# Lab 03 - Manage Azure resources by using Azure Resource Manager Templates

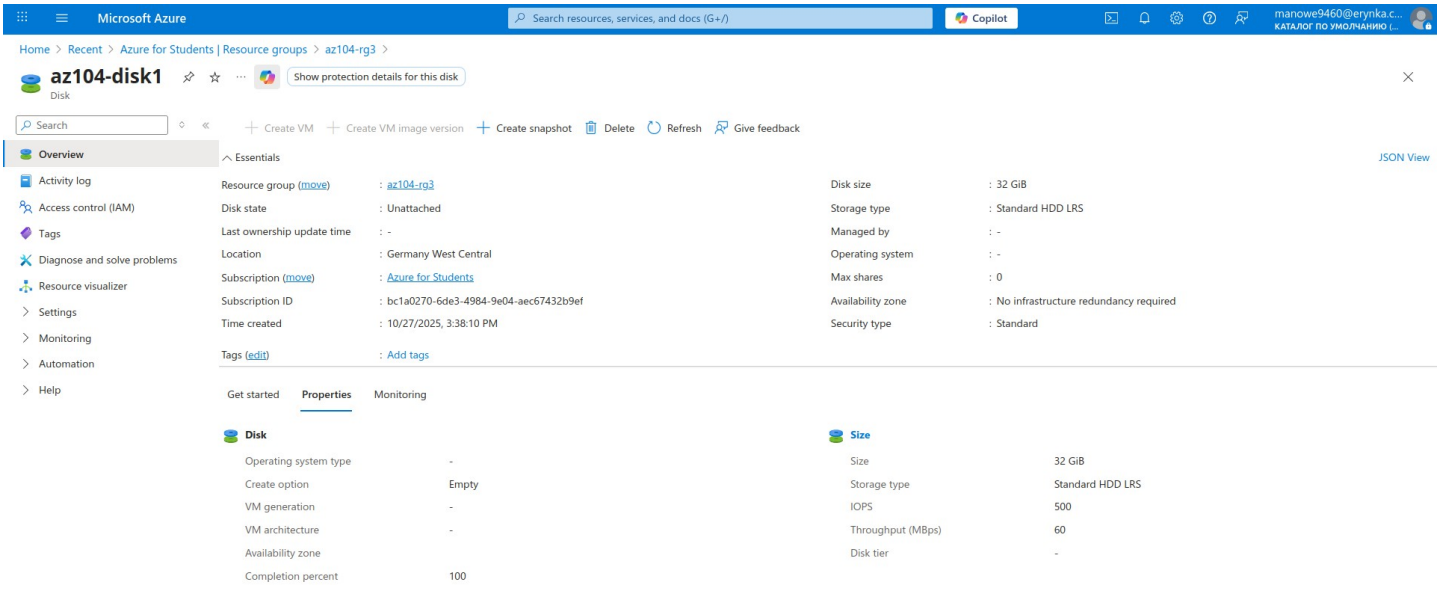
https://github.com/taras123programmer/CloudTechnologies/tree/main/lab3

## Task 1: Create an Azure Resource Manager template

```
user@user-HP-250-G7-Notebook-PC:/media/user/EEC899A8C8996F97/Cloud/lab3$ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/azuread versions matching "~> 2.48"...
- Finding hashicorp/azurerm versions matching "~> 3.100"...
- Installing hashicorp/azuread v2.53.1...
- Installed hashicorp/azuread v2.53.1 (signed by HashiCorp)
- Installing hashicorp/azurerm v3.117.1...
- Installed hashicorp/azurerm v3.117.1 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

```
user@user-HP-250-G7-Notebook-PC:/media/user/EEC899A8C8996F97/Cloud/lab3$ terraform plan
azurerm_resource_group.rg: Refreshing state... [id=/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create
-/+ destroy and then create replacement

Terraform will perform the following actions:

  # azurerm_managed_disk.disk1 will be created
  + resource "azurerm_managed_disk" "disk1" {
      + create_option                     = "Empty"
      + disk_iops_read_only               = (known after apply)
      + disk_iops_read_write              = (known after apply)
      + disk_mbps_read_only               = (known after apply)
      + disk_mbps_read_write              = (known after apply)
      + disk_size_gb                      = 32
      + id                                = (known after apply)
      + location                          = "germanywestcentral"
      + logical_sector_size               = (known after apply)
      + max_shares                        = (known after apply)
      + name                              = "az104-disk1"
      + optimized_frequent_attach_enabled = false
      + performance_plus_enabled          = false
      + public_network_access_enabled     = true
      + resource_group_name               = "az104-rg3"
      + source_uri                        = (known after apply)
      + storage_account_type              = "Standard_LRS"
      + tier                              = (known after apply)
    }

  # azurerm_resource_group.rg must be replaced
-/+ resource "azurerm_resource_group" "rg" {
      ~ id       = "/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3" -> (known after apply)
      ~ location = "eastus" -> "germanywestcentral" # forces replacement
        name     = "az104-rg3"
      - tags     = {} -> null
        # (1 unchanged attribute hidden)
    }

Plan: 2 to add, 0 to change, 1 to destroy.
```

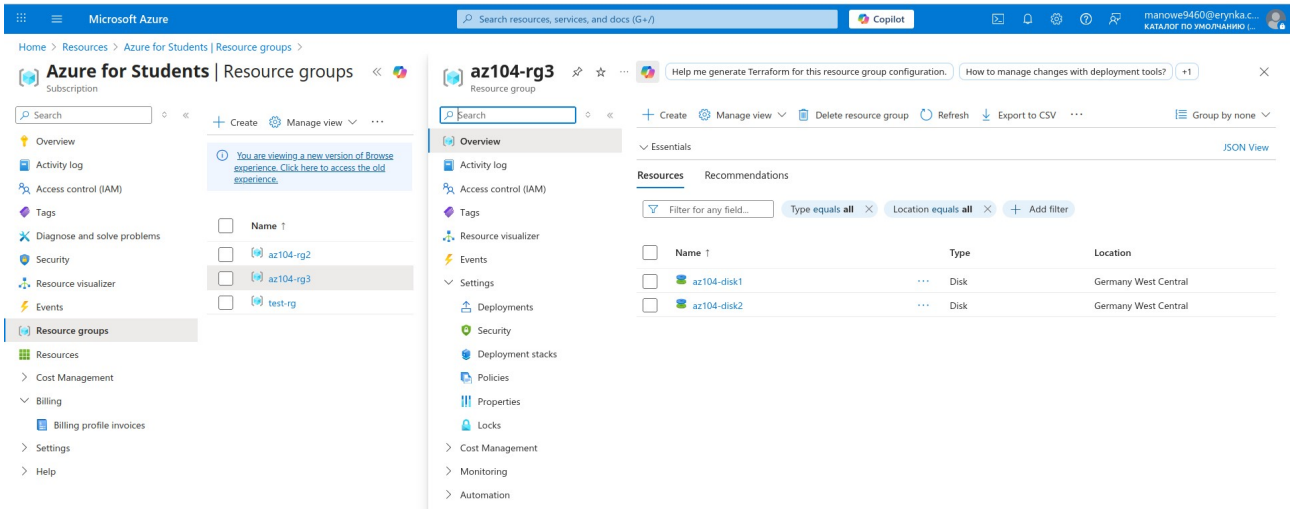
```
user@user-HP-250-G7-Notebook-PC:/media/user/EEC899A8C8996F97/Cloud/lab3$ terraform apply
      + disk_iops_read_write             = (known after apply)
      + disk_mbps_read_only              = (known after apply)
      + disk_mbps_read_write             = (known after apply)
      + disk_size_gb                     = 32
      + id                               = (known after apply)
      + location                         = "germanywestcentral"
      + logical_sector_size              = (known after apply)
      + max_shares                       = (known after apply)
      + name                             = "az104-disk1"
      + optimized_frequent_attach_enabled = false
      + performance_plus_enabled         = false
      + public_network_access_enabled    = true
      + resource_group_name              = "az104-rg3"
      + source_uri                       = (known after apply)
      + storage_account_type             = "Standard_LRS"
      + tier                             = (known after apply)
    }

  # azurerm_resource_group.rg must be replaced
-/+ resource "azurerm_resource_group" "rg" {
      ~ id       = "/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3" -> (known after apply)
      ~ location = "eastus" -> "germanywestcentral" # forces replacement
        name     = "az104-rg3"
      - tags     = {} -> null
        # (1 unchanged attribute hidden)
    }

Plan: 2 to add, 0 to change, 1 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

azurerm_resource_group.rg: Destroying... [id=/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3]
azurerm_resource_group.rg: Still destroying... [id=/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3, 00m10s elapsed]
azurerm_resource_group.rg: Destruction complete after 18s
azurerm_resource_group.rg: Creating...
azurerm_resource_group.rg: Still creating... [00m10s elapsed]
azurerm_resource_group.rg: Creation complete after 11s [id=/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3]
azurerm_managed_disk.disk1: Creating...
azurerm_managed_disk.disk1: Creation complete after 6s [id=/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3/providers/Microsoft.Compute/disks/az104-disk1]

Apply complete! Resources: 2 added, 0 changed, 1 destroyed.
```



## Task 2: Edit an Azure Resource Manager template and then redeploy the template

```
user@user-HP-250-G7-Notebook-PC:/media/user/EEC899A8C8996F97/Cloud/lab3$ terraform plan
azurerm_resource_group.rg: Refreshing state... [id=/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3]
azurerm_managed_disk.disk1: Refreshing state... [id=/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3/providers/Microsoft.Compute/disks
/az104-disk1]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # azurerm_managed_disk.disk2 will be created
  + resource "azurerm_managed_disk" "disk2" {
      + create_option                 = "Empty"
      + disk_iops_read_only           = (known after apply)
      + disk_iops_read_write          = (known after apply)
      + disk_mbps_read_only           = (known after apply)
      + disk_mbps_read_write          = (known after apply)
      + disk_size_gb                  = 32
      + id                            = (known after apply)
      + location                      = "germanywestcentral"
      + logical_sector_size           = (known after apply)
      + max_shares                    = (known after apply)
      + name                          = "az104-disk2"
      + optimized_frequent_attach_enabled = false
      + performance_plus_enabled      = false
      + public_network_access_enabled = true
      + resource_group_name           = "az104-rg3"
      + source_uri                    = (known after apply)
      + storage_account_type          = "Standard_LRS"
      + tier                          = (known after apply)
    }

Plan: 1 to add, 0 to change, 0 to destroy.
```

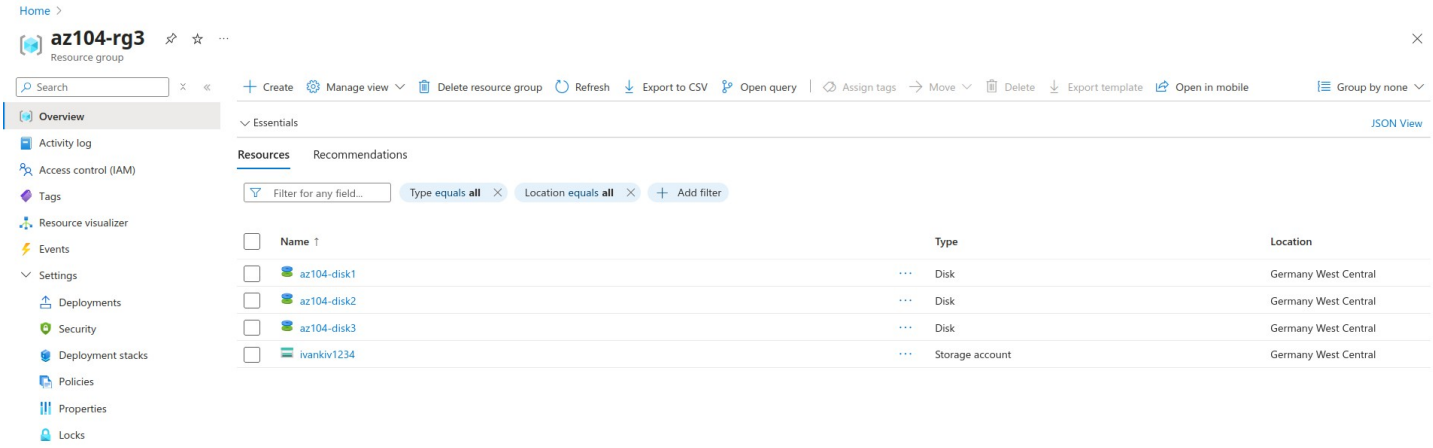
```
user@user-HP-250-G7-Notebook-PC:/media/user/EEC899A8C8996F97/Cloud/lab3$ terraform apply
  + resource "azurerm_managed_disk" "disk2" {
      + create_option                 = "Empty"
      + disk_iops_read_only           = (known after apply)
      + disk_iops_read_write          = (known after apply)
      + disk_mbps_read_only           = (known after apply)
      + disk_mbps_read_write          = (known after apply)
      + disk_size_gb                  = 32
      + id                            = (known after apply)
      + location                      = "germanywestcentral"
      + logical_sector_size           = (known after apply)
      + max_shares                    = (known after apply)
      + name                          = "az104-disk2"
      + optimized_frequent_attach_enabled = false
      + performance_plus_enabled      = false
      + public_network_access_enabled = true
      + resource_group_name           = "az104-rg3"
      + source_uri                    = (known after apply)
      + storage_account_type          = "Standard_LRS"
      + tier                          = (known after apply)
    }

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value:  yes

azurerm_managed_disk.disk2: Creating...
azurerm_managed_disk.disk2: Creation complete after 5s [id=/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/r
e/disks/az104-disk2]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```




# Task 3: Configure the Cloud Shell and deploy a template with PowerShell

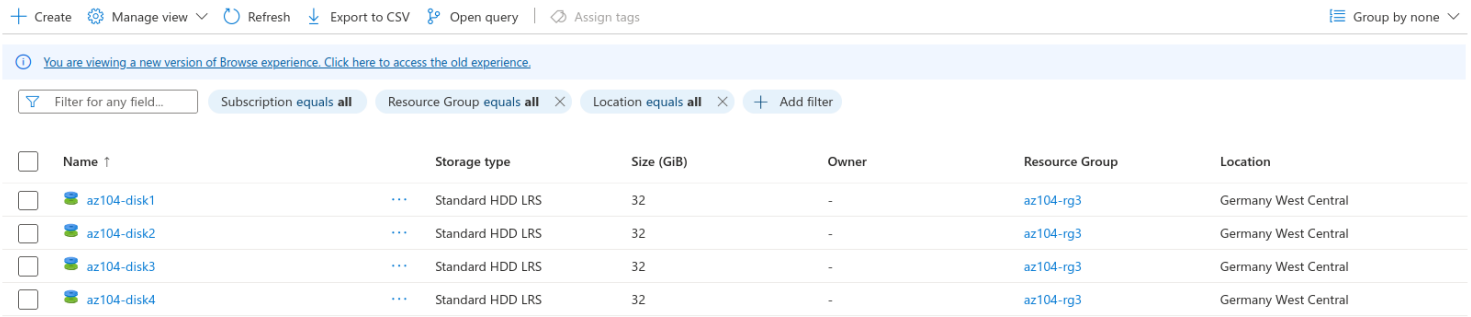
```
user@user-HP-250-G7-Notebook-PC:/media/user/EEC899A8C8996F97/Cloud/lab3$ terraform plan
      + sftp_enabled                    = false
      + shared_access_key_enabled       = true
      + table_encryption_key_type       = "Service"
      + tags                            = {
          + "environment" = "dev"
        }

      + blob_properties (known after apply)

      + network_rules (known after apply)

      + queue_properties (known after apply)

      + routing (known after apply)

      + share_properties (known after apply)

      + static_website (known after apply)
    }

  # azurerm_storage_share.fileshare will be created
  + resource "azurerm_storage_share" "fileshare" {
      + access_tier          = (known after apply)
      + enabled_protocol     = "SMB"
      + id                   = (known after apply)
      + metadata             = (known after apply)
      + name                 = "fs-cloudshell"
      + quota                = 100
      + resource_manager_id  = (known after apply)
      + storage_account_name = "stcloudshell123"
      + url                  = (known after apply)
    }

Plan: 2 to add, 0 to change, 0 to destroy.
```

```
user@user-HP-250-G7-Notebook-PC:/media/user/EEC899A8C8996F97/Cloud/lab3$ terraform apply
azurerm_storage_account.sa: Creating...
```




# Task 4: Deploy a template with the CLI

```
user@user-HP-250-G7-Notebook-PC:~/Downloads/ExportedTemplate-az104-rg3$  az deployment group create --resource-group az104-rg3 --template-file template.json --parameters parameters.json
{
  "id": "/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3/providers/Microsoft.Resources/deployments/template",
  "location": null,
  "name": "template",
  "properties": {
    "correlationId": "bd6815ff-4182-49e3-a10f-e98203ae5a91",
    "debugSetting": null,
    "dependencies": [],
    "diagnostics": null,
    "duration": "PT8.7173588S",
    "error": null,
    "extensions": null,
    "mode": "Incremental",
    "onErrorDeployment": null,
    "outputResources": [
      {
        "apiVersion": null,
        "extension": null,
        "id": "/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3/providers/Microsoft.Compute/disks/az104-disk1",
        "identifiers": null,
        "resourceGroup": "az104-rg3",
        "resourceType": null
      }
    ],
    "outputs": null,
    "parameters": {
      "disks_az104_disk4_name": {
        "type": "String",
        "value": "az104-disk1"
      }
    },
```



# Task 5: Deploy a resource by using Azure Bicep

```
ivankiv [ ~ ]$ nano azuredeploydisk.bicep
ivankiv [ ~ ]$  az deployment group create --resource-group az104-rg3 --template-file azuredeploydisk.bicep

{
  "id": "/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3/providers/Microsoft.Resources/deployments/azuredeploydisk",
  "location": null,
  "name": "azuredeploydisk",
  "properties": {
    "correlationId": "22dc53f7-7657-42be-9871-5e1cf5dda3dd",
    "debugSetting": null,
    "dependencies": [],
    "diagnostics": null,
    "duration": "PT7.52556S",
    "error": null,
    "extensions": null,
    "mode": "Incremental",
    "onErrorDeployment": null,
    "outputResources": [
      {
        "apiVersion": null,
        "extension": null,
        "id": "/subscriptions/bc1a0270-6de3-4984-9e04-aec67432b9ef/resourceGroups/az104-rg3/providers/Microsoft.Compute/disks/az104-disk5",
        "identifiers": null,
        "resourceGroup": "az104-rg3",
        "resourceType": null
      }
    ],
```

```
ivankiv [ ~ ]$  az disk list --output table

Name         ResourceGroup    Location            Zones    Sku              SizeGb    ProvisioningState
-----------  ---------------  ------------------  -------  ---------------  --------  -------------------
az104-disk1  AZ104-RG3        germanywestcentral           Standard_LRS     32        Succeeded
az104-disk2  AZ104-RG3        germanywestcentral           Standard_LRS     32        Succeeded
az104-disk3  AZ104-RG3        germanywestcentral           Standard_LRS     32        Succeeded
az104-disk4  AZ104-RG3        germanywestcentral           Standard_LRS     32        Succeeded
az104-disk5  AZ104-RG3        germanywestcentral           StandardSSD_LRS  32        Succeeded
```