Kubernetes

- Create deployments and services for two Pods:
  - One for the app version 1.0 from the last homework

notepad deployment-v1.yaml

cat deployment-v1.yaml

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> notepad .\deployment-v1.yaml
PS C:\Users\Ivan\Desktop\student-info-ngnix> cat .\deployment-v1.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: student-info-nginx-v1-deployment
spec:
  replicas: 2
  strategy:
    type: RollingUpdate
    rollingUpdate:
      maxSurge: 2
      maxUnavailable: 1
  selector:
    matchLabels:
      app: student-info-nginx
      version: "1.0"
  template:
    metadata:
      labels:
        app: student-info-nginx
        version: "1.0"
    spec:
      containers:
        - name: student-info-nginx-container
          image: ivkeex/student-info-nginx:1.0
          ports:
            - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  name: student-info-nginx-v1-service
spec:
  selector:
    app: student-info-nginx
    version: "1.0"
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

  - One for the app version 2.0 from the last homework

notepad deployment-v2.yaml

cat deployment-v2.yaml

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> notepad .\deployment-v2.yaml
PS C:\Users\Ivan\Desktop\student-info-ngnix> cat .\deployment-v2.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: student-info-nginx-v2-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: student-info-nginx
      version: "2.0"
  template:
    metadata:
      labels:
        app: student-info-nginx
        version: "2.0"
    spec:
      containers:
        - name: student-info-nginx-container
          image: ivkeex/student-info-nginx:2.0
          ports:
            - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  name: student-info-nginx-v2-service
spec:
  selector:
    app: student-info-nginx
    version: "2.0"
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

k3d cluster create mycluster –api-port 6443

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> k3d cluster create mycluster --api-port 6443
INFO[0000] Prep: Network
INFO[0000] Created network 'k3d-mycluster'
INFO[0000] Created image volume k3d-mycluster-images
INFO[0000] Starting new tools node...
INFO[0000] Starting node 'k3d-mycluster-tools'
INFO[0001] Creating node 'k3d-mycluster-server-0'
INFO[0001] Creating LoadBalancer 'k3d-mycluster-serverlb'
INFO[0001] Using the k3d-tools node to gather environment information
INFO[0001] Starting new tools node...
INFO[0001] Starting node 'k3d-mycluster-tools'
INFO[0002] Starting cluster 'mycluster'
INFO[0002] Starting servers...
INFO[0003] Starting node 'k3d-mycluster-server-0'
INFO[0006] All agents already running.
INFO[0006] Starting helpers...
INFO[0007] Starting node 'k3d-mycluster-serverlb'
INFO[0013] Injecting records for hostAliases (incl. host.k3d.internal) and for 3 network members into CoreDNS configmap...
INFO[0015] Cluster 'mycluster' created successfully!
                                                    INFO[0015] You can now use it like this:
                                                                                            kubectl cluster-info
```

kubectl config set-cluster k3d mycluster –server=https://localhost:6443

–insecure-skip-tls-verify=true

kubectl apply -f deployment-v1.yml

kubectl apply -f deployment-v1.yml

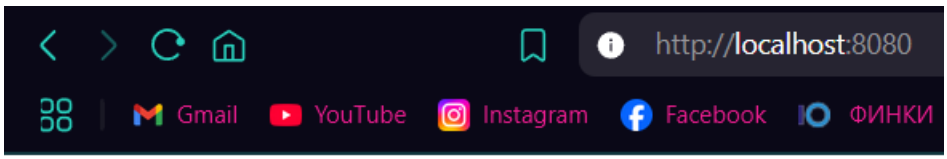
```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl config set-cluster k3d-mycluster --server=https://localhost:6443 --insecure-skip-tls-verify=true
>>
Cluster "k3d-mycluster" set.
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl apply -f deployment-v1.yaml
deployment.apps/student-info-nginx-v1-deployment created
service/student-info-nginx-v1-service created
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl apply -f deployment-v2.yaml
deployment.apps/student-info-nginx-v2-deployment created
service/student-info-nginx-v2-service created
```

kubectl get services

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl get services
NAME                            TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    AGE
kubernetes                      ClusterIP   10.43.0.1       <none>        443/TCP    3m53s
student-info-nginx-v1-service   ClusterIP   10.43.102.241   <none>        80/TCP     2m45s
student-info-nginx-v2-service   ClusterIP   10.43.36.131    <none>        80/TCP     2m35s
```

kubectl port-forwars service/student-info-nginx-v1-service 8080:80

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl port-forward service/student-info-nginx-v1-service 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
Handling connection for 8080
```



My Info

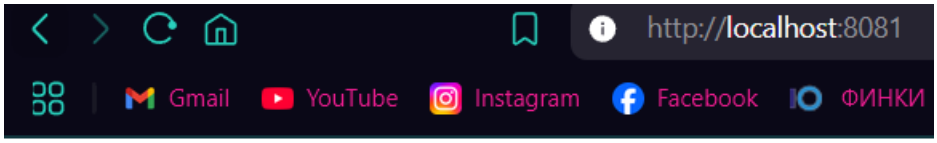
Name: Ivan

Surname: Pupinoski

Index: 223260

kubectl port-forwars service/student-info-nginx-v2-service 8081:80

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl port-forward service/student-info-nginx-v2-service 8081:80
>>
Forwarding from 127.0.0.1:8081 -> 80
Forwarding from [::1]:8081 -> 80
Handling connection for 8081
```



My Info

Name: Ivan

Surname: Pupinoski

Index: 223260

Year: 2025

- Installing NGINX Ingress Controller

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v1.9.6/deploy/static/provider/cloud/deploy.yaml
>>
namespace/ingress-nginx created
serviceaccount/ingress-nginx created
serviceaccount/ingress-nginx-admission created
role.rbac.authorization.k8s.io/ingress-nginx created
role.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrole.rbac.authorization.k8s.io/ingress-nginx created
clusterrole.rbac.authorization.k8s.io/ingress-nginx-admission created
rolebinding.rbac.authorization.k8s.io/ingress-nginx created
rolebinding.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx-admission created
configmap/ingress-nginx-controller created
service/ingress-nginx-controller created
service/ingress-nginx-controller-admission created
deployment.apps/ingress-nginx-controller created
job.batch/ingress-nginx-admission-create created
job.batch/ingress-nginx-admission-patch created
ingressclass.networking.k8s.io/nginx created
validatingwebhookconfiguration.admissionregistration.k8s.io/ingress-nginx-admission created
```

kubectl get pods -n ingress-nginx

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl get pods -n ingress-nginx
>>
NAME                                        READY   STATUS      RESTARTS   AGE
ingress-nginx-admission-create-pr55q        0/1     Completed   0          55s
ingress-nginx-admission-patch-8xrg4         0/1     Completed   1          55s
ingress-nginx-controller-576cdbb6db-pwvxb   1/1     Running     0          55s
```

- Create ingress pointing to the two apps:
  - path based: localhost/ver1
  - path based: localhost/ver2
  - host based: ver1..com
  - host based: ver2..com

notepad ingress.yaml

cat ingress.yaml

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> notepad ingress.yaml
PS C:\Users\Ivan\Desktop\student-info-ngnix> cat .\ingress.yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: student-info-ingress
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
  rules:
    - host: localhost
      http:
        paths:
          - path: /ver1
            pathType: Prefix
            backend:
              service:
                name: student-info-nginx-v1-service
                port:
                  number: 80
          - path: /ver2
            pathType: Prefix
            backend:
              service:
                name: student-info-nginx-v2-service
                port:
                  number: 80

    - host: ver1.223260.com
      http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: student-info-nginx-v1-service
                port:
                  number: 80

    - host: ver2.223260.com
      http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: student-info-nginx-v2-service
                port:
                  number: 80
```

- Deploy manifests and ingress

kubectl apply -f ingress.yaml

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl apply -f ingress.yaml
ingress.networking.k8s.io/student-info-ingress created
```

- Access the four ingress rules in your local browse

Editing hosts file

```
# Added by Docker Desktop
192.168.1.29 host.docker.internal
192.168.1.29 gateway.docker.internal
# To allow the same kube context to work on the host and the container:
127.0.0.1 kubernetes.docker.internal

127.0.0.1 ver1.223260.com
127.0.0.1 ver2.223260.com

# End of section
```

kubectl get pods-n ingress-nginx

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl get pods -n ingress-nginx
>>
NAME                                        READY   STATUS      RESTARTS   AGE
ingress-nginx-admission-create-pr55q        0/1     Completed   0          14m
ingress-nginx-admission-patch-8xrg4         0/1     Completed   1          14m
ingress-nginx-controller-576cdbb6db-pwvxb   1/1     Running     0          14m
```

kubectl get ing

```
PS C:\Users\Ivan\Desktop\student-info-ngnix> kubectl get ing
NAME                 CLASS     HOSTS                                    ADDRESS       PORTS   AGE
student-info-ingress   traefik   localhost,ver1.223260.com,ver2.223260.com   192.168.0.3   80      6m18s
```