1. Create a `myapp.yml` file in your local environment with the following content.

```yaml

kind: Namespace   ##kaynak olarak namespace olusturuyoruz kendimiza ait bir sanal olusturuyoruz

apiVersion: v1     ##bu sanal alan belirtil,ediginde default olan namespace kullanilir.

metadata:

   name: my-namespace

   labels:

      app: container-info

---

apiVersion: v1

kind: Service  ##kaynak olarak servis olusuturuluyor

metadata:

   name: container-info-svc

   namespace: my-namespace   # bu servis my namespace alainin da olusacak demektir.

   labels:

      app: container-info

spec:

   type: LoadBalancer   ##servis tipi olarak Loadbalancer secilmistir.

   ports:                  ##bu servis tipinin olusabilmesi icin cloud provider destegi  olmasi gerekmektedir

      - protocol: TCP      ##bu ornektede cluster i AWS uzerinden olusturuldugu icin servis tipi bu secilir

        port: 3000          ##servin portudur.

        nodePort: 30300   ##yayin yapan port

        targetPort: 80    ##container portu

   selector:

      app: container-info  #deploymentte olusan podlarin taninacpi ksimdir template label ayni olacak

---                               ##yoksa podlar taninamaz/

apiVersion: apps/v1

kind: Deployment   ##kaynak olarak deployment olusturulacaktir.

metadata:

  name: container-info-deploy

  namespace: my-namespace

  labels:

    app: container-info

spec:

  replicas: 3    ##kac tane pod olusturmak istendigi sayi girilir.

  selector:

    matchLabels:    ## burdaki etiketle templateteki label etiketleri ayni olmak zorunde

      app: container-info

  template:

    metadata:

      labels:

        app: container-info  ##bu isimlerin ayni olmasi lazim servis kaynagindaki selector kismi ayniolacak.

    spec:

      containers:

      - name: container-info

        image: techprodevops348/hellotech

        ports:

        - containerPort: 80

```