

Lesson 08 Demo 03

Connecting to an AKS Cluster from Azure Cloud Shell

Objective: To connect to an AKS cluster from Azure cloud shell to enhance the security and provide a consistent and accessible environment for AKS cluster related tasks

Tools required: Azure management tools

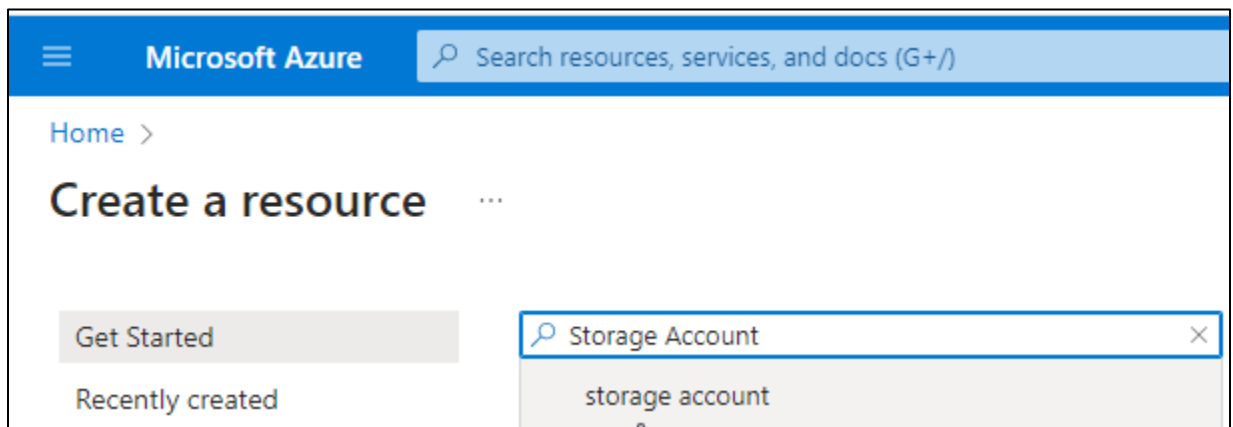
Prerequisites: An AKS cluster should already be set up (refer to the steps provided in Lesson 08, Demo 01 for guidance).

Steps to be followed:

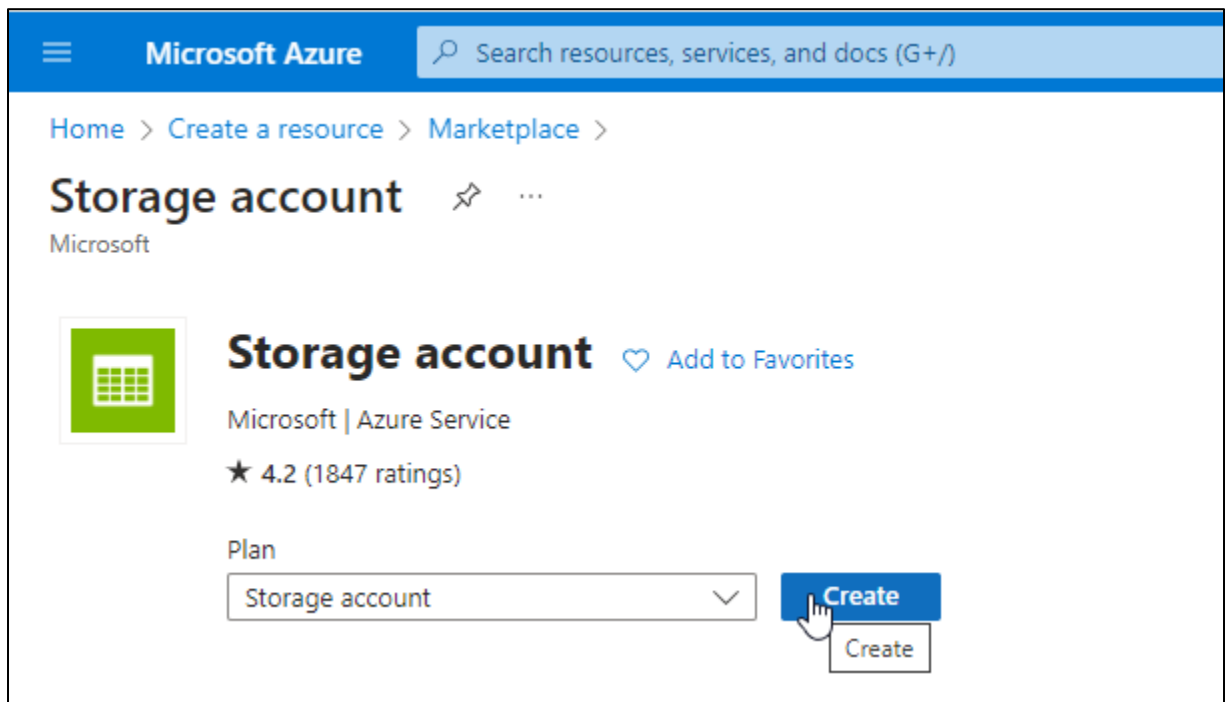
1. Create a storage account
2. Create a file share for the storage account
3. Set up the Azure Cloud Shell
4. Create a deployment and roll out an update for it

Step 1: Create a storage account

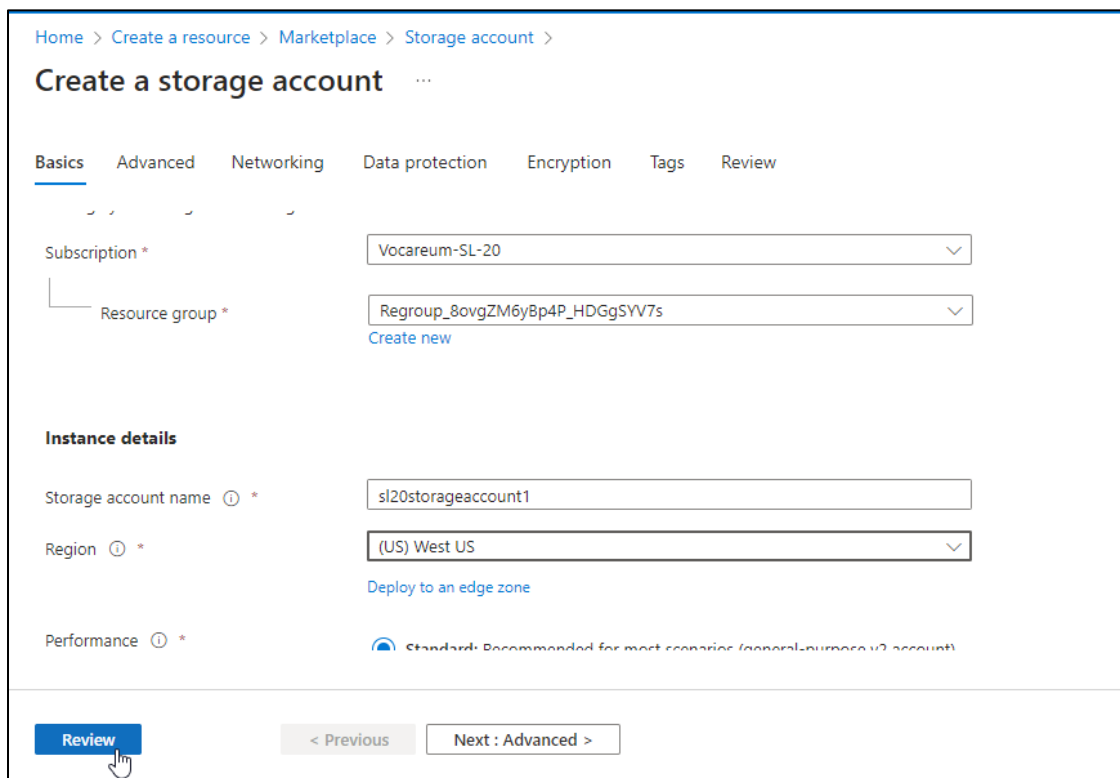
1.1 Navigate to the Azure portal, and search for and select **storage account**





1.2 On the storage account page, click on **Create**



1.3 Enter the details for the storage account, and click on **Review**



1.4 Review all the settings and click on **Create**

 **Microsoft Azure**  Search resources, services, and docs (G+/)

Home > Create a resource > Marketplace > Storage account >

Create a storage account ...

Basics Advanced Networking Data protection Encryption Tags Review

Basics

Subscription	Vocareum-SL-20
Resource Group	Regroup_8ovgZM6yBp4P_HDGgSYV7s
Location	westus
Storage account name	sl20storageaccount1
Deployment model	Resource manager
Performance	Standard
Replication	Read-access geo-redundant storage (RA-GRS)

Advanced

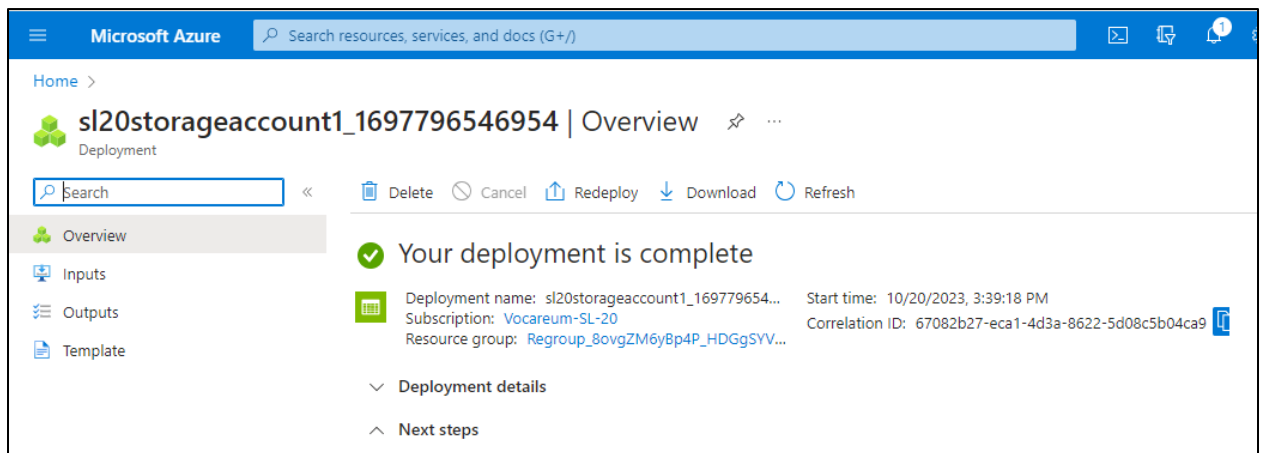
Enable hierarchical namespace	Disabled
Enable network file system v3	Disabled

Create

< Previous

Next >

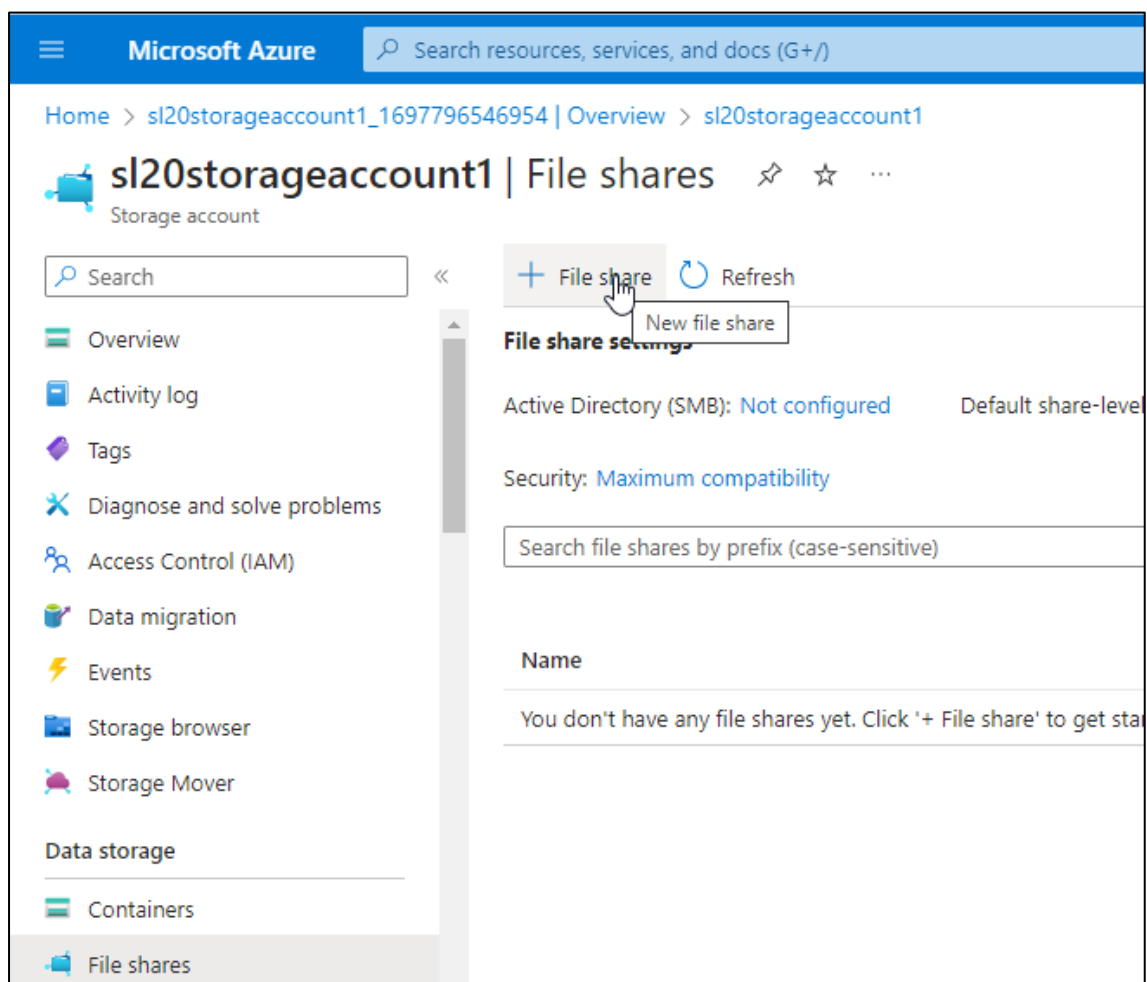
[Download a template for automation](#)



The storage account is successfully created.

Step 2: Create a file share for the storage account

2.1 Navigate to the storage account, click **File shares** under the Data storage section, and click on **+ File share**



2.2 Provide a name to the file share and click on **Review + create**

Microsoft Azure

Search resources, services, and docs (G+/)

[Home](#) > [sl20storageaccount1_1697796546954 | Overview](#) > [sl20storageaccount1 | File shares](#) >

New file share ...

Basics

Backup

Review + create

Name *


fileshare1

Tier *

Transaction optimized

Performance

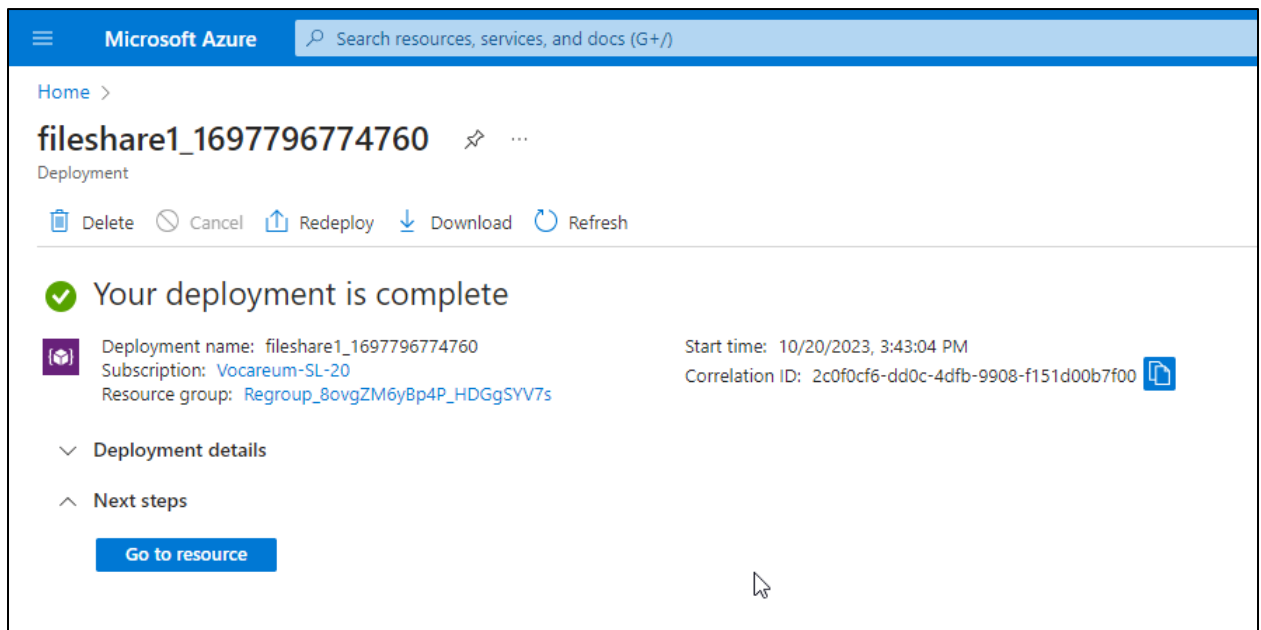
Maximum IO/s ⓘ	1000
Maximum capacity	5 TiB
Large file shares	Disabled

 You can improve performance and maximum share capacity by enabling large file shares for this storage account. [Learn](#)

Review + create

< Previous

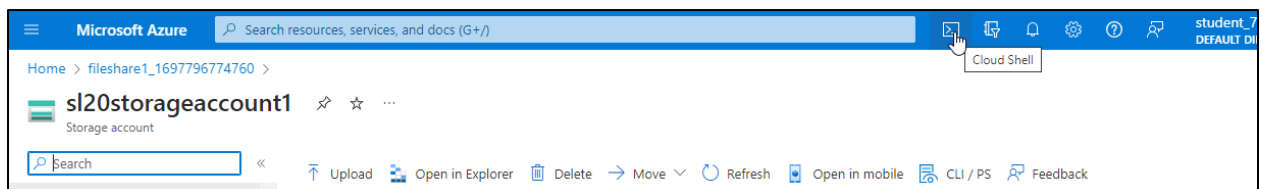
Next : Backup >



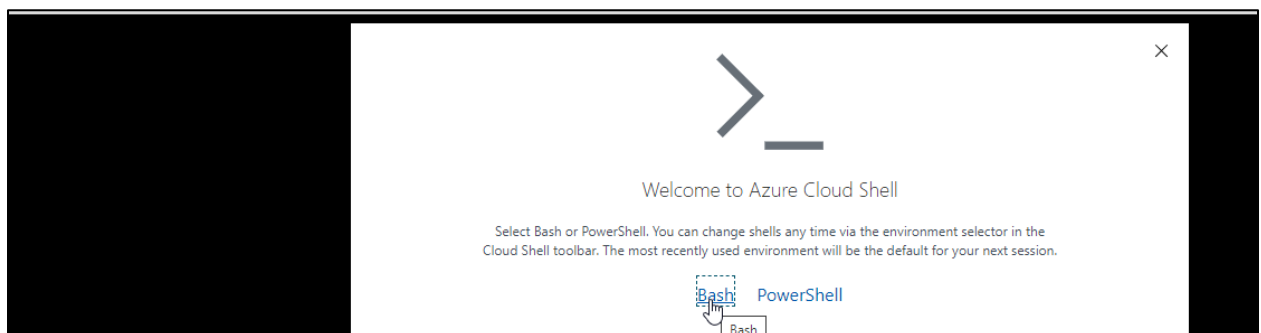
The file share is successfully created.

Step 3: Set up the Azure Cloud Shell

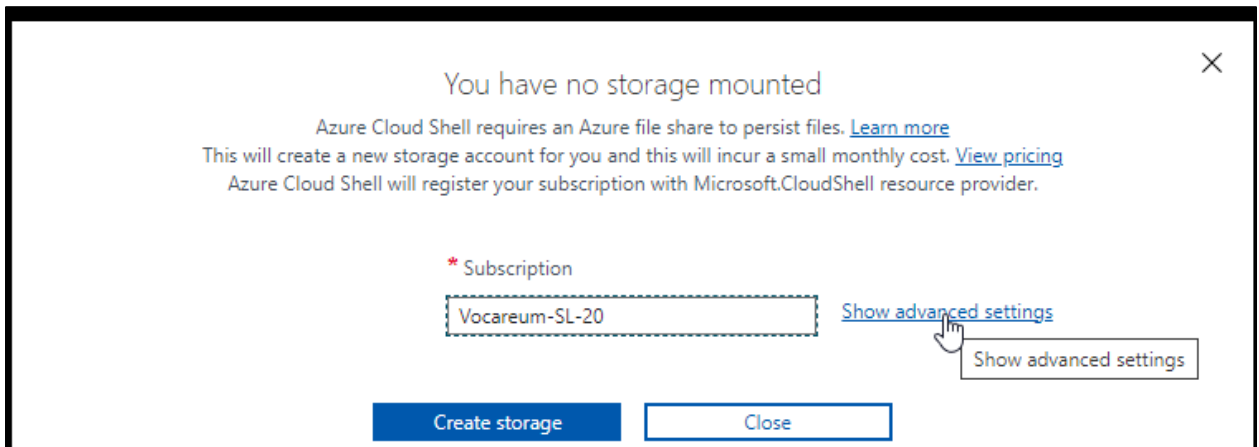
3.1 Click on the **Cloud Shell** icon from the top navigation bar



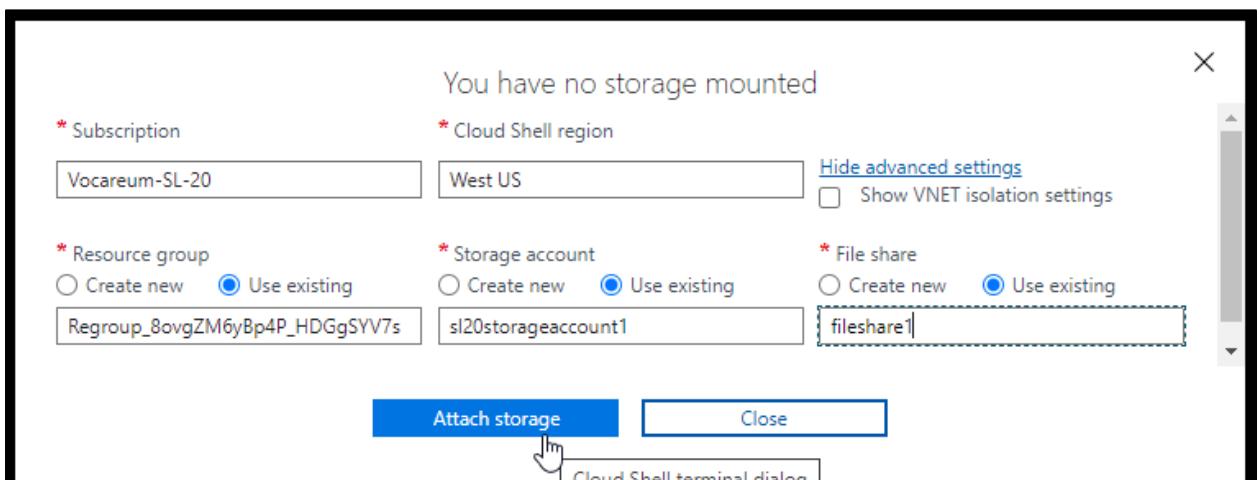
3.2 On the Azure Cloud Shell window, click on **Bash**



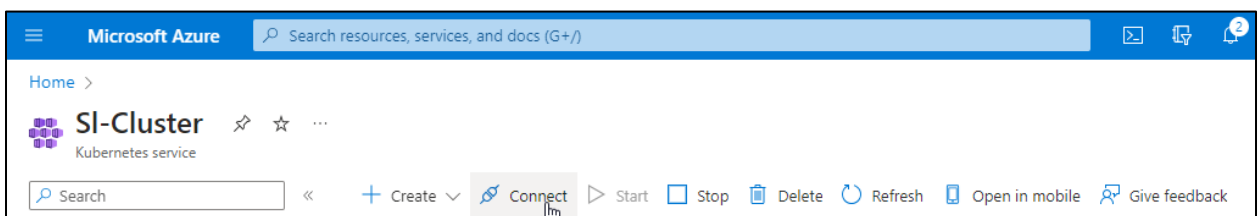
3.3 On the next screen, click on **Show advanced settings**



3.4 Fill in the required details as shown in the screenshot and click on **Attach storage**



3.5 Navigate to the cluster that you have created, and click on the **Connect** button



3.6 Copy the commands that you get on clicking on the Connect button

Connect to SI-Cluster ✕

Cloud shell
Azure CLI

Connect to your cluster using command line tooling to interact directly with cluster using kubectl, the command line tool for Kubernetes. Kubectl is available within the Azure Cloud Shell by default and can also be installed locally.

Set cluster context

- Open Cloud Shell
- Run the following commands

Set the cluster subscription

```
az account set --subscription 2f02f8ca-866d-410a-bfed-32876b28bc58
```

Download cluster credentials

```
az aks get-credentials --resource-group Regroup_8ovgZM6yBp4P_HDGgSYV7s -...
```

3.7 Navigate back to the bash window and paste the copied commands

```

Bash  [Icons]
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell

Storage fileshare subscription 2f02f8ca-866d-410a-bfed-32876b28bc58 is not registered to Microsoft.CloudShell Namespace. Please follow these instructions "https://aka.ms/RegisterCloudShell" to register. In future, unregistered subscriptions will have restricted access to CloudShell service.

student_7pmnz9x9lnwts2w2 [ ~ ]$ az account set --subscription 2f02f8ca-866d-410a-bfed-32876b28bc58
student_7pmnz9x9lnwts2w2 [ ~ ]$ az aks get-credentials --resource-group Regroup_8ovgZM6yBp4P_HDGgSYV7s --name SI-Cluster
Merged "SI-Cluster" as current context in /home/student_7pmnz9x9lnwts2w2/.kube/config
student_7pmnz9x9lnwts2w2 [ ~ ]$
  
```


3.8 Run the command **kubectl create namespace first-namespace** to create a namespace

```

Bash  ▾ | 🔌 ? ⚙️ 📄 📁 {} 📌

Welcome to Azure Cloud Shell

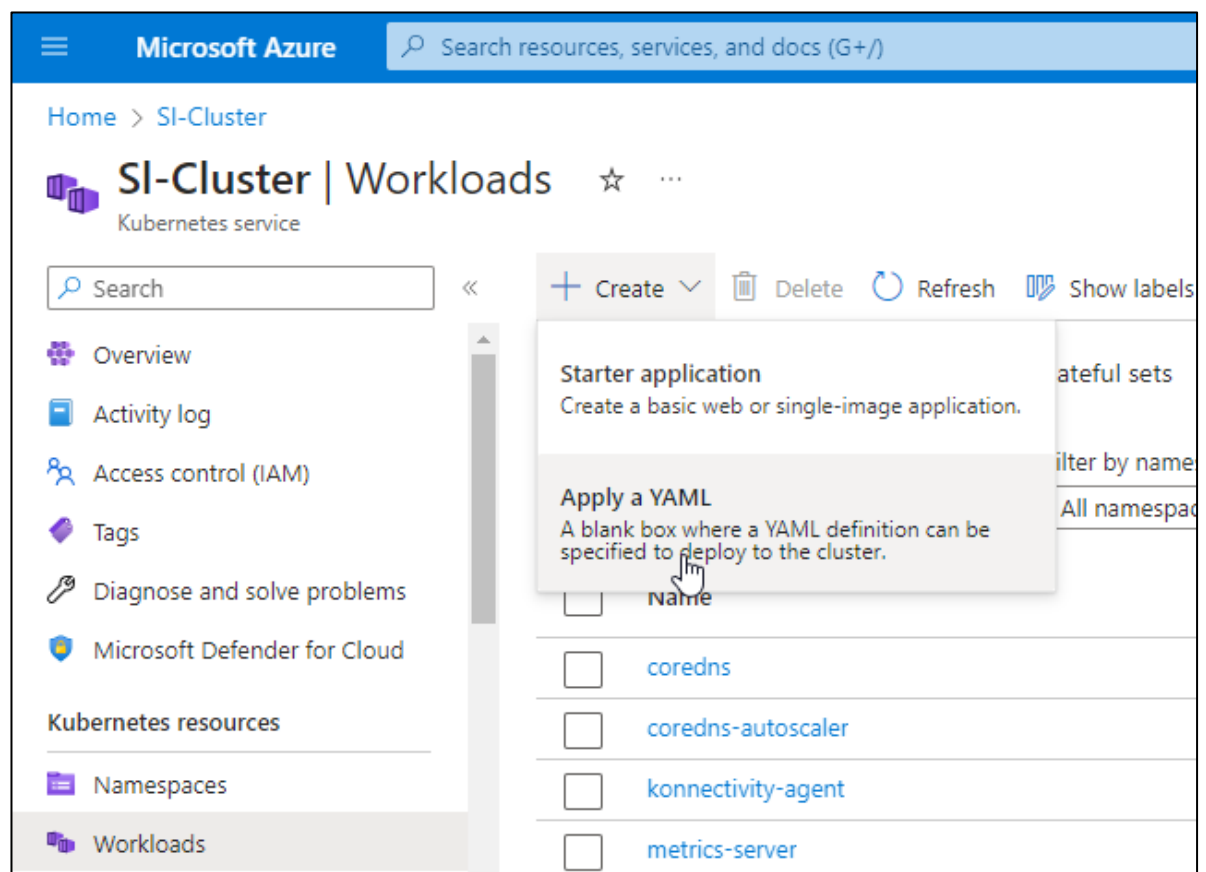
Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell

Storage fileshare subscription 2f02f8ca-866d-410a-bfed-32876b28bc58 is not registered to Microsoft
ions "https://aka.ms/RegisterCloudShell" to register. In future, unregistered subscriptions will be
removed.

student_7pnmz9x9lnwts2w2 [ ~ ]$ az account set --subscription 2f02f8ca-866d-410a-bfed-32876b2
student_7pnmz9x9lnwts2w2 [ ~ ]$ az aks get-credentials --resource-group Regroup_8ovg2M6yBp4P_
Merged "Sl-Cluster" as current context in /home/student_7pnmz9x9lnwts2w2/.kube/config
student_7pnmz9x9lnwts2w2 [ ~ ]$ kubectl create namespace first-namespace
namespace/first-namespace created
student_7pnmz9x9lnwts2w2 [ ~ ]$
  
```

Step 4: Create a deployment and roll out an update for it

4.1 Navigate to the cluster, select **Workloads** under Kubernetes resources section, click on the **Deployments** tab, click on **Create**, and choose **Apply a YAML** option



4.2 Add the following code in the YAML section, and click on **Add**

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: second-deployment
  namespace: first-namespace
  labels:
    app: second-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: second-deployment
  template:
    metadata:
      labels:
        app: second-deployment
    spec:
      containers:
        - name: nginx-container
          image: nginx:1.14.2
      ports:
        - containerPort: 80
```

Home > SI-Cluster | Workloads >

Add with YAML ...

Not sure where to start? [Deploy a quickstart application to get up and running.](#)

YAML JSON

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: second-deployment
5    namespace: first-namespace
6    labels:
7      app: second-deployment
8  spec:
9    replicas: 3
10   selector:
11     matchLabels:
12       app: second-deployment
13   template:
14     metadata:
15       labels:
16         app: second-deployment
```

Add Cancel

Home > SI-Cluster

SI-Cluster | Workloads ☆ ...

Kubernetes service

Search « + Create 🗑 Delete 🔄 Refresh 🏷 Show labels 🗨 Give feedback

Deployments Pods Replica sets Stateful sets Daemon sets Jobs Cron jobs

Filter by deployment name Filter by namespace

All namespaces 🔍 Add label filter

<input type="checkbox"/>	Name	Namespace	Ready	Up-to-date	Available	Age ↓
<input type="checkbox"/>	coredns	kube-system	✓ 2/2	2	2	9 minutes
<input type="checkbox"/>	coredns-autoscaler	kube-system	✓ 1/1	1	1	9 minutes
<input type="checkbox"/>	connectivity-agent	kube-system	✓ 2/2	2	2	9 minutes
<input type="checkbox"/>	metrics-server	kube-system	✓ 2/2	2	2	9 minutes
<input type="checkbox"/>	tigera-operator	tigera-operator	✓ 1/1	1	1	9 minutes
<input type="checkbox"/>	calico-kube-controllers	calico-system	✓ 1/1	1	1	7 minutes
<input type="checkbox"/>	calico-typha	calico-system	✓ 1/1	1	1	7 minutes
<input type="checkbox"/>	ama-logs-rs	kube-system	✓ 1/1	1	1	7 minutes
<input type="checkbox"/>	second-deployment	first-namespace	✓ 3/3	3	3	35 seconds

The deployment named **second-deployment** is successfully created.

4.3 Run the following commands to check the newly created deployment:

```
kubectl get deployments -n first-namespace
```

```
kubectl get rs -n first-namespace
```

```
kubectl get pods --show-labels -n first-namespace
```

```
Bash
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Storage fileshare subscription 2f02f8ca-866d-410a-bfed-32876b28bc58 is not registered to Microsoft.CloudShell Namespace. Please follow
instructions "https://aka.ms/RegisterCloudShell" to register. In future, unregistered subscriptions will have restricted access to CloudShell

student_7pmmz9x9lnwts2w2 [ ~ ]$ kubectl get deployments -n first-namespace
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
second-deployment   3/3     3            3           92s

student_7pmmz9x9lnwts2w2 [ ~ ]$ kubectl get rs -n first-namespace
NAME                                DESIRED   CURRENT   READY   AGE
second-deployment-7b57746758        3         3         3       112s

student_7pmmz9x9lnwts2w2 [ ~ ]$ kubectl get pods --show-labels -n first-namespace
NAME                                READY   STATUS    RESTARTS   AGE   LABELS
second-deployment-7b57746758-2twsb  1/1     Running   0          2m1s  app=second-deployment,pod-template-hash=7b57746758
second-deployment-7b57746758-vs6zq  1/1     Running   0          2m1s  app=second-deployment,pod-template-hash=7b57746758
second-deployment-7b57746758-zqn5l  1/1     Running   0          2m1s  app=second-deployment,pod-template-hash=7b57746758
student_7pmmz9x9lnwts2w2 [ ~ ]$
```

4.4 Run the following commands to update the **Nginx** image version to **1.20.2** in the second-deployment:

```
student_7pmmz9x9lnwts2w2 [ ~ ]$ kubectl set image deployment/second-deployment \
> -n first-namespace nginx-container=nginx:1.20.2 --record
Flag --record has been deprecated, --record will be removed in the future
deployment.apps/second-deployment image updated
student_7pmmz9x9lnwts2w2 [ ~ ]$
```

By following these steps, you have successfully connected to an AKS cluster from the Azure Cloud Shell.