

Ques 4 : Create a docker image from multiple methods likes Dockerfile, running containers

SOLN

Using a Dockerfile

```
/myapp
├── Dockerfile
├── app.js
└── package.json
```

Docker file :

Base image
FROM node:18-alpine

Set working directory
WORKDIR /app

Copy files
COPY . .

Install dependencies
RUN npm install

Start the app
CMD ["npm", "start"]

docker build -t myapp-image .

```
PS D:\OneDrive\Desktop\my portfolio> cd frontend
PS D:\OneDrive\Desktop\my portfolio\frontend> docker build -t myportfolio-image .
[+] Building 6.0s (14/14) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 310B
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [internal] load metadata for docker.io/library/node:18-alpine
=> [auth] library/node:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [stage-1 1/2] FROM docker.io/library/nginx:alpine
=> [internal] load build context
=> => transferring context: 3.61MB
=> [builder 1/5] FROM docker.io/library/node:18-alpine@sha256:8d6421d663b4c28fd3ebc498332f249011d118945588d0a35cb9bc4b8ca09d9e
=> CACHED [builder 2/5] WORKDIR /app
=> CACHED [builder 3/5] COPY . .
=> CACHED [builder 4/5] RUN npm install
=> CACHED [builder 5/5] RUN npm run build
=> CACHED [stage-1 2/2] COPY --from=builder /app/build /usr/share/nginx/html
=> exporting to image
=> => exporting layers
=> => writing image sha256:e1434b72b301bd7f189b5bf043a06a5f2b8d82ed98fa5bf428d5920dfaa1235f
=> => naming to docker.io/library/myportfolio-image
PS D:\OneDrive\Desktop\my portfolio\frontend>
PS D:\OneDrive\Desktop\my portfolio\frontend>
PS D:\OneDrive\Desktop\my portfolio\frontend>
PS D:\OneDrive\Desktop\my portfolio\frontend>
PS D:\OneDrive\Desktop\my portfolio\frontend> docker image ls
REPOSITORY          TAG       IMAGE ID       CREATED        SIZE
myportfolio-image   latest    e1434b72b301  18 minutes ago  59.4MB
```

2 > From a Running Container (Manual to Image)

Run or exec into the container and make changes

`docker run -it node:18-alpine sh`

Create Image from That Container:

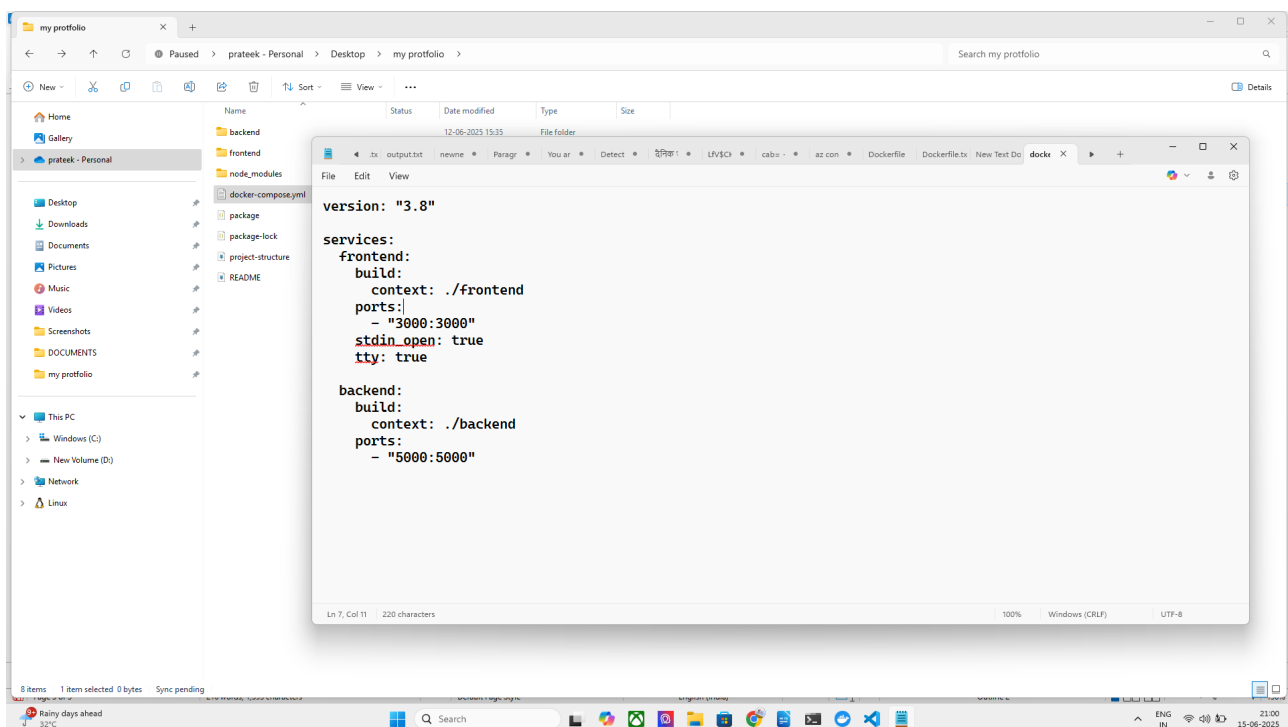
`docker commit <container_id_or_name> my-custom-image`

```
C:\Users\prate>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS        NAMES
63e3f936ad0a   node:18-alpine   "docker-entrypoint.s..." About a minute ago   Exited (0) About a minute ago           wonderful_ishizaka
35cc901ad308   myportfolio-frontend   "/docker-entrypoint.s..." 22 minutes ago     Exited (0) 19 minutes ago           myportfolio-frontend-1
aaaac8855e7a   myportfolio-backend   "docker-entrypoint.s..." 22 minutes ago     Exited (137) 19 minutes ago           myportfolio-backend-1
3d7dc131f7f8   my-python-app     "python app.py"          About an hour ago   Exited (0) About an hour ago           eager_ishizaka
a4f2e4e2aa17   my-python-app     "python app.py"          2 hours ago        Exited (0) 2 hours ago           distracted_feynman
cfd6a39427a3   ubuntu            "/bin/bash"              2 hours ago        Exited (130) 2 hours ago           youthful_clarke
859f02866489   dafae1c9b14       "/docker-entrypoint.s..." 3 hours ago        Exited (0) 3 hours ago           quizzical_wright
1a2f1dfc5008   dafae1c9b14       "/docker-entrypoint.s..." 3 hours ago        Exited (0) 3 hours ago           musing_fermi

C:\Users\prate>docker commit 63e3f936ad0a new_image
sha256:e4b8d466ac3cf8ca414c03ba56e5467810ab917c29c3e9ff9b3d5e945d5005e4

C:\Users\prate>docker image ls
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
new_image     latest    e4b8d466ac3c   8 seconds ago  127MB
myportfolio-frontend   latest    cb66c6ca0e59   22 minutes ago  59.4MB
myportfolio-image     latest    e1434b72b301   22 minutes ago  59.4MB
myportfolio-backend   latest    f3c8e4a635f3   23 minutes ago  131MB
<none>         <none>    da30a22769d8   34 minutes ago  551MB
my-python-app   latest    44c082b8ea65   2 hours ago     127MB
ubuntu          latest    bf16bdcff9c9   2 weeks ago     78.1MB
nginx           alpine    6769dc3a703c   2 months ago    48.2MB
nginx           latest    1e5f3c5b981a   2 months ago    192MB
prateek2004/nginx-custom latest    1e5f3c5b981a   2 months ago    192MB
node            18-alpine ee77c6cd7c18   2 months ago    127MB
```

3. Using docker - compose + Dockerfile



NOW THE COMMAND CHANGES TO

docker-compose up -build

```
Windows PowerShell
PS D:\OneDrive\Desktop\my protfolio> docker-compose up --build
time="2025-06-15T21:21:33+05:30" level=warning msg="D:\OneDrive\Desktop\my protfolio\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
Compose can now delegate builds to bake for better performance.
To do so, set COMPOSE_BAKE=true.
[*] Building 49.6s (22/22) FINISHED
=> [backend internal] load build definition from Dockerfile
=> => transferring dockerfile: 137B
=> [frontend internal] load metadata for docker.io/library/node:18-alpine
=> [backend auth] library/node:pull token for registry-1.docker.io
=> [backend internal] load .dockerignore
=> => transferring context: 2B
=> [frontend builder 1/5] FROM docker.io/library/node:18-alpine@sha256:8d6421d663b4c28fd3ebc498332f249011d118945588d0a35cb9bc4b8ca09d9e
=> [backend internal] load build context
=> => transferring context: 3.35MB
=> CACHED [frontend builder 2/5] WORKDIR /app
=> [backend 3/4] COPY . .
=> [backend 4/4] RUN npm install
=> [backend] exporting to image
=> => exporting layers
=> => writing image sha256:f3c8e4a635f3e59b044ce394b8d851958353d81178d75c3b855dce91c598aef1
=> => naming to docker.io/library/myprotfolio-backend
=> [backend] resolving provenance for metadata file
=> [frontend internal] load build definition from Dockerfile
=> => transferring dockerfile: 310B
=> [frontend internal] load metadata for docker.io/library/nginx:alpine
=> [frontend internal] load .dockerignore
=> => transferring context: 2B
=> CACHED [frontend stage-1 1/2] FROM docker.io/library/nginx:alpine
=> [frontend internal] load build context
=> => transferring context: 3.61MB
=> [frontend builder 3/5] COPY . .
=> [frontend builder 4/5] RUN npm install
=> [frontend builder 5/5] RUN npm run build
=> [frontend stage-1 2/2] COPY --from=builder /app/build /usr/share/nginx/html
=> [frontend] exporting to image
=> => exporting layers
=> => writing image sha256:cb66c6ca0e59f2883bd0e7484157fe0877ba86f98bcb694d9afd8c862cb4c63
=> => naming to docker.io/library/myprotfolio-frontend
=> [frontend] resolving provenance for metadata file
[*] Running 4/4
✓ backend Built 0.0s
✓ frontend Built 0.0s
✓ Container myprotfolio-backend-1 Created 0.1s
✓ Container myprotfolio-frontend-1 Recreated 0.1s
Attaching to backend-1, frontend-1
backend-1 | Server running on port 5000
backend-1 | Frontend being served from: /frontend/build
frontend-1 | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
frontend-1 | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
frontend-1 | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
```

No images found matching "ls": did you mean "docker image ls"?

```
PS D:\OneDrive\Desktop\my protfolio> docker image ls
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
myprotfolio-frontend	latest	cb66c6ca0e59	3 minutes ago	59.4MB
myprotfolio-backend	latest	f3c8e4a635f3	4 minutes ago	131MB
<none>	<none>	da30a22769d8	14 minutes ago	551MB

4. Using docker export and import (Advanced)

Export container filesystem

docker export <container_id> > container.tar

Import it as an image

for linux

cat container.tar | docker import --my-exported-image

for windows

docker import container.tar my-exported-image

```
C:\Users\prate>docker images
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
my-exported-image    latest      4e8f7fbcc187 6 seconds ago 127MB
```

5. Manually via docker buildx or buildctl