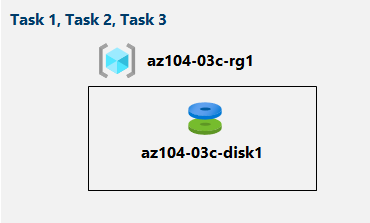
Lab URL:

<https://microsoftlearning.github.io/AZ-104-MicrosoftAzureAdministrator/Instructions/Labs/LAB_03c-Manage_Azure_Resources_by_Using_Azure_PowerShell.html>

Architecture diagram

[](https://microsoftlearning.github.io/AZ-104-MicrosoftAzureAdministrator/Instructions/media/lab03c.png)

Instructions

Exercise 1

Task 1: Start a PowerShell session in Azure Cloud Shell

In this task, you will open a PowerShell session in Cloud Shell.

1. In the portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
2. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

**Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

1. If prompted, click **Create storage**, and wait until the Azure Cloud Shell pane is displayed.

Graphical user interface, text, application

Description automatically generated

1. Ensure **PowerShell** appears in the drop-down menu in the upper-left corner of the Cloud Shell pane.

Graphical user interface, text

Description automatically generated

Task 2: Create a resource group and an Azure managed disk by using Azure PowerShell

In this task, you will create a resource group and an Azure managed disk by using Azure PowerShell session within Cloud Shell

1. To create a resource group in the same Azure region as the **az104-03b-rg1** resource group you created in the previous lab, from the PowerShell session within Cloud Shell, run the following:

CodeCopy

$location = (Get-AzResourceGroup -Name az104-03b-rg1).Location

$rgName = 'az104-03c-rg1'

New-AzResourceGroup -Name $rgName -Location $location

Graphical user interface, text

Description automatically generated

1. To retrieve properties of the newly created resource group, run the following:

CodeCopy

Get-AzResourceGroup -Name $rgName

1. To create a new managed disk with the same characteristics as those you created in the previous labs of this module, run the following:

CodeCopy

$diskConfig = New-AzDiskConfig `

-Location $location `

-CreateOption Empty `

-DiskSizeGB 32 `

-Sku Standard\_LRS

$diskName = 'az104-03c-disk1'

New-AzDisk `

-ResourceGroupName $rgName `

-DiskName $diskName `

-Disk $diskConfig

Text

Description automatically generated

1. To retrieve properties of the newly created disk, run the following:

CodeCopy

Get-AzDisk -ResourceGroupName $rgName -Name $diskName

Task 3: Configure the managed disk by using Azure PowerShell

In this task, you will be managing the configuration of the Azure managed disk by using Azure PowerShell session within Cloud Shell.

1. To increase the size of the Azure managed disk to **64 GB**, from the PowerShell session within Cloud Shell, run the following:

CodeCopy

New-AzDiskUpdateConfig -DiskSizeGB 64 | Update-AzDisk -ResourceGroupName $rgName -DiskName $diskName

Graphical user interface, text

Description automatically generated

1. To verify that the change took effect, run the following:

CodeCopy

Get-AzDisk -ResourceGroupName $rgName -Name $diskName

1. To verify the current SKU as **Standard\_LRS**, run the following:

CodeCopy

(Get-AzDisk -ResourceGroupName $rgName -Name $diskName).Sku

A picture containing graphical user interface

Description automatically generated

1. To change the disk performance SKU to **Premium\_LRS**, from the PowerShell session within Cloud Shell, run the following:

CodeCopy

New-AzDiskUpdateConfig -Sku Premium\_LRS | Update-AzDisk -ResourceGroupName $rgName -DiskName $diskName

1. To verify that the change took effect, run the following:

CodeCopy

A picture containing graphical user interface

Description automatically generated

(Get-AzDisk -ResourceGroupName $rgName -Name $diskName).Sku

Clean up resources

**Note**: Do not delete resources you deployed in this lab. You will reference them in the next lab of this module.

Review

In this lab, you have:

* Started a PowerShell session in Azure Cloud Shell
* Created a resource group and an Azure managed disk by using Azure PowerShell
* Configured the managed disk by using Azure PowerShell