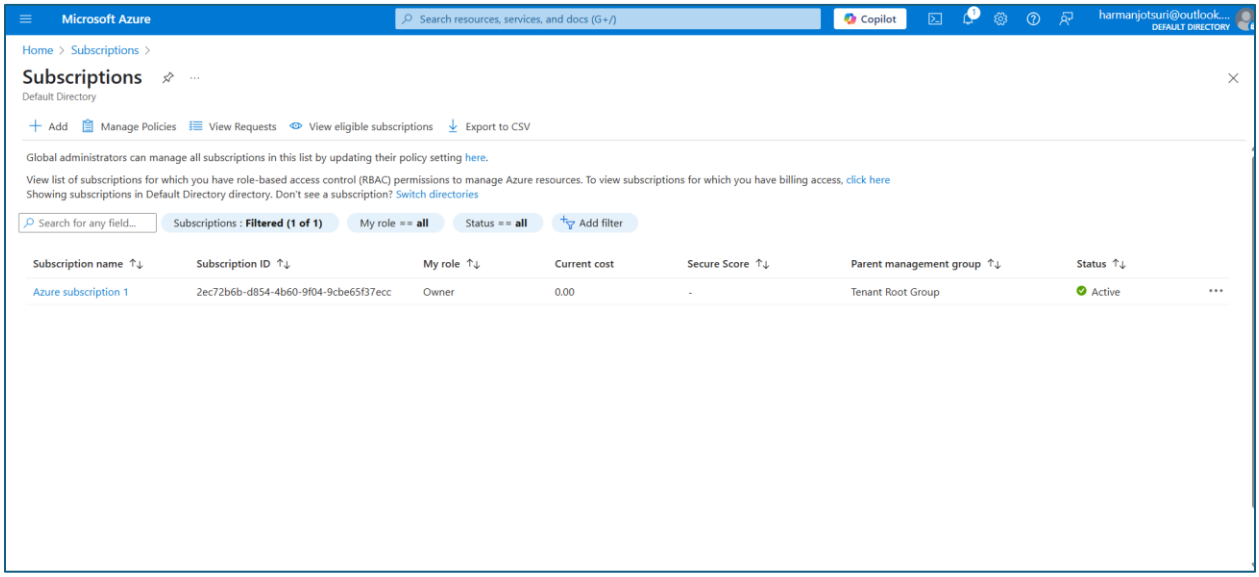
Deploying a Dockerized Application on Azure Kubernetes Service (AKS)

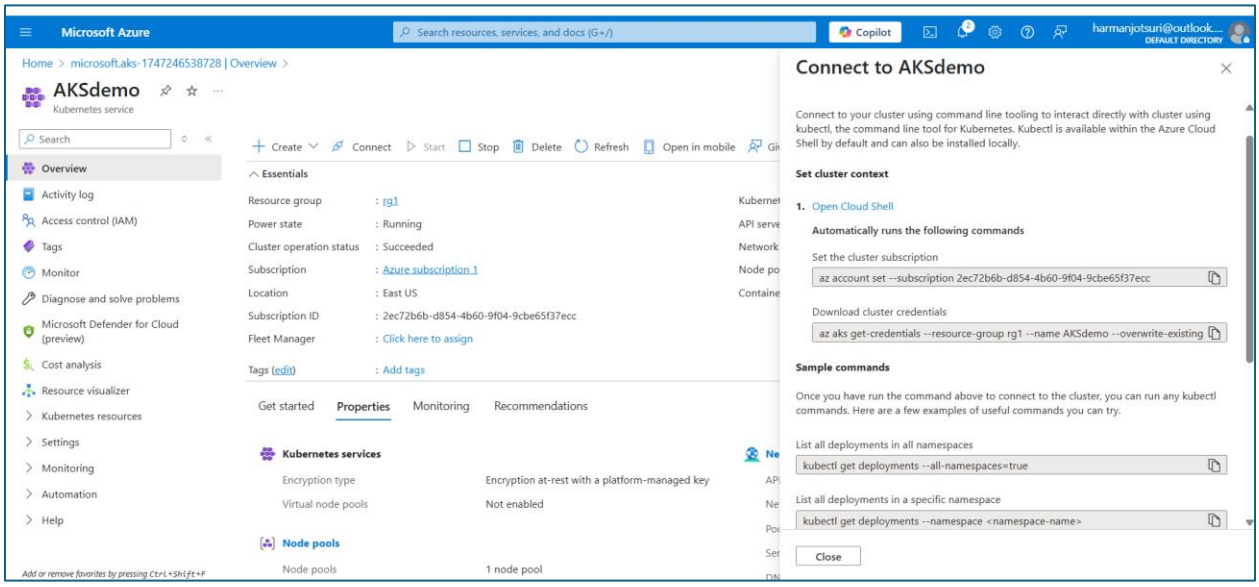
Objective: Containerize and deploy an application on AKS using Docker.

Github URL: https://github.com/singh932/NodeJSDockerK8sAzure

Steps:

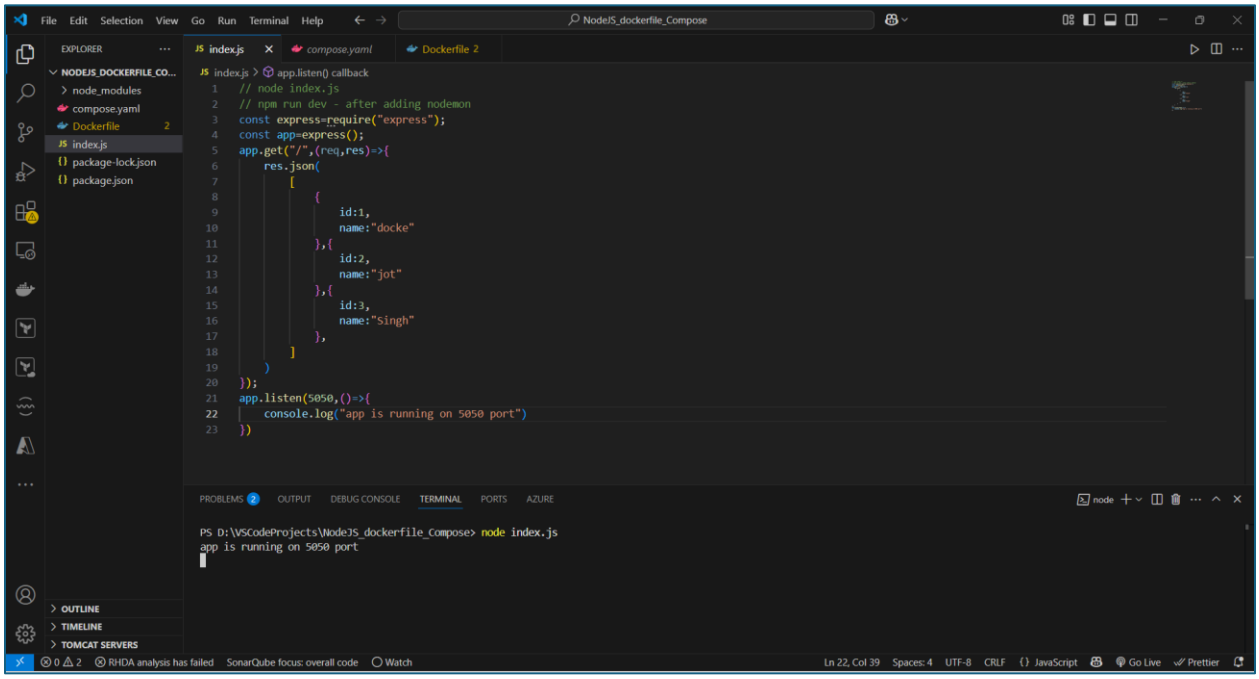
1.Set up an Azure account and create an AKS cluster.

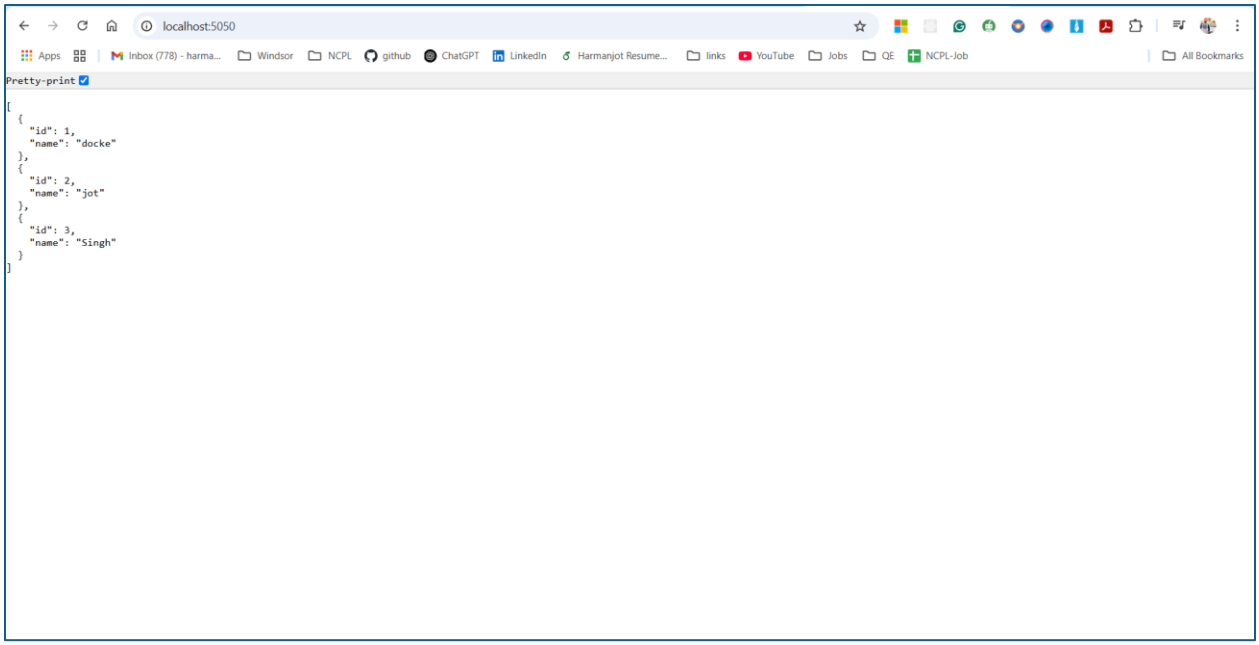




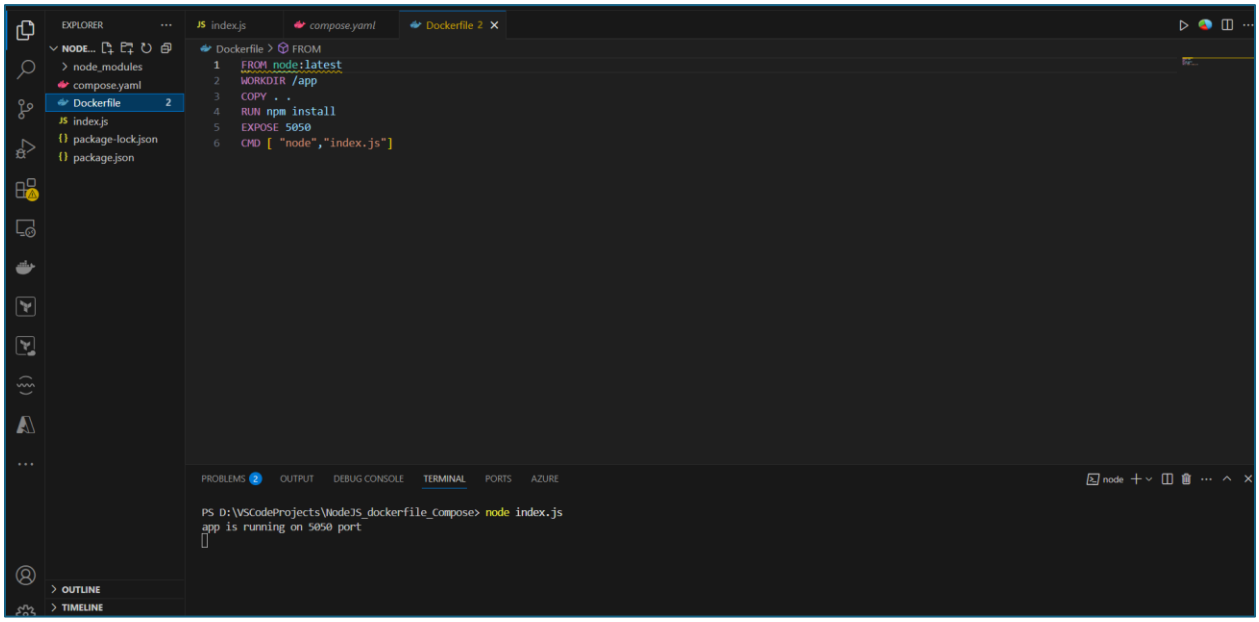
```
PROBLEMS 8   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS   AZURE

PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> az aks get-credentials --resource-group rg1 --name AKSdemo --overwrite-existing
```

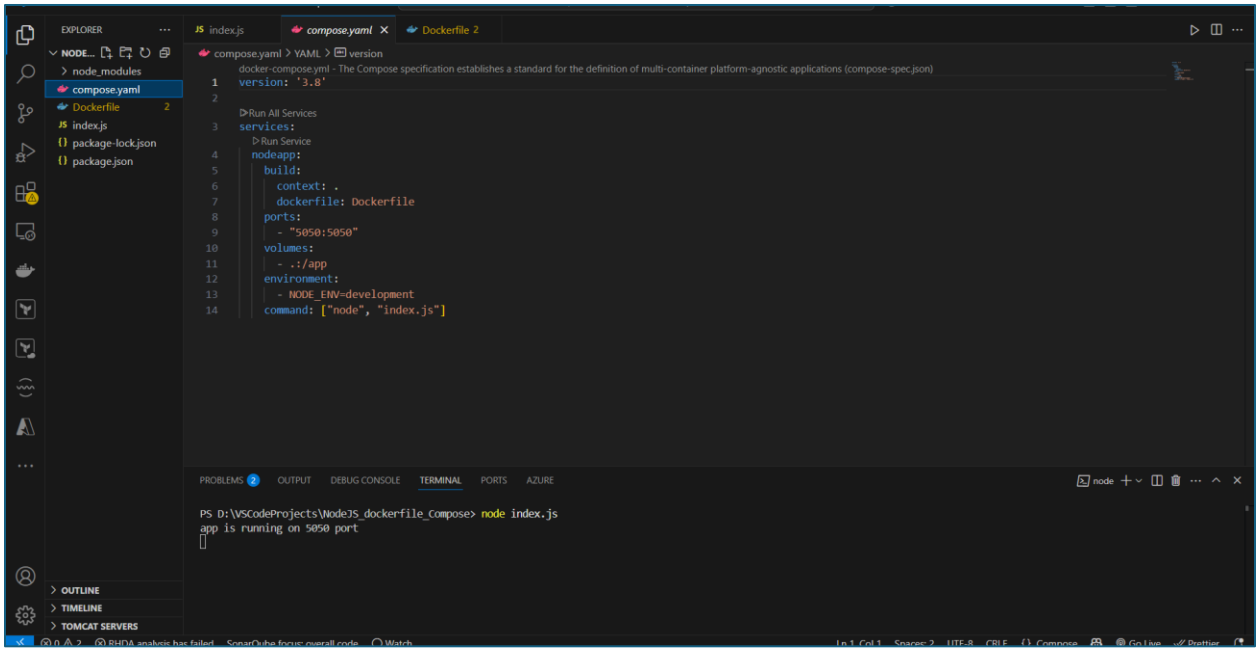
2.Dockerize a sample application (e.g., a web app or a microservice) by writing a Dockerfile.

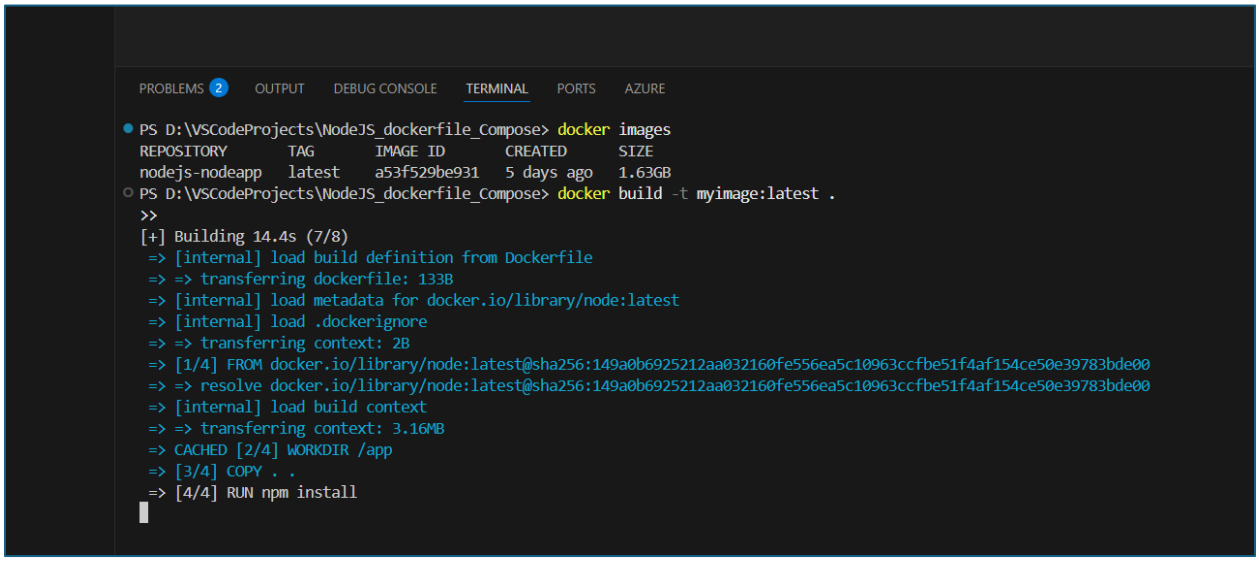




3.Build a Docker image from the Dockerfile.

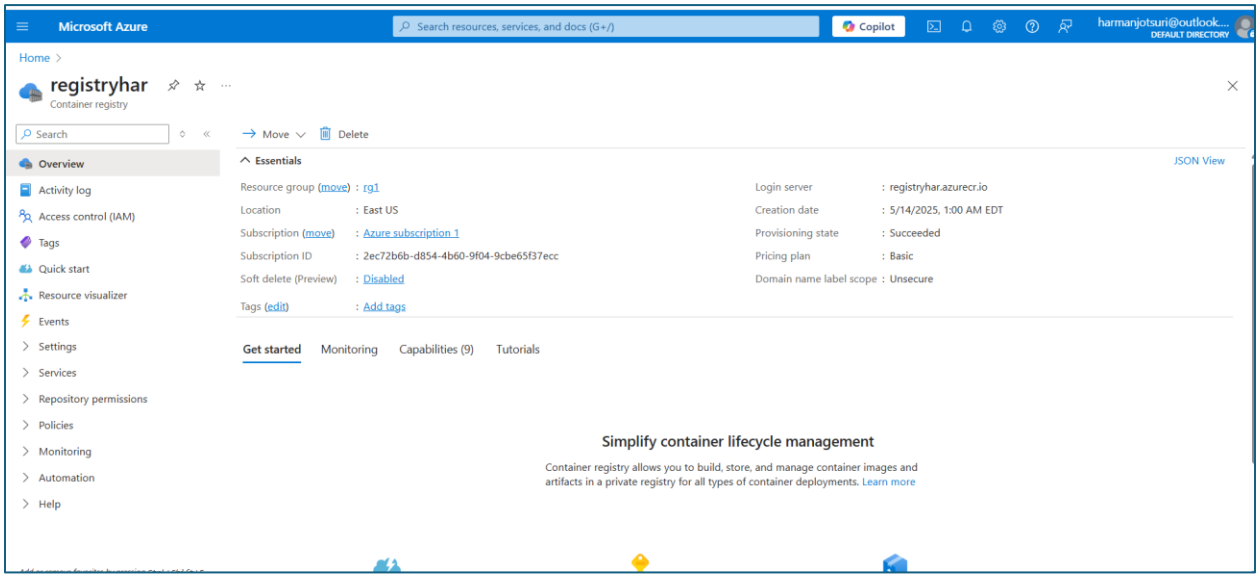



EXPLORER
NODE...
node_modules
compose.yaml
Dockerfile 2
index.js
package-lock.json
package.json
index.js
compose.yaml
Dockerfile 2
compose.yaml > YAML > version
docker-compose.yml - The Compose specification establishes a standard for the definition of multi-container platform-agnostic applications (compose-spec.json)
version: '3.8'
Run All Services
services:
Run Service
nodeapp:
build:
context: .
dockerfile: Dockerfile
ports:
- "5050:5050"
volumes:
- .:/app
environment:
- NODE_ENV=development
command: ["node", "index.js"]
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
AZURE
node
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> node index.js
app is running on 5050 port
OUTLINE
TIMELINE
TOMCAT SERVERS

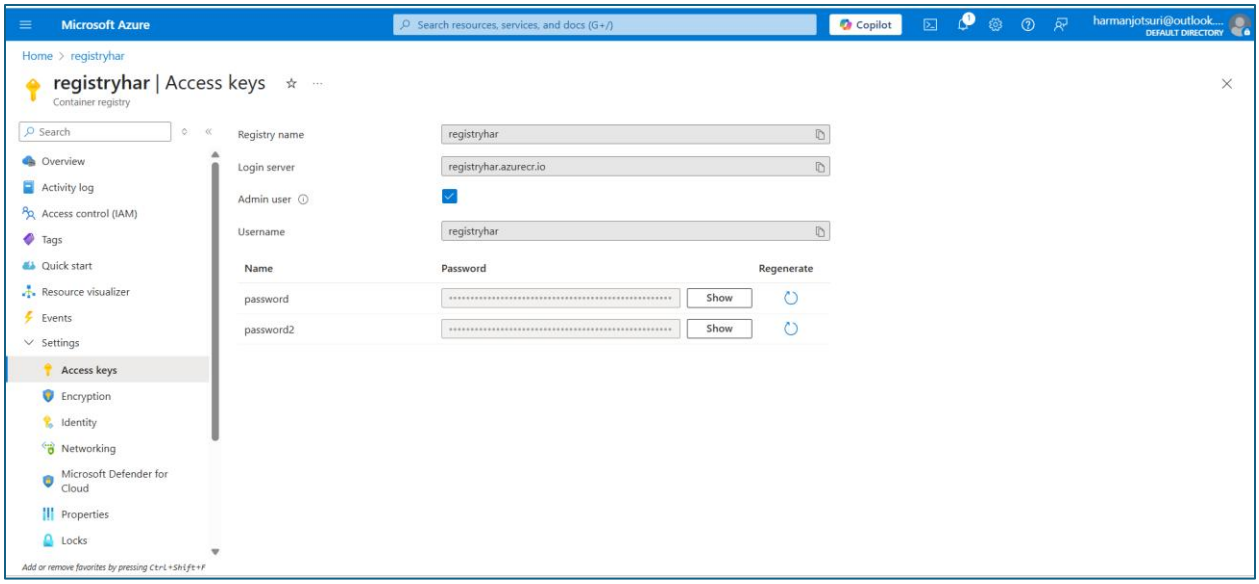

PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
AZURE
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
nodejs-nodeapp latest a53f529be931 5 days ago 1.63GB
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> docker build -t myimage:latest .
>>
[+] Building 14.4s (7/8)
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 133B
=> [internal] load metadata for docker.io/library/node:latest
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/4] FROM docker.io/library/node:latest@sha256:149a0b6925212aa032160fe556ea5c10963ccfbe51f4af154ce50e39783bde00
=> => resolve docker.io/library/node:latest@sha256:149a0b6925212aa032160fe556ea5c10963ccfbe51f4af154ce50e39783bde00
=> [internal] load build context
=> => transferring context: 3.16MB
=> CACHED [2/4] WORKDIR /app
=> [3/4] COPY . .
=> [4/4] RUN npm install

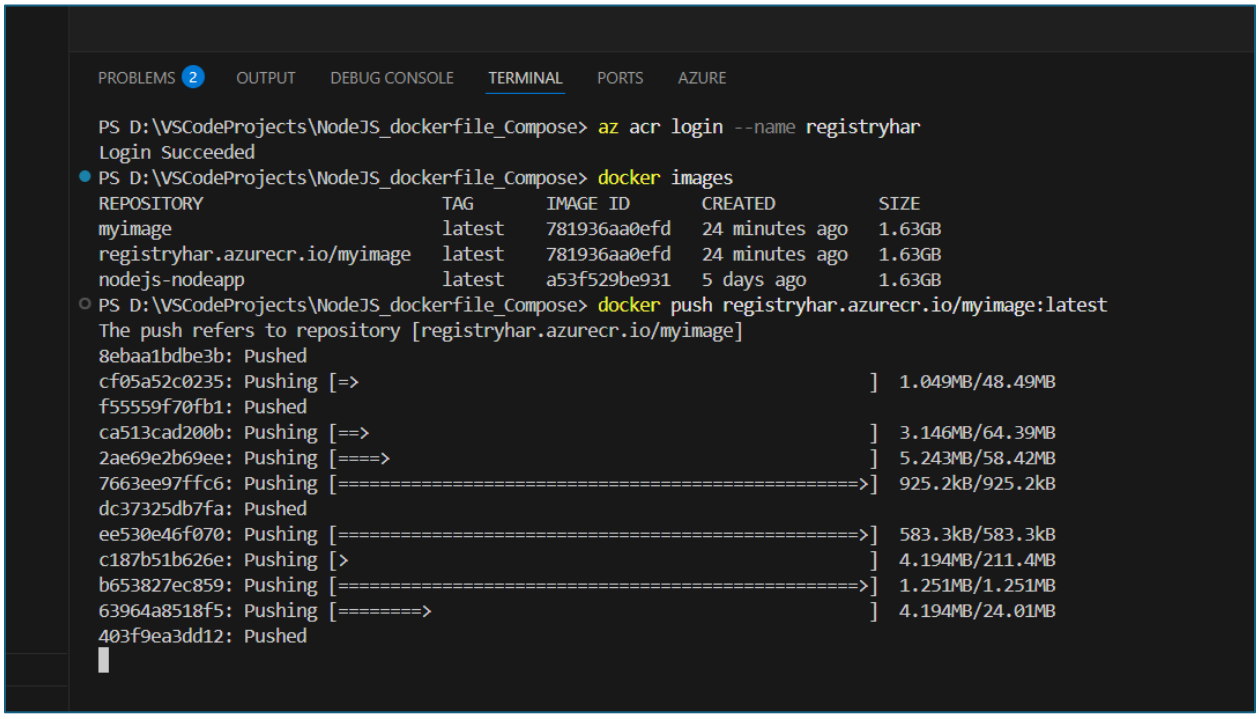
```
PROBLEMS 2   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS   AZURE

PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> docker images
REPOSITORY      TAG       IMAGE ID       CREATED          SIZE
myimage         latest    781936aa0efd   27 seconds ago   1.63GB
nodejs-nodeapp  latest    a53f529be931   5 days ago       1.63GB
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose>
```

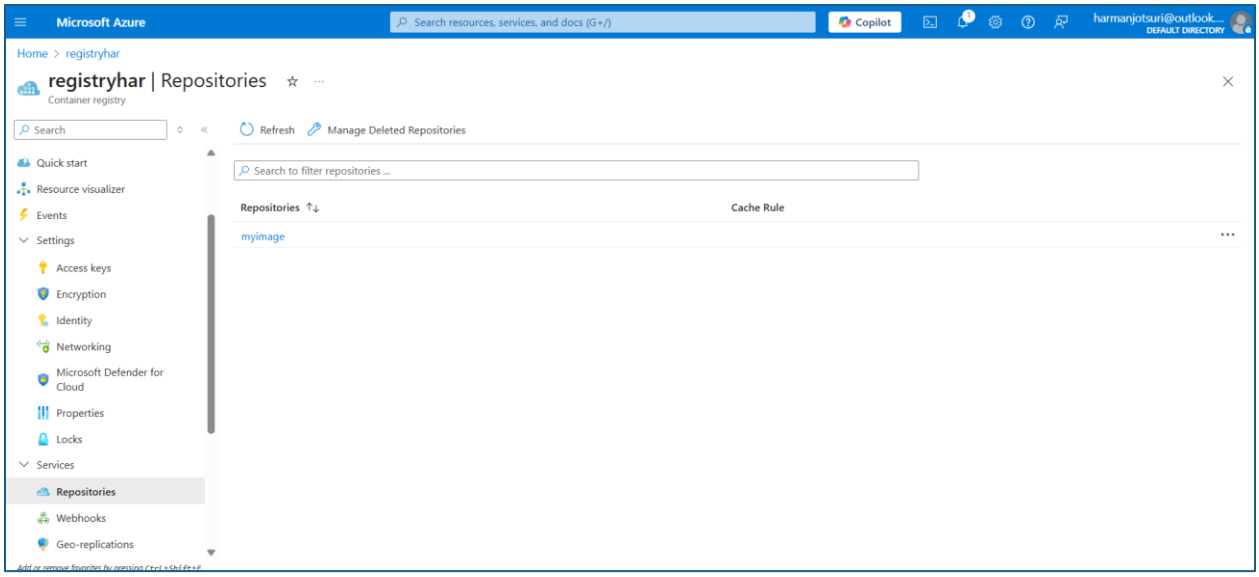
4.Push the Docker image to a container registry (e.g., Azure Container Registry).




```
PROBLEMS 2   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS   AZURE

PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> docker images
REPOSITORY      TAG       IMAGE ID       CREATED          SIZE
myimage         latest    781936aa0efd   27 seconds ago   1.63GB
nodejs-nodeapp  latest    a53f529be931   5 days ago       1.63GB
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> docker tag myimage registryhar.azurecr.io/myimage:latest
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> docker images
REPOSITORY                      TAG       IMAGE ID       CREATED          SIZE
myimage                         latest    781936aa0efd   18 minutes ago   1.63GB
registryhar.azurecr.io/myimage  latest    781936aa0efd   18 minutes ago   1.63GB
nodejs-nodeapp                  latest    a53f529be931   5 days ago       1.63GB
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose>
```


Microsoft Azure
Search resources, services, and docs (G+/)
Copilot
harmanjotsuri@outlook....
DEFAULT DIRECTORY
Home > registryhar
registryhar | Access keys
Container registry
Search
Overview
Activity log
Access control (IAM)
Tags
Quick start
Resource visualizer
Events
Settings
Access keys
Encryption
Identity
Networking
Microsoft Defender for Cloud
Properties
Locks
Add or remove favorites by pressing Ctrl+Shift+F
Registry name
registryhar
Login server
registryhar.azurecr.io
Admin user
Username
registryhar
Name
Password
Regenerate
password
Show
password2
Show


PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
AZURE
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> az acr login --name registryhar
Login Succeeded
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
myimage latest 781936aa0efd 24 minutes ago 1.63GB
registryhar.azurecr.io/myimage latest 781936aa0efd 24 minutes ago 1.63GB
nodejs-nodeapp latest a53f529be931 5 days ago 1.63GB
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> docker push registryhar.azurecr.io/myimage:latest
The push refers to repository [registryhar.azurecr.io/myimage]
8ebaa1bdbe3b: Pushed
cf05a52c0235: Pushing [=>] 1.049MB/48.49MB
f55559f70fb1: Pushed
ca513cad200b: Pushing [==>] 3.146MB/64.39MB
2ae69e2b69ee: Pushing [====>] 5.243MB/58.42MB
7663ee97ffc6: Pushing [==========>] 925.2kB/925.2kB
dc37325db7fa: Pushed
ee530e46f070: Pushing [==========>] 583.3kB/583.3kB
c187b51b626e: Pushing [>] 4.194MB/211.4MB
b653827ec859: Pushing [==========>] 1.251MB/1.251MB
63964a8518f5: Pushing [========>] 4.194MB/24.01MB
403f9ea3dd12: Pushed



5.Create a Kubernetes deployment file (YAML) for the application.

To let Kubernetes pull from ACR, you'll need to create an image pull secret using your Azure credentials:

kubectl create secret docker-registry acr-secret \

--docker-server=<your-registry-name>.azurecr.io \

--docker-username=<your-acr-username> \

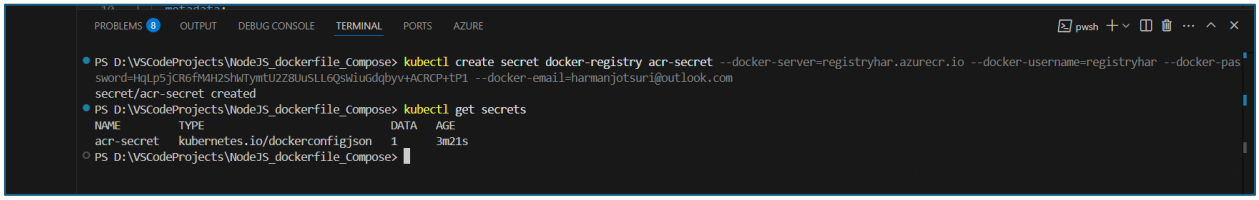
--docker-password=<your-acr-password> \

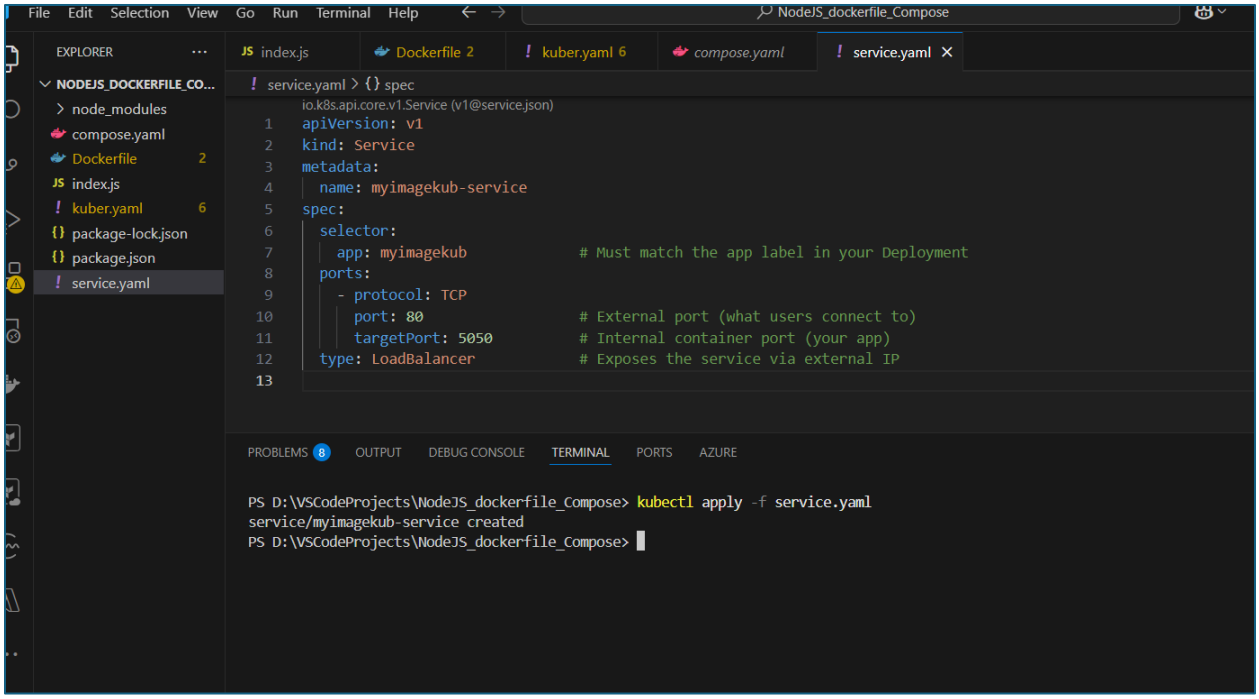
--docker-email=<your-email>

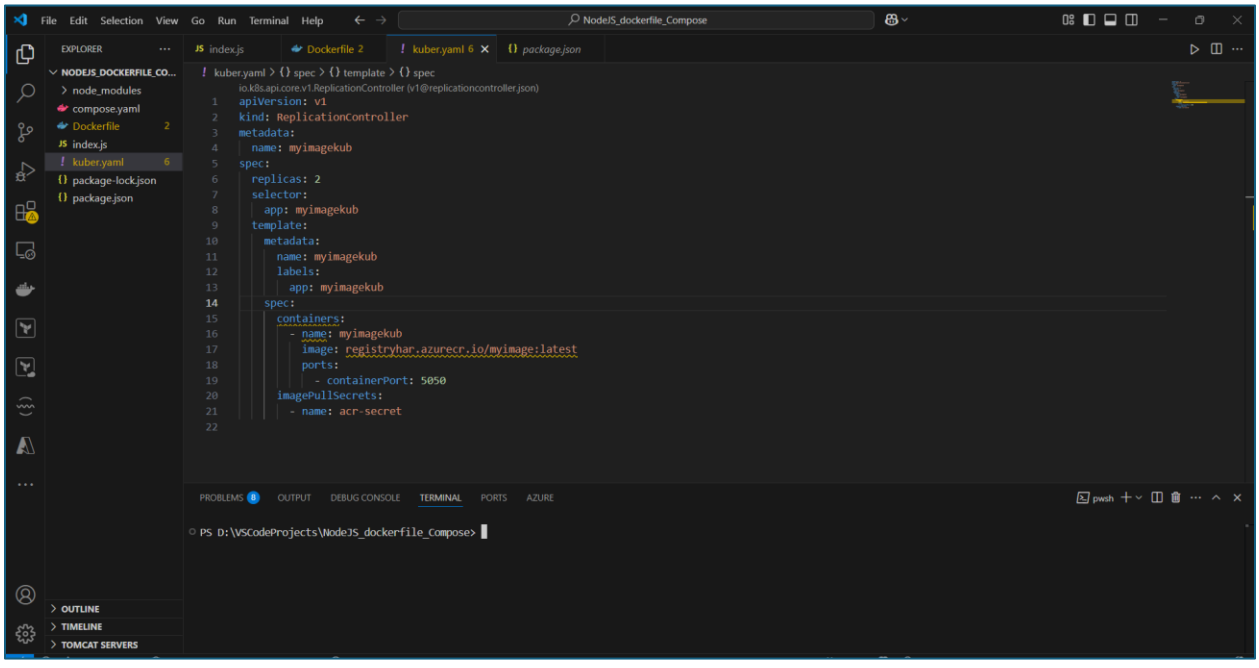
ERROR:

1. exactly one NAME is required, got 2 — means PowerShell is interpreting the line incorrectly, usually due to improper quotation or line continuation syntax.

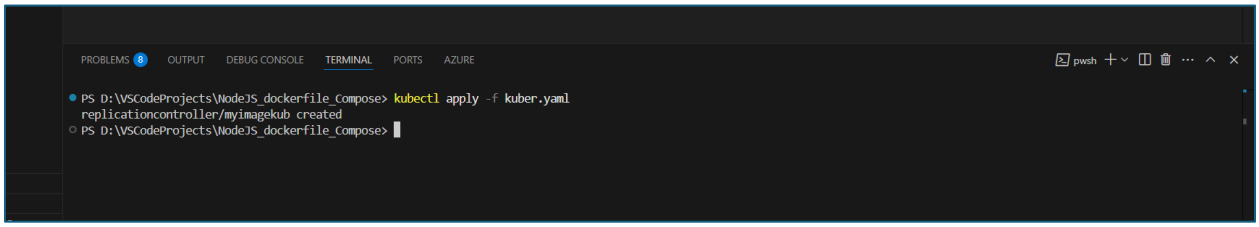
2. | Missing expression after unary operator '--'.



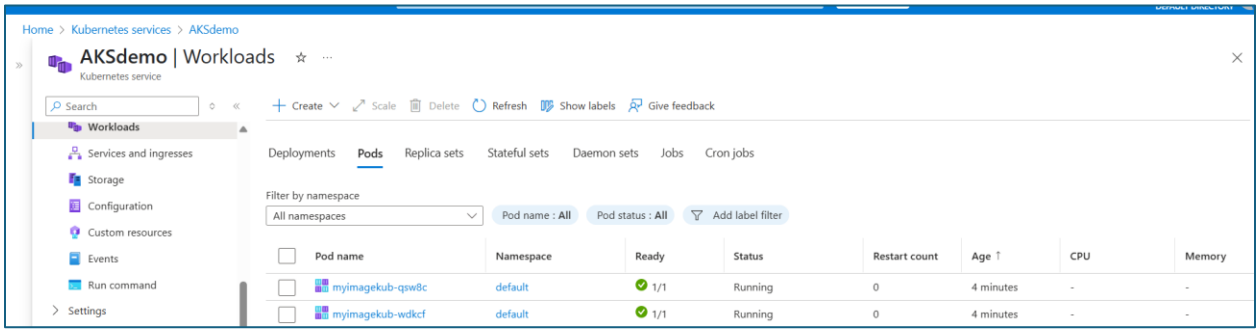




6.Deploy the application on the AKS cluster using the deployment file.

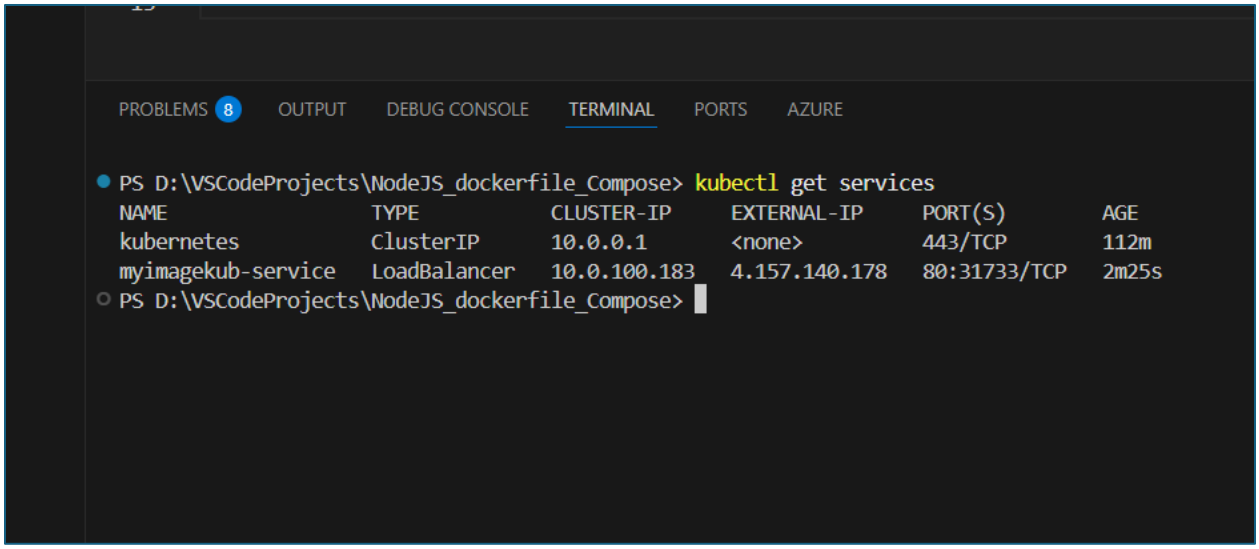


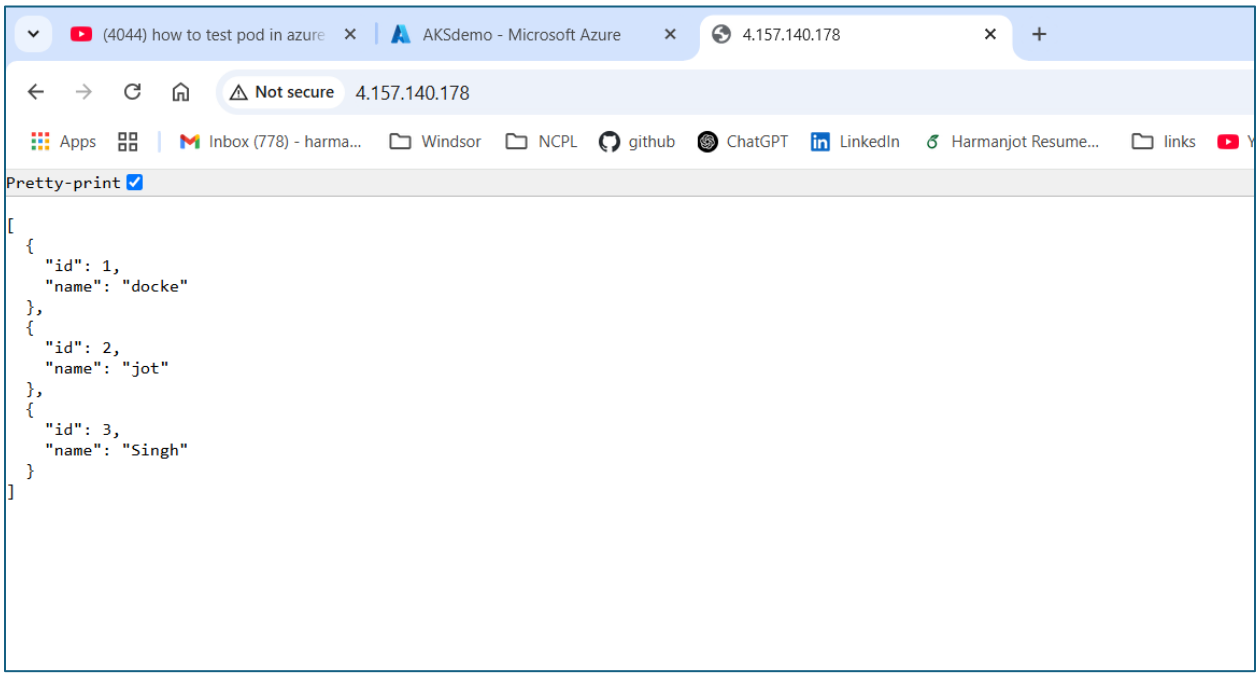
```
21        - name: acr-secret

PROBLEMS 8  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  AZURE

PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> kubectl apply -f kuber.yaml
replicationcontroller/myimagekub created
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> kubectl get pods
NAME              READY   STATUS    RESTARTS   AGE
myimagekub-qsw8c  1/1     Running   0          2m11s
myimagekub-wdkcf  1/1     Running   0          2m11s
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose>
```



7.Test the deployed application to ensure its functioning correctly.

```
22

PROBLEMS 8  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  AZURE

PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> kubectl get pods
NAME              READY   STATUS    RESTARTS   AGE
myimagekub-qsw8c  1/1     Running   0          8m49s
myimagekub-wdkcf  1/1     Running   0          8m49s
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> kubectl logs myimagekub
error: error from server (NotFound): pods "myimagekub" not found in namespace "default"
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> kubectl logs myimagekub-qsw8c
app is running on 5050 port
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> kubectl logs myimagekub-wdkcf
app is running on 5050 port
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose>
```





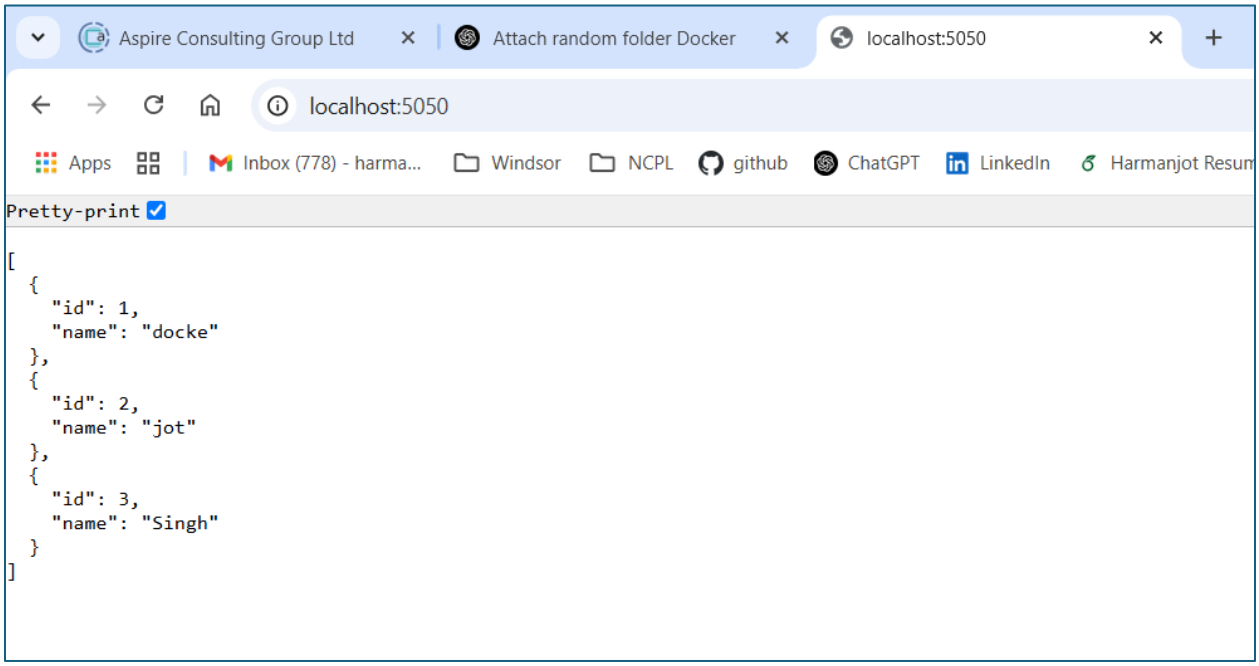
2nd way:

kubectl get pods

kubectl port-forward pod/myimagekub-7f8c7c9b4b-2v5z9 5050:5050

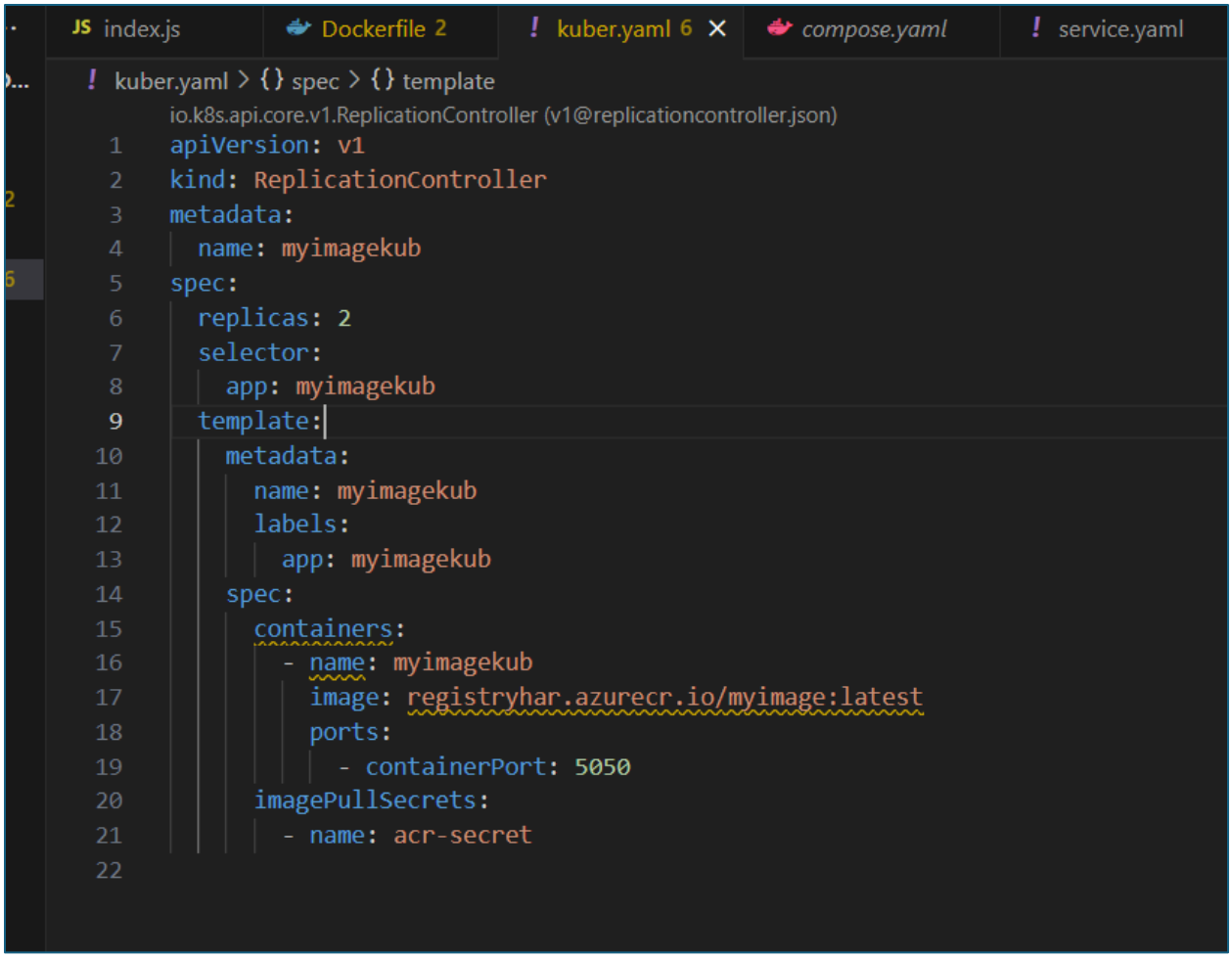
http://localhost:5050

```
NAME                 TYPE           CLUSTER-IP     EXTERNAL-IP     PORT(S)          AGE
kubernetes           ClusterIP      10.0.0.1       <none>          443/TCP          112m
myimagekub-service   LoadBalancer   10.0.100.183   4.157.140.178   80:31733/TCP     2m25s
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> kubectl get pods
NAME                 READY   STATUS    RESTARTS   AGE
myimagekub-qsw8c     1/1     Running   0          53m
myimagekub-wdkcf     1/1     Running   0          53m
PS D:\VSCodeProjects\NodeJS_dockerfile_Compose> kubectl port-forward pod/myimagekub-qsw8c 5050:5050
Forwarding from 127.0.0.1:5050 -> 5050
Forwarding from [::1]:5050 -> 5050
Handling connection for 5050
Handling connection for 5050
Handling connection for 5050
```



Note: Once we stop the command(kubectl port-forward pod/myimagekub-7f8c7c9b4b-2v5z9 5050:5050) that we can not access the application locally.

8.Scale the application by adjusting the replica count in the deployment file.



Document the entire process, including Dockerizing the application, deploying it on AKS, and scaling it.

Share the deployed application's endpoint and the documentation

Deliverables:

- Deployed application's endpoint
- Documentation describing the Dockerization and deployment process