**Module 3: Azure Administration**

**Exercise 1: Manage Azure resources by Using the Azure Portal**

#### Task 1: Create resource groups and deploy resources to resource groups

In this task, you will use the Azure portal to create resource groups and create a disk in the resource group.

1. Sign in to the [Azure portal](https://portal.azure.com/).
2. Search for and select **Resource groups**.
3. On the **Resource groups** blade, click **+ Add** and create a resource group with the following settings:

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Subscription | **the name of the Azure subscription you will use in this lab** |
| Resource Group | **az104-03a-rg1** |
| Region | **the name of any Azure region available in the subscription you will use in this lab** |

1. Click **Review + Create** and then click **Create**.
2. In the Azure portal, search for and select **Disks**, click **+ Add**, and specify the following settings:

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Subscription | **the name of the Azure subscription you will use in this lab** |
| Resource Group | **az104-03a-rg1** |
| Disk name | **az104-03a-disk1** |
| Region | **the name of any Azure region available in the subscription you will use in this lab** |
| Availability zone | **None** |
| Source type | **None** |

**Note**: When creating a resource, you have the option of creating a new resource group or using an existing one.

1. Change the disk type and size to **Standard HDD** and **32 GiB**, respectively.
2. Click **Review + Create** and then click **Create**.

**Note**: Wait until the disk is created. This should take less than a minute.

#### Task 2: Move resources between resource groups

In this task, we will move the disk resource you created in the previous task to a new resource group.

1. Search for and select **Resource groups**.
2. On the **Resource groups** blade, click the entry representing the **az104-03a-rg1** resource group you created in the previous task.
3. From the **Overview** blade of the resource group, in the list of resource group resources, select the entry representing the newly created disk, click **Move** in the toolbar, and, in the drop-down list, select **Move to another resource group**.

**Note**: This method allows you to move multiple resources at the same time.

1. On the **Move resources** blade, click **Create a new group**.
2. In the **Resource group** text box, type **az104-03a-rg2**, select the checkbox **I understand that tools and scripts associated with moved resources will not work until I update them to use new resource IDs**, and click **OK**.

**Note**: Do not wait for the move to complete but instead proceed to the next task. The move might take about 10 minutes. You can determine that the operation was completed by monitoring activity log entries of the source or target resource group. Revisit this step once you complete the next task.

#### Task 3: Implement resource locks

In this task, you will apply a resource lock to an Azure resource group containing a disk resource.

1. In the Azure portal, search for and select **Disks**, click **+ Add**, and specify the following settings:

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Subscription | **the name of the Azure subscription you will use in this lab** |
| Resource Group | **az104-03a-rg3** |
| Disk name | **az104-03a-disk2** |
| Region | **the name of any Azure region available in the subscription you will use in this lab** |
| Availability zone | **None** |
| Source type | **None** |

1. Set the disk type and size to **Standard HDD** and **32 GiB**, respectively.
2. Click **Review + Create** and then click **Create**.
3. In the Azure portal, search for and select **Resource groups**.
4. In the list of resource groups, click the entry representing the **az104-03a-rg3** resource group.
5. On the **az104-03a-rg3** resource group blade, click **Locks** and add a lock with the following settings:

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Lock name | **az104-03a-delete-lock** |
| Lock type | **Delete** |

1. On the **az104-03a-rg3** resource group blade, click **Overview**, in the list of resource group resources, select the entry representing the disk you created earlier in this task, and click **Delete** in the toolbar.
2. When prompted **Do you want to delete all the selected resources?**, in the **Confirm delete** text box, type **yes** and click **Delete**.
3. You should see an error message, notifying about the failed delete operation.

**Note**: As the error message states, this is expected due to the delete lock applied on the resource group level.

1. Navigate back to the list of resources of the **az104-03a-rg3** resource group and click the entry representing the **az104-03a-disk2** resource.
2. On the **az104-03a-disk2** blade, in the **Settings** section, click **Configuration**, set the disk type and size to **Premium SSD** and **64 GiB**, respectively, and save the change. Verify that the change was successful.

**Note**: This is expected, since the resource group-level lock applies to delete operations only.

#### Clean up resources

**Note**: Do not delete resources you deployed in this lab. You will be using them in the next lab of this module. Remove only the resource lock you created in this lab.

1. Navigate to the **az104-03a-rg3** resource group blade, display its **Locks** blade, and remove the lock **az104-03a-delete-lock** by clicking the **Delete** link on the right-hand side of the **Delete** lock entry.

**Exercise 2: Manage Azure resources by Using ARM Templates**

#### Task 1: Review an ARM template for deployment of an Azure managed disk

In this task, you will create an Azure disk resource by using an Azure Resource Manager template.

1. Sign in to the [Azure portal](https://portal.azure.com/).
2. In the Azure portal, search for and select **Resource groups**.
3. In the list of resource groups, click **az104-03a-rg1**.
4. On the **az104-03a-rg1** resource group blade, in the **Settings** section, click **Deployments**.
5. On the **az104-03a-rg1 - Deployments** blade, click the first entry in the list of deployments and then click **View template**.

**Note**: Review the content of the template and note that you have the option to download it to the local computer, add it to the library, and re-deploy it.

1. Click **Download** and save the compressed file containing the template and parameters files to the **Downloads** folder on your lab computer.
2. Extract the content of the downloaded file into the **Downloads** folder on your lab computer.

**Note**: These files are also available as **az104-03b-md-template.json** and **az104-03b-md-parameters.json**

#### Task 2: Create an Azure managed disk by using an ARM template

1. In the Azure portal, search for and select **Template deployment (Deploy a custom template)**.
2. On the **Custom deployment** blade, click **Build your own template in the editor**.
3. On the **Edit template** blade, click **Load file** and upload the template file you downloaded in the previous step.
4. Within the editor pane, remove the following lines:

*"sourceResourceId": {*

*"type": "String"*

*},*

*"sourceUri": {*

*"type": "String"*

*},*

*"osType": {*

*"type": "String"*

*},*

*},*

*"hyperVGeneration": {*

*"defaultValue": "V1",*

*"type": "String"*

*"osType": "[parameters('osType')]"*

**Note**: These parameters are removed since they are not applicable to the current deployment. In particular, sourceResourceId, sourceUri, osType, and hyperVGeneration parameters are applicable to creating an Azure disk from an existing VHD file.

1. In addition, remove the trailing comma from the following line:

*"diskSizeGB": "[parameters('diskSizeGb')]",*

**Note**: This is necessary to account for the syntax rules of JSON-based ARM templates.

1. Save the changes.
2. Back on the **Custom deployment** blade, click **Edit parameters**.
3. On the **Edit parameters** blade, click **Load file** and upload the parameters file **az104-03b-md-parameters.json** and save the changes.
4. Back on the **Custom deployment** blade, specify the following settings:

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Subscription | **the name of the Azure subscription you will use in this lab** |
| Resource Group | **the name of a new resource group az104-03b-rg1** |
| Location | **the name of any Azure region available in the subscription you are using in this lab** |
| Disk name | **az104-03b-disk1** |
| Location | **accept the default value** |
| Sku | **Standard\_LRS** |
| Disk Size Gb | **32 GB** |
| Create Option | **empty** |
| Disk Encryption Set Type | **EncryptionAtRestWithPlatformKey** |

1. Select the checkbox **I agree to the terms and conditions stated above** and click **Purchase**.
2. Verify that the deployment completed successfully.

#### Task 3: Review the ARM template-based deployment of the managed disk

1. In the Azure portal, search for and select **Resource groups**.
2. In the list of resource groups, click **az104-03b-rg1**.
3. On the **az104-03b-rg1** resource group blade, in the **Settings** section, click **Deployments**.
4. From the **az104-03b-rg1 - Deployments** blade, click the first entry in the list of deployments and review the content of the **Input** and **Template** blades.

#### Clean up resources

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

**Exercise 3: Manage Azure resources by Using Azure PowerShell**

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

#### 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:

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

*$rgName = 'az104-03c-rg1'*

*New-AzResourceGroup -Name $rgName -Location $location*

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

*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:

*$diskConfig = New-AzDiskConfig `*

*-Location $location `*

*-CreateOption Empty `*

*-DiskSizeGB 32 `*

*-Sku Standard\_LRS*

*$diskName = 'az104-03c-disk1'*

*New-AzDisk `*

*-ResourceGroupName $rgName `*

*-DiskName $diskName `*

*-Disk $diskConfig*

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

*Get-AzDisk -ResourceGroupName $rgName -Name $diskName*

#### Task 3: Configure the managed disk by using Azure PowerShell

In this task, you will managing 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:

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

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

*Get-AzDisk -ResourceGroupName $rgName -Name $diskName*

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

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

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

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

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

*(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.

**Exercise 4: Manage Azure resources by Using Azure CLI**

#### Task 1: Start a Bash session in Azure Cloud Shell

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

1. From 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 **Bash**.

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

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

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

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

*$LOCATION=$(az group show --name 'az104-03c-rg1' --query location --out tsv)*

*$RGNAME='az104-03d-rg1'*

*az group create --name $RGNAME --location $LOCATION*

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

*az group show --name $RGNAME*

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

$DISKNAME='az104-03d-disk1'

az disk create --resource-group $RGNAME --name $DISKNAME --sku Standard\_LRS --size-gb 32

**Note**: When using multi-line syntax, ensure that each line ends with back-slash (\) with no trailing spaces and that there are no leading spaces at the beginning of each line.

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

*az disk show --resource-group $RGNAME --name $DISKNAME*

#### Task 3: Configure the managed disk by using Azure CLI

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

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

*az disk update --resource-group $RGNAME --name $DISKNAME --size-gb 64*

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

*az disk show --resource-group $RGNAME --name $DISKNAME --query diskSizeGb*

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

*az disk update --resource-group $RGNAME --name $DISKNAME --sku 'Premium\_LRS'*

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

*az disk show --resource-group $RGNAME --name $DISKNAME --query sku*

#### Clean up resources

**Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **Bash** shell session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

*az group list --query "[?starts\_with(name,'az104-03')].name" --output tsv*

1. Delete all resource groups you created throughout the labs of this module by running the following command:

*az group list --query "[?starts\_with(name,'az104-03')].[name]" --output tsv | xargs -L1 bash -c 'az group delete --name $0 --no-wait --yes'*

**Note**: The command executes asynchronously (as determined by the –nowait parameter), so while you will be able to run another Azure CLI command immediately afterwards within the same Bash session, it will take a few minutes before the resource groups are actually removed.