

**Міністерство освіти і науки України  
Карпатський національний університет  
імені В.Стефаника**

**Факультет математики та інформатики  
Кафедра інформаційних технологій**

**Хмарні технології**

**Практична робота №3**

**Тема:** Manage Azure resources by using Azure Resource Manager Templates

**Мета:** Навчитися автоматизувати розгортання ресурсів

Виконав: Андрусяк І.Р.  
Група ІПЗ-41  
Дата: 25 листопада 2025р.  
Викладач: Поварчук Д.Д.

Івано-Франківськ — 2025

### Task 1: Створення керованого диску через Terraform

```
1 resource "azurerm_resource_group" "rg" {
2   name      = var.resource_group_name
3   location  = var.location
4 }
5
6 resource "azurerm_managed_disk" "disk1" {
7   name          = var.disk_name
8   location      = azurerm_resource_group.rg.location
9   resource_group_name = azurerm_resource_group.rg.name
10  storage_account_type = var.storage_account_type
11  create_option       = "Empty"
12  disk_size_gb        = var.disk_size_gb
13
14  tags = {
15    environment = "lab"
16    task        = "task1"
17  }
18 }
```

File: main.tf

```
1 output "disk_id" {
2   value = azurerm_managed_disk.disk1.id
3 }
4
5 output "disk_name" {
6   value = azurerm_managed_disk.disk1.name
7 }
8
9 output "resource_group_name" {
10  value = azurerm_resource_group.rg.name
11 }
```

File: outputs.tf

```

1  terraform {
2      required_providers {
3          azurerm = {
4              source  = "hashicorp/azurerm"
5              version = "~>3.0"
6          }
7      }
8  }
9
10 provider "azurerm" {
11     features {}
12 }
```

File: providers.tf

```

1  []
2      "creationData": {
3          "createOption": "Empty"
4      },
5      "diskIOPSReadWrite": 500,
6      "diskMBpsReadWrite": 60,
7      "diskSizeBytes": 34359738368,
8      "diskSizeGB": 32,
9      "diskState": "Unattached",
10     "encryption": {
11         "type": "EncryptionAtRestWithPlatformKey"
12     },
13     "id": "/subscriptions/16763438-d8c0-4e82-b6b9-1c54738e6db3/resourceGroups/az104-rg3/providers/Microsoft.Compute/disks/az104-disk1",
14     "location": "eastus",
15     "name": "az104-disk1",
16     "networkAccessPolicy": "AllowAll",
17     "optimizedForFrequentAttach": false,
18     "provisioningState": "Succeeded",
19     "publicNetworkAccess": "Enabled",
20     "resourceGroup": "az104-rg3",
21     "sku": {
22         "name": "Standard_LRS",
23         "tier": "Standard"
24     },
25     "tags": {
26         "environment": "lab",
27         "task": "task1"
28     },
29     "timeCreated": "2025-11-25T17:11:44.1983519+00:00",
30     "type": "Microsoft.Compute/disks",
31     "uniqueId": "abd7b8c6-7a5c-4028-af4a-25b42de479d7"
32 }
33 ]
```

File: disk-template.json

```

1  {
2      "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
3      "contentVersion": "1.0.0.0",
4      "parameters": {
5          "disks_az104_disk1_name": {
6              "type": "String"
7          }
8      },
9      "resources": [
10         {
11             "apiVersion": "2025-01-02",
12             "location": "eastus",
13             "name": "[parameters('disks_az104_disk1_name')]",
14             "properties": {
15                 "creationData": {
16                     "createOption": "Empty"
17                 },
18                 "diskIOPSReadWrite": 500,
19                 "diskMBpsReadWrite": 60,
20                 "diskSizeGB": 32,
21                 "encryption": {
22                     "type": "EncryptionAtRestWithPlatformKey"
23                 },
24                 "networkAccessPolicy": "AllowAll",
25                 "optimizedForFrequentAttach": false,
26                 "publicNetworkAccess": "Enabled"
27             },
28             "sku": {
29                 "name": "Standard_LRS",
30                 "tier": "Standard"
31             },
32             "tags": {
33                 "environment": "lab",
34                 "task": "task1"
35             },
36             "type": "Microsoft.Compute/disks"
37         },
38     ],
39     "variables": {}
40 }

```

File: export-template.json

## Task 2: Редагування та повторне розгортання шаблону через Terraform

```

1  data "azurerm_resource_group" "existing" {
2      name = var.resource_group_name
3  }
4
5  resource "azurerm_managed_disk" "disk2" {
6      name           = var.disk_name
7      location       = data.azurerm_resource_group.existing.location
8      resource_group_name = data.azurerm_resource_group.existing.name
9      storage_account_type = var.storage_account_type
10     create_option    = "Empty"
11     disk_size_gb    = var.disk_size_gb
12
13     tags = {
14         environment = "lab"
15         task        = "task2"
16     }
17 }

```

File: main.tf

```
1  output "disk_id" {
2    value = azurerm_managed_disk.disk2.id
3  }
4
5  output "disk_name" {
6    value = azurerm_managed_disk.disk2.name
7 }
```

File: outputs.tf

```
1  terraform {
2    required_providers {
3      azurerm = {
4        source  = "hashicorp/azurerm"
5        version = "~>3.0"
6      }
7    }
8  }
9
10 provider "azurerm" {
11   features {}
12 }
```

File: providers.tf

```
1  variable "resource_group_name" {
2    default = "az104-rg3"
3  }
4
5  variable "location" {
6    default = "East US"
7  }
8
9  variable "disk_name" {
10   default = "az104-disk2"
11 }
12
13 variable "disk_size_gb" {
14   default = 32
15 }
16
17 variable "storage_account_type" {
18   default = "Standard_LRS"
19 }
```

File: variables.tf

### Task 3: Розгортання шаблону через Terraform (замість PowerShell)

```
1  data "azurerm_resource_group" "existing" {
2    name = var.resource_group_name
3  }
4
5  resource "azurerm_managed_disk" "disk3" [
6    name          = var.disk_name
7    location      = data.azure_rm_resource_group.existing.location
8    resource_group_name = data.azure_rm_resource_group.existing.name
9    storage_account_type = var.storage_account_type
10   create_option     = "Empty"
11   disk_size_gb     = var.disk_size_gb
12
13   tags = {
14     environment = "lab"
15     task        = "task3"
16   }
17 ]
```

File: main.tf

```
1  output "disk_id" {
2    value = azurerm_managed_disk.disk3.id
3  }
4
5  output "disk_name" {
6    value = azurerm_managed_disk.disk3.name
7  }
```

File: outputs.tf

```
1  terraform {
2    required_providers {
3      azurerm = {
4        source  = "hashicorp/azurerm"
5        version = "~>3.0"
6      }
7    }
8  }
9
10 provider "azurerm" {
11   features {}
12 }
```

File: providers.tf

```
1  variable "resource_group_name" {
2    default = "az104-rg3"
3  }
4
5  variable "location" {
6    default = "East US"
7  }
8
9  variable "disk_name" {
10   default = "az104-disk3"
11 }
12
13 variable "disk_size_gb" {
14   default = 32
15 }
16
17 variable "storage_account_type" {
18   default = "Standard_LRS"
19 }
```

File: variables.tf

## Task 4: Розгортання ARM шаблону через Azure CLI

```
1  {
2      "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentParameters.json#",
3      "contentVersion": "1.0.0.0",
4      "parameters": {
5          "disk_name": {
6              "value": "az104-disk4"
7          },
8          "location": {
9              "value": "eastus"
10         },
11         "diskSizeGB": {
12             "value": 32
13         },
14         "diskSku": {
15             "value": "Standard_LRS"
16         }
17     }
18 }
```

File: parameters.json

```
[{"$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#", "contentVersion": "1.0.0.0", "parameters": { "disk_name": { "type": "string", "defaultValue": "az104-disk4" }, "location": { "type": "string", "defaultValue": "eastus" }, "diskSizeGB": { "type": "int", "defaultValue": 32 }, "diskSku": { "type": "string", "defaultValue": "Standard_LRS", "allowedValues": [ "Standard_LRS", "StandardSSD_LRS", "Premium_LRS" ] } }, "resources": [ { "type": "Microsoft.Compute/disks", "apiVersion": "2023-10-02", "name": "[parameters('disk_name')]", "location": "[parameters('location')]", "sku": { "name": "[parameters('diskSku')]" }, "properties": { "creationData": { "createOption": "Empty" }, "diskSizeGB": "[parameters('diskSizeGB')]" }, "tags": { "environment": "lab", "task": "task4" } } ], "outputs": { "diskId": { "type": "string", "value": "[resourceId('Microsoft.Compute/disks', parameters('disk_name'))]" }, "diskName": { "type": "string", "value": "[parameters('disk_name')]" } } } ]
```

File: templates.json

## Task 5: Розгортання ресурсу через Azure Bicep

```

lab3 > task5 > azurdeploydisk.bicep
1   @description('Name of the managed disk')
2   param managedDiskName string = 'az104-disk5'
3
4   @description('Location for the disk')
5   param location string = resourceGroup().location
6
7   @description('Size of the disk in GB')
8   param diskSizeinGiB int = 32
9
10  @description('Storage account type for the disk')
11  @allowed([
12    'Standard_LRS'
13    'StandardSSD_LRS'
14    'Premium_LRS'
15  ])
16  param diskSku string = 'StandardSSD_LRS'
17
18  resource managedDisk 'Microsoft.Compute/disks@2023-10-02' = {
19    name: managedDiskName
20    location: location
21    sku: {
22      name: diskSku
23    }
24    properties: {
25      creationData: {
26        createOption: 'Empty'
27      }
28      diskSizeGB: diskSizeinGiB
29    }
30    tags: {
31      environment: 'lab'
32      task: 'task5'
33    }
34  }
35
36  output diskId string = managedDisk.id
37  output diskName string = managedDisk.name
38  output diskSizeGB int = managedDisk.properties.diskSizeGB

```

```

illoi@illoi-Aspire-A715-42G:~/study/azure_labs/lab3/task5$ az disk list --resource-group az104-rg3 --output table
Name          ResourceGroup  Location  Zones  Sku           SizeGb  ProvisioningState
-----        -----
az104-disk1  az104-rg3    eastus    Standard_LRS  32     Succeeded
az104-disk2  az104-rg3    eastus    Standard_LRS  32     Succeeded
az104-disk3  az104-rg3    eastus    Standard_LRS  32     Succeeded
az104-disk4  az104-rg3    eastus    Standard_LRS  32     Succeeded
az104-disk5  az104-rg3    eastus    StandardSSD_LRS 32     Succeeded

```

## Висновок

У ході виконання роботи було проведено комплексне порівняння основних інструментів автоматизації в екосистемі Azure.

- Багатоплатформний підхід (Terraform): Реалізовано створення керованих дисків через Terraform, що дозволило оцінити переваги декларативного стилю та управління станом інфраструктури (state file).

- Шаблонізація ARM (JSON): Сформовано та розгорнуто ARM-шаблони для створення дисків, що продемонструвало "рідний" спосіб опису ресурсів Azure у форматі JSON.
- Azure Bicep: Опановано новітній інструмент Вісер, який пропонує значно простіший та лаконічніший синтаксис у порівнянні з класичними ARM-шаблонами, забезпечуючи ту саму функціональність.
- Практична валідація: За допомогою Azure CLI було успішно перевірено наявність та статус розгорнутих дисків (az104-disk1 — az104-disk5), що підтвердило коректність роботи всіх використаних методів автоматизації .

Гітхаб: [https://github.com/Iolloizaur/azure\\_labs/tree/main/lab3](https://github.com/Iolloizaur/azure_labs/tree/main/lab3)