Introduction

» Intro Application

HTML

Create an HTML document that provides your basic information, along with answers to the questions below. Feel free to customize the look and feel of the page to make yourself stand out. For example, include a photo of yourself!

```
cat > intro.html <<EOF
<html>
  <head><title>Hello K8s</title></head>
  <body>
    <h1>Your Name Here</h1>
    <h2>Your Company</h2>
    <h3>Your Role</h3>
    <p>Describe your experience with containers</p>
    <p>Describe your experience with Kubernetes</p>
    <p>What I hope to learn in this class:</p>
    <ul>
      <li>Tell us what you hope to learn here</li>
      <li>What else?</li>
      <li>Don't be shy, what else?</li>
    </ul>
  </body>
</html>
EOF
```

ConfigMap

Save your HTML content in Kubernetes.

```
kubectl create configmap nginx-config --from-literal index.html="`cat intro.html`"
```

View the HTML content that is saved in Kubernetes.

```
kubectl describe configmap nginx-config
```

Deployment & Service

Define a new deployment and service for the HTML content.

```
cat > intro-app.yaml <<EOF
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: intro-app
spec:
  replicas: 1
  template:
    metadata:
      labels:
        app: intro-app
    spec:
      containers:
        - name: intro-app
          image: nginx:1.13
          ports:
          - containerPort: 80
          volumeMounts:
          - mountPath: /usr/share/nginx/html
            name: config-volume
      volumes:
        - name: config-volume
          configMap:
            name: nginx-config
---
apiVersion: v1
kind: Service
metadata:
  name: intro-app
spec:
  selector:
    app: intro-app
  ports:
    - port: 80
      protocol: TCP
      targetPort: 80
EOF
```

Create the deployment and service for the HTML content.

```
kubectl create -f intro-app.yaml
```

Ingress

Save the hostname for the lab cluster into a shell variable for easy use. Remember to replace `labXX` with your actual lab number.

```
export CLUSTER=ford-labXX.coreostrain.me
echo $CLUSTER
```

Obtain one of the IP addresses for the lab cluster load balancer.

```
host $CLUSTER
```

Save an IP address returned into a shell variable.

```
export ELB_IP=`host $CLUSTER | awk 'NR==1{print $4}'`
echo $ELB_IP
```

Define a shell variable for the DNS name for the application.

```
export INTRO_HOST=$USER.$ELB_IP.xip.io
echo $INTRO_HOST
```

Define an ingress rule for the `intro-app` service.

```
cat > intro-ingress.yaml <<EOF
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: intro-app
  annotations:
    kubernetes.io/ingress.class: "tectonic"
spec:
  rules:
    - host: $INTRO_HOST
      http:
        paths:
          - path: /
            backend:
              serviceName: intro-app
              servicePort: 80
EOF
```

Create the ingress resource.

```
kubectl create -f intro-ingress.yaml
```

The HTML content should be available at:

```
echo http://$INTRO_HOST
```

After you are finished presenting, delete the objects.

```
kubectl delete ingress intro-app
kubectl delete service intro-app
kubectl delete deployment intro-app
kubectl delete configmap nginx-config
```