Practical Task 7: Configure and Use ConfigMaps and Secrets in AKS

Requirements:

- 1. Create a ConfigMap to store non-sensitive configuration data with only the required key-value pairs for the application.
- 2. Create a Kubernetes Secret to store sensitive data (e.g., API keys) with the least amount of information needed.
- 3. Update the application deployment to use the ConfigMap and Secret.
- 4. Remove the ConfigMap, Secret, and deployment after testing.

Practical Task 8: Scale Applications in AKS

Requirements:

- 1. Deploy a stateless application to the AKS cluster using minimal resource specifications.
- 2. Use the kubectl scale command to manually scale the application to only 2–3 replicas for testing.
- 3. Set up Horizontal Pod Autoscaler (HPA) with reasonable CPU usage thresholds to minimize pod creation.

Clients list

ID ^	Name ≑	Email \$	Shipping Address \$	Active 	Operations
1	Mercury	mercury@gmail.com	123 Mercury St, Solar City, SC 12345	true	Lock
2	Venus	venus@gmail.com	456 Venus Ave, Solar City, SC 12345	true	Lock
3	Earth	earth@gmail.com	789 Earth Blvd, Solar City, SC 12345	true	Lock
4	Mars	lamGod@gmail.com	101 Mars Rd, Solar City, SC 12345	true	Lock
5	Jupiter	jupiter@gmail.com	202 Jupiter Ln, Solar City, SC 12345	true	Lock



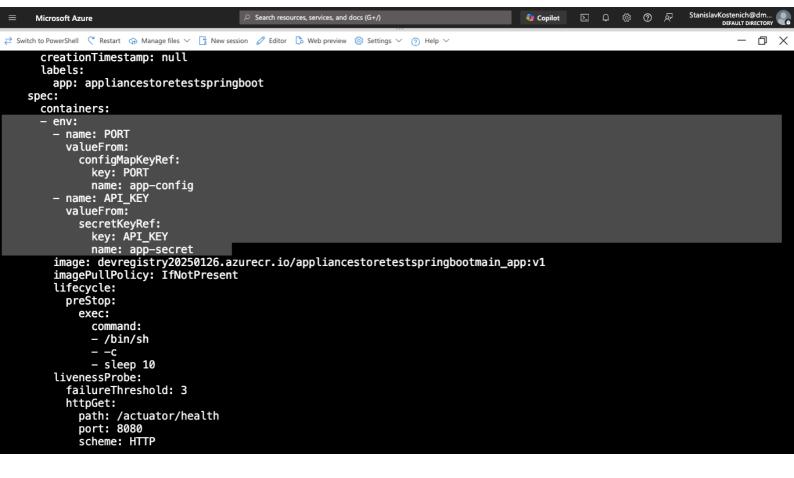
```
∠ Switch to PowerShell 
∠ Restart 
→ Manage files 
∠ 
→ New session 
∠ Editor 
→ Web preview 
→ Settings 
∠ 
→ Help 

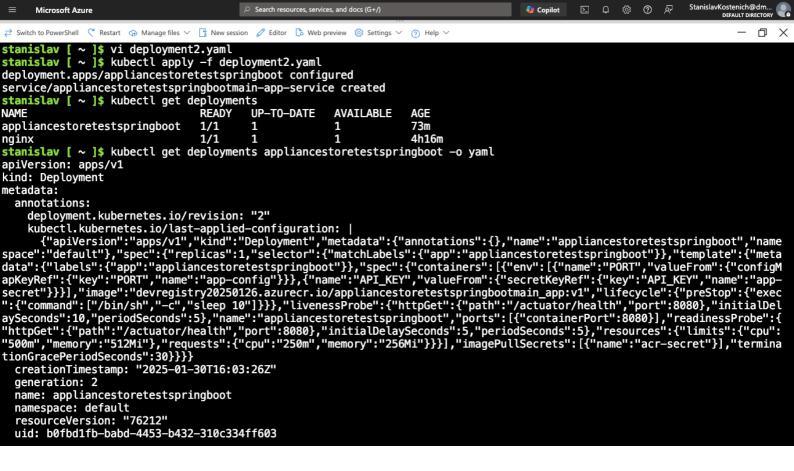
                                                                                                                                   ₽
* connect to 48.214.169.41 port 80 from 10.0.0.250 port 36058 failed: Connection timed out
* Failed to connect to 48.214.169.41 port 80 after 131561 ms: Couldn't connect to server
* Closing connection curl: (28) Failed to connect to 48.214.169.41 port 80 after 131561 ms: Couldn't connect to server
stanislav [ ~ ]$ kubectl get deployments
NAME
                                  READY
                                           UP-T0-DATE
                                                         AVAILABLE
                                                                       AGE
appliancestoretestspringboot
                                  1/1
                                                                       78m
                                                         1
                                                         1
                                                                       4h21m
nginx
                                  1/1
stanislav [ ~ ]$ vi deployment2.yaml
stanislav [ ~ ]$ kubectl apply -f deployment2.yaml
deployment.apps/appliancestoretestspringboot unchanged
service/appliancestoretestspringbootmain—app—service configured stanislav [ ~ ]$ kubectl get deployments
Name
                                  READY
                                           UP-TO-DATE
                                                         AVAILABLE
                                                                       AGE
appliancestoretestspringboot
                                  1/1
                                                                       81m
                                           1
                                                         1
nginx
                                  1/1
                                           1
                                                         1
                                                                       4h24m
stanislav [ ~ ]$ kubectl get services
NAME
                                                    TYPE
                                                                     CLUSTER-IP
                                                                                    EXTERNAL-IP
                                                                                                       PORT(S)
                                                                                                                        AGE
appliancestoretestspringboot-service
                                                    LoadBalancer
                                                                     10.0.26.214
                                                                                    52.230.159.186
                                                                                                       80:32301/TCP
                                                                                                                        81m
                                                                                                       80:32607/TCP
appliancestoretestspringbootmain-app-service
                                                    LoadBalancer
                                                                     10.0.63.221
                                                                                    48.214.169.41
                                                                                                                        8m52s
                                                                     10.0.0.1
10.0.151.67
kubernetes
                                                    ClusterIP
                                                                                     <none>
                                                                                                       443/TCP
                                                                                                                        5h18m
                                                    LoadBalancer
                                                                                    64.236.63.178
                                                                                                       80:31137/TCP
nginx
                                                                                                                        4h24m
stanislav [ ~ ]$ curl -v http://48.214.169.41
    Trying 48.214.169.41:80..
 Connected to 48.214.169.41 (48.214.169.41) port 80
  GET / HTTP/1.1
 Host: 48.214.169.41
  User-Agent: curl/8.8.0
 Accept: */*
* Request completely sent off
```

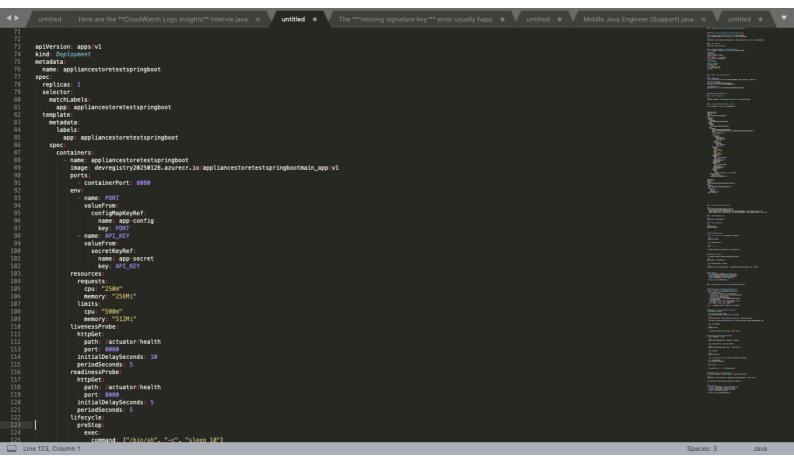
Copilot

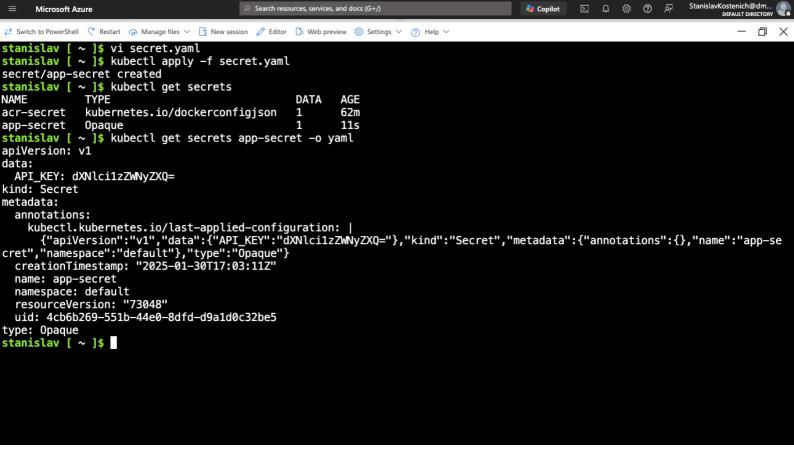
(

Microsoft Azure







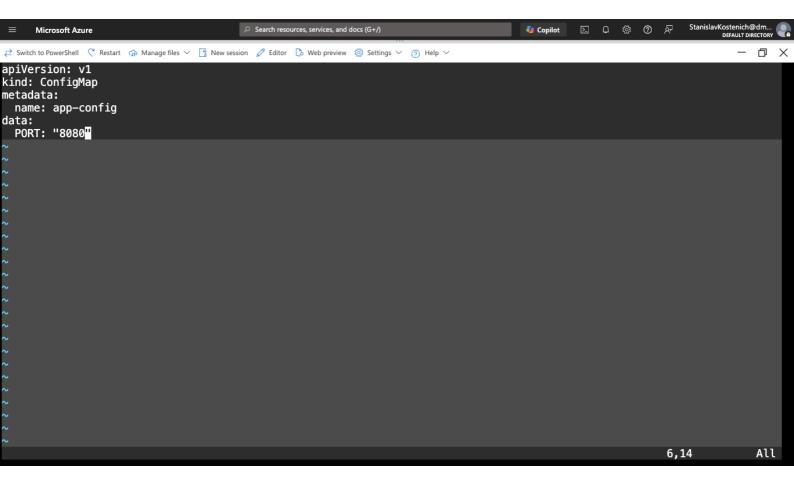


```
🔁 Switch to PowerShell 🦿 Restart 😘 Manage files 🗸 📑 New session 🧷 Editor 🖒 Web preview 🐯 Settings 🗸 👩 Help 🗸
                                                                                                                                                   stanislav [ ~ ]$ vi configmap.yaml
stanislav [ ~ ]$ kubectl apply -f configmap.yaml
configmap/app-config created
stanislav [ ~ ]$ kubectl get configmaps configmap -o yaml
Error from server (NotFound): configmaps "configmap" not found
stanislav [ ~ ]$ kubectl get configmaps
                                AGE
NAME
                       DATA
app-config
                                3m4s
kube-root-ca.crt 1 4h53m

stanislav [ ~ ]$ kubectl get configmaps app-config -o yaml
apiVersion: v1
data:
  PORT: "8080"
kind: ConfigMap
metadata:
  annotations:
    pace":"default"}}
  creationTimestamp: "2025-01-30T16:57:45Z"
  name: app-config
namespace: default
resourceVersion: "71732"
uid: 00f3714f-39ce-4817-8c58-047c8e774cb1 stanislav [ ~ ]$
```

Copilot

Microsoft Azure



Practical Task 7: Configure and Use ConfigMaps and Secrets in AKS

Requirements:

- 1. Create a ConfigMap to store non-sensitive configuration data with only the required key-value pairs for the application.
- 2. Create a Kubernetes Secret to store sensitive data (e.g., API keys) with the least amount of information needed.
- 3. Update the application deployment to use the ConfigMap and Secret.
- 4. Remove the ConfigMap, Secret, and deployment after testing.

Practical Task 8: Scale Applications in AKS

Requirements:

- 1. Deploy a stateless application to the AKS cluster using minimal resource specifications.
- 2. Use the kubectl scale command to manually scale the application to only 2–3 replicas for testing.
- 3. Set up Horizontal Pod Autoscaler (HPA) with reasonable CPU usage thresholds to minimize pod creation.