpc:~/11/task4# docker compose up -d
WARN[0000] /root/11/task4/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential configuration
[+] Running 2/2
  ✓ Container webcont        Started                               1.3s
  ✓ Container mysql_db        Started                               0.6s
root@vishalpc:~/11/task4# cat docker-compose.yml
version: '1'
services:
  web:
    image: nginx:latest
    container_name: webcont
    ports:
      - "82:80"
    depends_on:
      - db
    networks:
      - app_network
    volumes:
      - ./html:/usr/share/nginx/html
```

**41. Container logging:** Run a Docker container that generates log output, and practice using docker logs to view and analyze the logs.

```
root@vishalpc:~/11/task5# docker logs webcont
/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: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/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
2025/11/11 11:05:42 [notice] 1#1: using the "epoll" event method
2025/11/11 11:05:42 [notice] 1#1: nginx/1.29.3
2025/11/11 11:05:42 [notice] 1#1: built by gcc 14.2.0 (Debian 14.2.0-19)
2025/11/11 11:05:42 [notice] 1#1: OS: Linux 6.6.87.2-microsoft-standard-WSL2
2025/11/11 11:05:42 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/11/11 11:05:42 [notice] 1#1: start worker processes
2025/11/11 11:05:42 [notice] 1#1: start worker process 29
2025/11/11 11:05:42 [notice] 1#1: start worker process 30
2025/11/11 11:05:42 [notice] 1#1: start worker process 31
2025/11/11 11:05:42 [notice] 1#1: start worker process 32
```

```

root@vishalpc:~/11/task5# docker logs --tail -5 webcont
/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: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/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
2025/11/11 11:05:42 [notice] 1#1: using the "epoll" event method
2025/11/11 11:05:42 [notice] 1#1: nginx/1.29.3
2025/11/11 11:05:42 [notice] 1#1: built by gcc 14.2.0 (Debian 14.2.0-19)
2025/11/11 11:05:42 [notice] 1#1: OS: Linux 6.6.87.2-microsoft-standard-WSL2
2025/11/11 11:05:42 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/11/11 11:05:42 [notice] 1#1: start worker processes
2025/11/11 11:05:42 [notice] 1#1: start worker process 29
2025/11/11 11:05:42 [notice] 1#1: start worker process 30
2025/11/11 11:05:42 [notice] 1#1: start worker process 31
2025/11/11 11:05:42 [notice] 1#1: start worker process 32
2025/11/11 11:05:42 [notice] 1#1: start worker process 33
2025/11/11 11:05:42 [notice] 1#1: start worker process 34

```

- 42.** Container resource monitoring: Run a Docker container that consumes system resources (e.g., CPU or memory) and practice using docker stats and docker top to monitor the container's resource usage

```

root@vishalpc:~/11/task5# docker run -dit --name task06 --cpus=1 --memory=256m -p91:80 nginx
da96c4882fc44750522c3d4787988ca5481f2c90bd8b7391380fc176323c2892
root@vishalpc:~/11/task5#

```

```

root@vishalpc:~/11/task5# docker top task06
UID          PID  PPID      C      STIME     TTY      TIME     CMD
root        9745  9722      0   11:22 pts/0    00:00:00  nginx
x: master process nginx -g daemon off;
message+    9821  9745      0   11:22 pts/0    00:00:00  nginx
x: worker process
message+    9822  9745      0   11:22 pts/0    00:00:00  nginx
x: worker process
message+    9823  9745      0   11:22 pts/0    00:00:00  nginx
x: worker process
message+    9824  9745      0   11:22 pts/0    00:00:00  nginx
x: worker process
message+    9825  9745      0   11:22 pts/0    00:00:00  nginx
x: worker process
message+    9826  9745      0   11:22 pts/0    00:00:00  nginx
x: worker process
message+    9827  9745      0   11:22 pts/0    00:00:00  nginx
x: worker process
message+    9828  9745      0   11:22 pts/0    00:00:00  nginx
x: worker process
root@vishalpc:~/11/task5# docker stats task06

```

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET I/O	BLOCK I/O	PIDS
da96c4882fc4	task06	0.00%	8.758MiB / 256MiB	3.42%	866B / 126B	4.02MB / 12.3kB	9

**43. Bridge network:** Create two Docker containers using different images, such as NGINX and MySQL. Connect them using a user-defined bridge network and ensure they can communicate with each other.

```
7b983ddce354  mysql      "docker-entrypoint.s..."  2 hours ago   Up 2 hours    3306/tcp, 33060/tcp          mycont22
518f2c294a96  nginx      "/docker-entrypoint..."  2 hours ago   Up 2 hours    80/tcp                         newname
root@vishalpc:~/11/task5#
```

```
]
root@vishalpc:~/11/task1# docker exec -it mycont22 /bin/bash
bash-5.1# curl newname
this is the app running on my localhost
bash-5.1#
bash-5.1#
bash-5.1#
bash-5.1# curl 172.18.0.3
this is the app running on my localhost
bash-5.1#
bash-5.1#
```

**44. Port mapping:** Run a containerized web server (e.g., NGINX or Apache) and map its default port (80 or 443) to a custom port on the host machine. Verify that the web server is accessible through the custom port on the host.

```
root@vishalpc:~/11/task5# docker run -dit --name webtask09 -p443:80 nginx
db66085e2a51774fc7191064699a002807dfb0ae01581aa0b97f87d92ee22a73
root@vishalpc:~/11/task5# docker exec -it db66085e2a51774f /bin/bash
root@db66085e2a51:/# echo "this is the website running for task09" index.html
this is the website running for task09 index.html
root@db66085e2a51:/# echo "this is the website running for task09" > index.html
root@db66085e2a51:/# mv index.html /usr/share/nginx/html/
root@db66085e2a51:/# exit
exit
root@vishalpc:~/11/task5#
```



this is the website running for task09

**45. Host networking:** Run a container with the host networking mode and compare its network configuration with that of the host machine. Explain the advantages and disadvantages of using host networking for containers.

```
[root@vishalpc:~/11/task5# docker inspect task10 | grep -i "network"
    "NetworkMode": "host",
    "NetworkSettings": {
        "Networks": {
            "NetworkID": "7fa72baa000a7c59668294761e98f8a94e17ed324560f8c514363bd1c2197e17",
root@vishalpc:~/11/task5# ]
```

### Advantages

- **Improved performance:** Because the container shares the host's network stack, there is no overhead from Network Address Translation (NAT) or port mapping, leading to faster network communication.
- **Direct access:** The container can directly access services on the host without needing port mapping, as it uses the host's IP address and ports.

### Disadvantages

```
[root@vishalpc:~/11/task5# docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
2d04fdc9c9cb   bridge     bridge      local
7fa72baa000a   host       host       local
5d7b351bf968   mynetwork   bridge      local
5df6ee3e10cb   none       null      local
3bb5b04106a9   task4_app_network   bridge      local
root@vishalpc:~/11/task5# docker run -dit --name task10 --network host nginx
3e35e8c3afdd813fc627bb0d8e16cab9539cab27a6c5d4a9b5a63c0276687113
```

- **Lack of isolation:** The host network driver bypasses the isolation provided by Docker's default network, meaning a security breach in one container could potentially compromise the host machine and other containers on the same host.
- **Port conflicts:** Multiple containers using the host network cannot use the same port, as they are all sharing the same network namespace. This prevents running two services that need to listen on the same port on the same host.

**46. Named volume:** Launch a containerized database (e.g., MySQL or PostgreSQL) using a named volume to store its data. Stop and remove the container, then recreate it using the same named volume. Verify that the data persists across container recreations.

```
[root@vishalpc:/var/lib/docker/volumes/myvolume/_data# docker run -dit --name mysql10 -v sql_volume:/var/lib/mysql/ --env MYSQL_ROOT_PASSWORD="root123" mysql
4bb70d7054b6b79bdd090361bf57fce5e763b21c12d4e974631b313f528bc633
root@vishalpc:/var/lib/docker/volumes/myvolume/_data# docker exec -it mysql10 /bin/bash
bash-5.1# mysql -u root -proot123
```

```

mysql> select * from company
-> ;
ERROR 1146 (42S02): Table 'company.company' doesn't exist
mysql> select * from employees;
+----+-----+-----+-----+-----+-----+
| id | first_name | last_name | department | salary | hire_date |
+----+-----+-----+-----+-----+-----+
| 1 | Vishal     | Bobhate    | Cloud      | 75000.00 | 2023-04-01 |
| 2 | Riya       | Sharma     | DevOps     | 80000.00 | 2022-11-15 |
| 3 | Amit        | Patil      | Security   | 90000.00 | 2021-06-10 |
| 4 | Neha       | Kadam      | Development | 70000.00 | 2024-01-05 |
+----+-----+-----+-----+-----+-----+
4 rows in set (0.002 sec)

mysql> exit
Bye
bash-5.1# exit
exit

```

```

root@vishalpc:~/mysql# docker rm mysql10 mysql4
mysql10

```

```

root@vishalpc:~/mysql# docker run -dit --name newmysql -v sql_volume:/var/lib/mysql/ --env MYSQL_ROOT_PASSWORD="root123" mysql
c5570e30330dec5739823fdb56cd69a450c9e9efc0a73b15f8da84dde4a42dd7
root@vishalpc:~/mysql# docker exec -it c5570e30330d /bin/bash
bash-5.1# mysql -u root -proot123;
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 9.5.0 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| company |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.162 sec)

```

```

mysql> use company;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

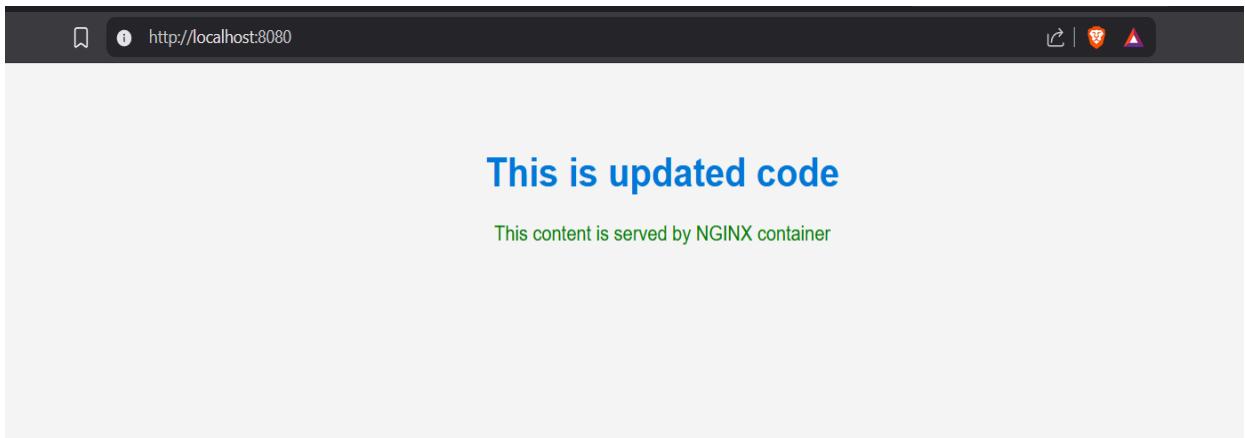
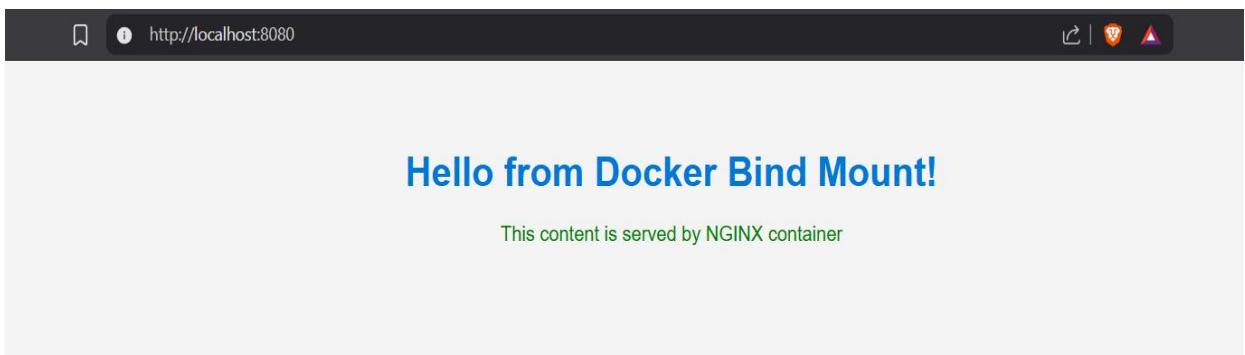
Database changed
mysql> select * from employees;
+----+-----+-----+-----+-----+
| id | first_name | last_name | department | salary | hire_date |
+----+-----+-----+-----+-----+
| 1 | Vishal     | Bobhate    | Cloud      | 75000.00 | 2023-04-01 |
| 2 | Riya       | Sharma     | DevOps     | 80000.00 | 2022-11-15 |
| 3 | Amit        | Patil      | Security   | 90000.00 | 2021-06-10 |
| 4 | Neha       | Kadam      | Development | 70000.00 | 2024-01-05 |
+----+-----+-----+-----+-----+
4 rows in set (0.002 sec)

mysql> exit;
Bye
bash-5.1# exit
exit
root@vishalpc:~/mysql# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
c5570e30330d mysql "docker-entrypoint.s..." 3 minutes ago Up 3 minutes 3306/tcp, 33060/tcp newmysql

```

**47. Bind mount:** Create a simple web application with a local directory containing HTML, CSS, and JavaScript files. Run a web server container (e.g., NGINX or Apache) and bind mount the local directory to the container's webroot. Verify that the web application is accessible and update the local files to see if changes are reflected in the container.

```
root@vishalpc:~# mkdir webaap
root@vishalpc:~# cd webaap/
root@vishalpc:~/webaap# nano index.html
root@vishalpc:~/webaap# nano style.css
root@vishalpc:~/webaap# nano script.js
root@vishalpc:~/webaap# docker run -d --name bindmountapp -p 8080:80 -v $(pwd):/usr/share/nginx/html nginx
61e000f7549c2ff9a485c88ff891ff13f4dcfddfe6bd21f04a12ad5fb91e8232
root@vishalpc:~/webaap# curl localhost
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
      border: 1px solid black;
      padding: 15px; }
```



**48. Volume backups:** Using a container with a named volume, create a backup of the volume's data by either exporting it as a tarball or copying the data to a host directory. Restore the data to a new container and verify that the restoration was successful.

```
root@vishalpc:~/localbackup# cd localbackup/
root@vishalpc:~/localbackup# docker run -dit --name backup_cont11 -v backup_vol1:/data -v $(pwd):/backup nginx
a601fcb5c12b9b76b2f1241c1299eb00f5b79da01cf6c2b2389b3fd9c3832c7b
root@vishalpc:~/localbackup# docker exec -it backup_cont11 /bin/bash
root@a601fcb5c12b:#
ls
backup boot dev docker-entrypoint.sh home lib64 mnt proc run srv tmp var
bin data docker-entrypoint.d etc lib media opt root sbin sys usr
root@a601fcb5c12b:#
mkdir hello
root@a601fcb5c12b:#
cd hello/
root@a601fcb5c12b:hello# touch 1.txt 2.txt 3.txt 4.txt
root@a601fcb5c12b:hello# ls
1.txt 2.txt 3.txt 4.txt
```

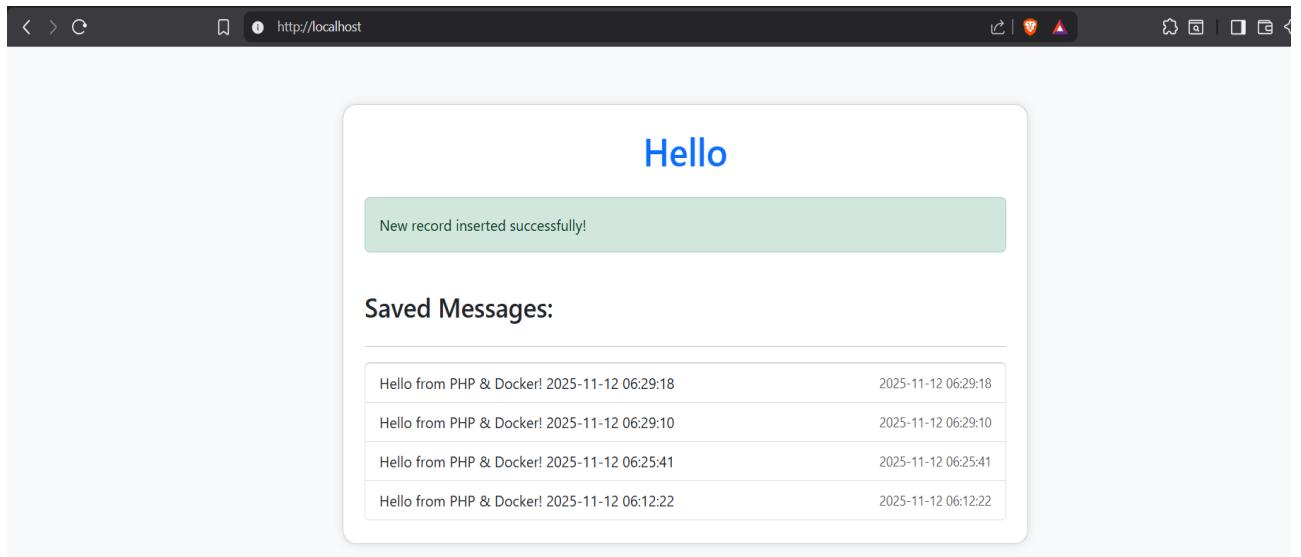
```
root@vishalpc:~/localbackup# docker exec -it backup_cont1 /bin/bash
root@176544a7fe1f:#
ls
backup boot dev docker-entrypoint.sh home lib64 mnt proc run srv tmp var
bin data docker-entrypoint.d etc lib media opt root sbin sys usr
```

```
root@a601fcb5c12b:#
ls backup/
cont1_backup.tar.gz
root@a601fcb5c12b:#
exit
root@vishalpc:~/localbackup# ls
cont1_backup.tar.gz
root@vishalpc:~/localbackup# docker run -dit --name backup_cont22 -v $(pwd):/backup nginx
4bec8ff48f03dbe26beea6d5f1f0b70ea45b8088726862b56a9bb56ad06d936e
root@vishalpc:~/localbackup# docker exec -it backup_cont22 /bin/bash
root@4bec8ff48f03:#
ls
backup bin boot dev docker-entrypoint.d docker-entrypoint.sh etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
root@4bec8ff48f03:#
ls backup/
cont1_backup.tar.gz
root@4bec8ff48f03:#
tar xzf backup/cont1_backup.tar.gz -C mnt/
gzip: stdin: not in gzip format
tar: Child returned status 1
tar: Error is not recoverable: exiting now
root@4bec8ff48f03:#
tar xvf backup/cont1_backup.tar.gz -C mnt/
data/
data/hello/
data/hello/2.txt
data/hello/1.txt
data/hello/3.txt
data/hello/4.txt
data/task.txt
data/creating the 1st volume at backup_cont1
root@4bec8ff48f03:#
ls mnt/
data
root@4bec8ff48f03:#
ls mnt/data/
'creating the 1st volume at backup_cont1'  hello  task.txt
root@4bec8ff48f03:|
```

**49. Docker Compose networking:** Create a Docker Compose file that defines a multi-container application with a frontend, backend, and database. Set up custom networks and volumes for the services and ensure that they can communicate with each other and store data persistently

```
root@vishalpc:~/localbackup/task14/task_50# docker compose up -d
WARN[0000] /root/localbackup/task14/task_50/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 3/3
  ✓ Container task_50-mydb-1  Running
  ✓ Container task_50-myapp-1  Running
  ✓ Container task_50-myweb-1  Running
root@vishalpc:~/localbackup/task14/task_50# tree
.
├── app
│   └── Dockerfile
│   └── index.php
└── docker-compose.yml
└── nginx
    └── default.conf

3 directories, 4 files
root@vishalpc:~/localbackup/task14/task_50#
```



```
root@vishalpc:~/localbackup/task14/task_50# docker exec -it task_50-mydb-1 /bin/bash
bash-5.1# mysql -u root -pPass@123
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.4.7 MySQL Community Server - GPL
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| wordpressdb |
+-----+
5 rows in set (0.03 sec)

mysql> use wordpressdb
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_wordpressdb |
+-----+
| messages |
+-----+
1 row in set (0.01 sec)

mysql> select * from messages;
+-----+-----+
| id | message | created_at |
+-----+-----+
| 1 | Hello from PHP & Docker! 2025-11-12 06:12:22 | 2025-11-12 06:12:22 |
| 2 | Hello from PHP & Docker! 2025-11-12 06:25:41 | 2025-11-12 06:25:41 |
| 3 | Hello from PHP & Docker! 2025-11-12 06:29:10 | 2025-11-12 06:29:10 |
| 4 | Hello from PHP & Docker! 2025-11-12 06:29:18 | 2025-11-12 06:29:18 |
+-----+
4 rows in set (0.00 sec)

mysql> |
```

