Tasks To Be Performed:

1. Use the previous deployment

2. Deploy a NGINX deployment of 3 replicas

3. Create a NGINX service of type ClusterIP

4. Create an ingress service/ Apache to Apache service/ NGINX to NGINX service

**Procedure: -**

**apiVersion**: apps/v1  
**kind**: Deployment  
**metadata**:  
 **name**: nginx-deployment  
 **labels**:  
 **app**: nginx  
**spec**:  
 **replicas**: 3  
 **selector**:  
 **matchLabels**:  
 **app**: nginx  
 **template**:  
 **metadata**:  
 **labels**:  
 **app**: nginx  
 **spec**:  
 **containers**:  
 - **name**: nginx  
 **image**: **nginx**:latest  
 **ports**:  
 - **containerPort**: 80

**apiVersion**: v1  
**kind**: Service  
**metadata**:  
 **name**: nginx-service  
**spec**:  
 **type**: ClusterIP  
 **selector**:  
 **app**: nginx  
 **ports**:  
 - **protocol**: TCP  
 **port**: 80  
 **targetPort**: 80  
 **nodePort**: 30007

<https://kubernetes.io/docs/concepts/services-networking/ingress/> more information about the ingress.

**apiVersion**: networking.k8s.io/v1  
**kind**: Ingress  
**metadata**:  
 **name**: my-ingress  
 **annotations**:  
 nginx.ingress.kubernetes.io/**rewrite-target**: /  
**spec**:  
# **ingressClassName**: nginx-example  
 **rules**:  
 - **host**: example.com  
 - **http**:  
 **paths**:  
 - **path**: /  
 **pathType**: Prefix  
 **backend**:  
 **service**:  
 **name**: my-service  
 **port**:  
 **number**: 80

apache.yaml

apiVersion: apps/v1  
kind: Deployment  
metadata:  
 name: apache-deployment  
 labels:  
 app: apache  
spec:  
 replicas: 2  
 selector:  
 matchLabels:  
 app: apache  
 template:  
 metadata:  
 labels:  
 app: apache  
 spec:  
 containers:  
 - name: apache  
 image: apache:latest  
 ports:  
 - containerPort: 81  
---  
apiVersion: v1  
kind: Service  
metadata:  
 name: apache-service  
spec:  
 type: NodePort  
 selector:  
 app: apache  
 ports:  
 - protocol: TCP  
 port: 80  
 targetPort: 80  
 nodePort: 30008