**Maven home: C:\Users\niles\.m2\wrapper\dists\apache-maven-3.9.4-bin\32a55694\apache-maven-3.9.4**

<server>

<id>registry.hub.docker.com</id>

<username>XXXX </username>

<password>XXXXX@91</password>

</server>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.1</version>

<configuration>

<source>17</source>

<target>17</target>

<fork>true</fork>

<executable>C:\Program Files\Eclipse Adoptium\jdk-17.0.8.7-hotspot\bin\javac</executable>

</configuration>

</plugin>

<plugin>

<groupId>com.google.cloud.tools</groupId>

<artifactId>jib-maven-plugin</artifactId>

<version>3.4.0</version>

<configuration>

<from>

<image>openjdk:17-alpine</image>

</from>

<to>

<image>registry.hub.docker.com/nileshzarkar/${project.artifactId}:${project.version}</image>

</to>

<container>

<ports>

<port>8060</port>

</ports>

</container>

</configuration>

<executions>

<execution>

<id>build-and-push-docker-image</id>

<phase>package</phase>

<goals>

<goal>build</goal>

</goals>

</execution>

</executions>

</plugin>

<groupId>com.example</groupId>

<artifactId>service-registry</artifactId>

**<version>1.0</version>**

<name>service-registry</name>

<description>Demo project for Spring Boot</description>

NOTE: Make same changes for rest of the services  
  
\github\sb-user-department\service-registry>docker logout

\github\sb-user-department\service-registry>**mvnw.cmd clean install**

\github\sb-user-department\config-server>docker logout

\github\sb-user-department\config-server>**mvnw.cmd clean install**

\github\sb-user-department\department-service>docker logout

\github\sb-user-department\department-service>**mvnw.cmd clean install**

\github\sb-user-department\employee-service>docker logout

\github\sb-user-department\employee-service>**mvnw.cmd clean install**

\github\sb-user-department\api-gateway>docker logout

\github\sb-user-department\api-gateway>**mvnw.cmd clean install**

**Updated yaml files of services for Kubernetes Deployment  
api-gateway/src/main/resources/application.yml**server:

port: 8060

spring:

application:

name: api-gateway

cloud:

config:

enabled: false

gateway:

routes:

- id: employee-service

uri: lb://employee-service

predicates:

- Path=/employee/\*\*

- id: department-service

uri: lb://department-service

predicates:

- Path=/department/\*\*

eureka:

instance:

preferIpAddress: true

hostname: eureka-0

client:

registerWithEureka: true

fetchRegistry: true

serviceUrl:

defaultZone: http://eureka-0.eureka.default.svc.cluster.local:8761/eureka

management:

tracing:

sampling:

probability: 1.0

**config-server/src/main/resources/application.yml**server:

port: 8088

# spring:

# profiles:

# active: native

# cloud:

# config:

# server:

# bootstrap: true

eureka:

instance:

preferIpAddress: true

hostname: eureka-0

client:

register-with-eureka: true

fetch-registry: true

serviceUrl:

defaultZone: http://eureka-0.eureka.default.svc.cluster.local:8761/eureka/

spring:

application:

name: config-server

profiles:

active: git

cloud:

config:

server:

git:

uri: https://github.com/nileshzarkar/config.git

clone-on-start: true

default-label: main

**department-service/src/main/resources/application.yml**server:

port: 8001

spring:

application:

name: department-service

cloud:

config:

enabled: false

# config:

# import: optional:configserver:http://localhost:8088/

eureka:

instance:

preferIpAddress: true

hostname: eureka-0

client:

registerWithEureka: true

fetchRegistry: true

serviceUrl:

defaultZone: http://eureka-0.eureka.default.svc.cluster.local:8761/eureka

management:

tracing:

sampling:

probability: 1.0

**employee-service/src/main/resources/application.yml**server:

port: 8002

spring:

application:

name: employee-service

cloud:

config:

enabled: false

# config:

# import: optional:configserver:http://localhost:8088/

eureka:

instance:

preferIpAddress: true

hostname: eureka-0

client:

registerWithEureka: true

fetchRegistry: true

serviceUrl:

defaultZone: http://eureka-0.eureka.default.svc.cluster.local:8761/eureka

management:

tracing:

sampling:

probability: 1.0

\github\sb-user-department\k8s> **minikube start --driver=hyperv**

\github\sb-user-department\k8s> kubectl config use-context minikube

\github\sb-user-department\k8s> minikube status

\github\sb-user-department\k8s> minikube dashboard

\github\sb-user-department\k8s> minikube ip

172.26.100.124

\github\sb-user-department\k8s> ping 172.26.100.124

**k8s\service-registry.yml**

apiVersion: v1

kind: ConfigMap

metadata:

name: eureka-cm

data:

eureka\_service\_address: http://eureka-0.eureka:8761/eureka

---

apiVersion: v1

kind: Service

metadata:

name: eureka-lb

labels:

app: eureka

spec:

selector:

app: eureka

type: NodePort

ports:

- port: 80

targetPort: 8761

---

apiVersion: v1

kind: Service

metadata:

name: eureka

labels:

app: eureka

spec:

clusterIP: None

ports:

- port: 8761

name: eureka

selector:

app: eureka

---

apiVersion: apps/v1

kind: StatefulSet

metadata:

name: eureka

spec:

serviceName: "eureka"

replicas: 1

selector:

matchLabels:

app: eureka

template:

metadata:

labels:

app: eureka

spec:

containers:

- name: eureka

image: nileshzarkar/service-registry:1.0

imagePullPolicy: Always

ports:

- containerPort: 8761

env:

- name: EUREKA\_SERVER\_ADDRESS

valueFrom:

configMapKeyRef:

name: eureka-cm

key: eureka\_service\_address

**k8s\config-server.yml (SKIPPED IN K8S DEPLOYMENT)**

apiVersion: apps/v1

kind: Deployment

metadata:

name: cloud-config-server-app

labels:

app: cloud-config-server-app

spec:

replicas: 1

selector:

matchLabels:

app: cloud-config-server-app

template:

metadata:

labels:

app: cloud-config-server-app

spec:

containers:

- name: cloud-config-server-app

image: nileshzarkar/config-server:1.0

imagePullPolicy: Always

ports:

- containerPort: 8088

---

apiVersion: v1

kind: Service

metadata:

name: cloud-config-server-svc

spec:

ports:

- targetPort: 8088

port: 80

selector:

app: cloud-config-server-app

**k8s\department-service-deploy.yml**apiVersion: apps/v1

kind: Deployment

metadata:

name: department-service-app

labels:

app: department-service-app

spec:

replicas: 1

selector:

matchLabels:

app: department-service-app

template:

metadata:

labels:

app: department-service-app

spec:

containers:

- name: department-service-app

image: nileshzarkar/department-service:1.0

imagePullPolicy: Always

ports:

- containerPort: 8001

---

apiVersion: v1

kind: Service

metadata:

name: department-service-svc

spec:

type: ClusterIP

ports:

- targetPort: 8001

port: 80

selector:

app: department-service-app

**k8s\employee-service-deploy.yml**apiVersion: apps/v1

kind: Deployment

metadata:

name: employee-service-app

labels:

app: employee-service-app

spec:

replicas: 1

selector:

matchLabels:

app: employee-service-app

template:

metadata:

labels:

app: employee-service-app

spec:

containers:

- name: employee-service-app

image: nileshzarkar/employee-service:1.0

imagePullPolicy: Always

ports:

- containerPort: 8002

---

apiVersion: v1

kind: Service

metadata:

name: employee-service-svc

spec:

ports:

- targetPort: 8002

port: 80

selector:

app: employee-service-app

**k8s\api-gateway.yml**apiVersion: apps/v1

kind: Deployment

metadata:

name: api-gateway-app

labels:

app: api-gateway-app

spec:

replicas: 1

selector:

matchLabels:

app: api-gateway-app

template:

metadata:

labels:

app: api-gateway-app

spec:

containers:

- name: api-gateway-app

image: nileshzarkar/api-gateway:1.0

imagePullPolicy: Always

ports:

- containerPort: 8060

---

apiVersion: v1

kind: Service

metadata:

name: api-gateway-svc

spec:

type: LoadBalancer

ports:

- port: 80

targetPort: 8060

protocol: TCP

selector:

app: api-gateway-app

**k8s\deploy.bat**

kubectl create -f .\service-registry.yml

kubectl create -f .\config-server.yml

kubectl create -f .\department-service-deploy.yml

kubectl create -f .\employee-service-deploy.yml

kubectl create -f .\api-gateway.yml

URL for eureka-service  
<http://172.26.100.124:32706/> 🡺 The port will be different check service in K8S

**k8s\delete.bat**

kubectl delete -f .\service-registry.yml

kubectl delete -f .\config-server.yml

kubectl delete -f .\department-service-deploy.yml

kubectl delete -f .\employee-service-deploy.yml

kubectl delete -f .\api-gateway.yml

***30932 – This is the API gateway exposed port***

POST <http://172.26.100.124:30932/department>  
{

    "id": 1,

    "name": "IT"

}

POST <http://172.26.100.124:30932/employee>  
{

    "id": 1,

    "departmentId": 2,

    "name": "XYZ",

    "age": 12,

    "position": "manager"

}

GET <http://172.26.100.124:30932/department/with-employees>