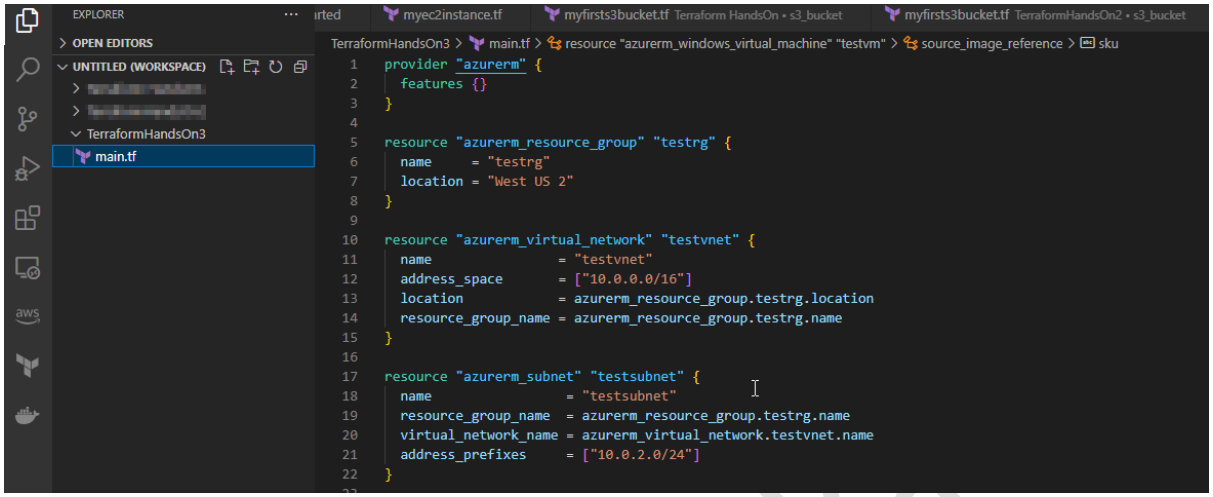
Creation of AzureVM resource set via Terraform

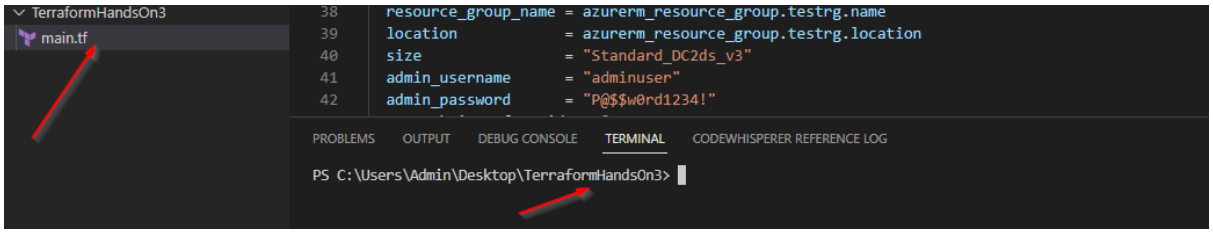
Create new folder **TerraformHandsOn3** and the **main.tf** file for creating the Azure VM



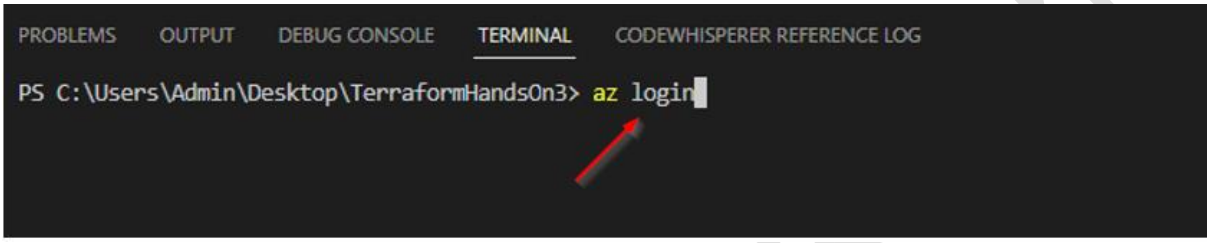

```
resource "azurerm_network_interface" "testnic" {
  name                = "testnic"
  location            = azurerm_resource_group.testrg.location
  resource_group_name = azurerm_resource_group.testrg.name

  ip_configuration {
    name                          = "internal"
    subnet_id                     = azurerm_subnet.testsubnet.id
    private_ip_address_allocation = "Dynamic"
  }
}
```

```
resource "azurerm_windows_virtual_machine" "testvm" {
  name                = "testvm"
  resource_group_name = azurerm_resource_group.testrg.name
  location            = azurerm_resource_group.testrg.location
  size                = "Standard_DC2ds_v3"
  admin_username      = "adminuser"
  admin_password      = "P@$$w0rd1234!"
  network_interface_ids = [
    azurerm_network_interface.testnic.id,
  ]

  os_disk {
    caching              = "ReadWrite"
    storage_account_type = "Standard_LRS"
  }

  source_image_reference {
    publisher = "MicrosoftWindowsServer"
    offer     = "WindowsServer"
    sku       = "2019-datacenter-gensecond"
    version   = "latest"
  }

}
```

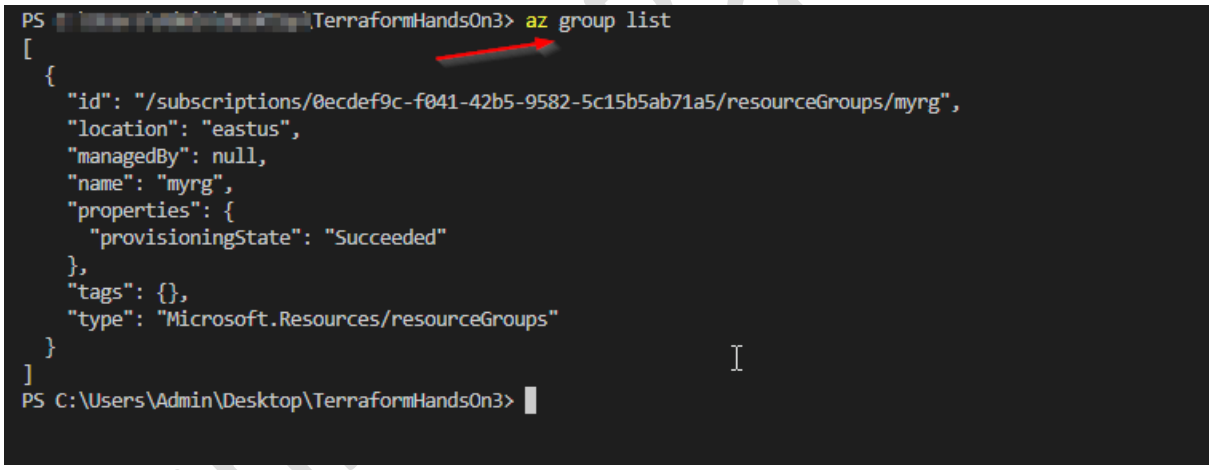
In the terminal, change to the directory where the **main.tf** is located as pointed below



First command to execute is to login to the azure account from the terminal as shown below, this will invoke the azure portal login page, once logged in with username and password the terminal will be integrated with azure account.



To test the azure portal integration with vscode terminal execute the below command to list the resource group available in your linked account



The first terraform command to be executed is **terraform init** to initialize the directory **TerraformHandOn3**

```
PS [redacted]\TerraformHandsOn3> terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/azurerm...
- Installing hashicorp/azurerm v3.38.0...
- Installed hashicorp/azurerm v3.38.0 (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.
PS [redacted]\TerraformHandsOn3>
```

Next 2 commands we should execute as below to check the code format and validate the configuration

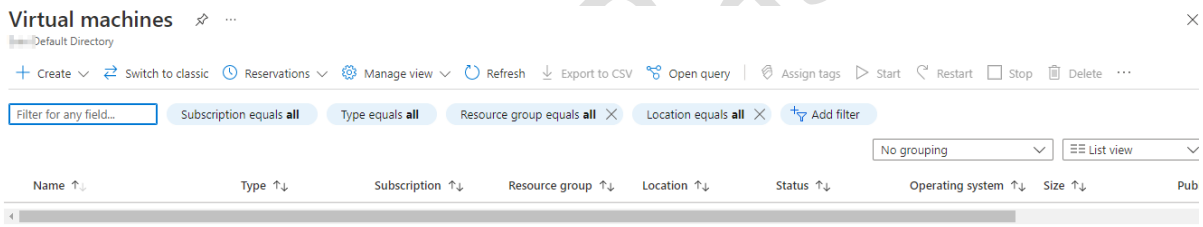
```
PS [redacted]\TerraformHandsOn3> terraform fmt
PS [redacted]\TerraformHandsOn3> terraform validate
Success! The configuration is valid.

PS [redacted]\TerraformHandsOn3>
```

The next command to be executed is terraform plan which generate the plan for the resource creation

```
PS [redacted]\TerraformHandsOn3> terraform plan

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_network_interface.testnic will be created
  + resource "azurerm_network_interface" "testnic" {
      + applied_dns_servers           = (known after apply)
      + dns_servers                   = (known after apply)
      + enable_accelerated_networking = false
      + enable_ip_forwarding          = false
      + id                            = (known after apply)
      + internal_dns_name_label       = (known after apply)
      + internal_domain_name_suffix   = (known after apply)
      + location                      = "westus2"
      + mac_address                   = (known after apply)
      + name                          = "testnic"
      + private_ip_address            = (known after apply)
      + private_ip_addresses          = (known after apply)
      + resource_group_name           = "testrg"
      + virtual_machine_id            = (known after apply)
```

```
    + os_disk {
        + caching                   = "ReadWrite"
        + disk_size_gb              = (known after apply)
        + name                      = (known after apply)
        + storage_account_type      = "Standard_LRS"
        + write_accelerator_enabled = false
      }

    + source_image_reference {
        + offer     = "WindowsServer"
        + publisher = "MicrosoftWindowsServer"
        + sku       = "2019-datacenter-gensecond"
        + version   = "latest"
      }

    + termination_notification {
        + enabled = (known after apply)
        + timeout = (known after apply)
      }
  }

Plan: 5 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run
"terraform apply" now.
PS [redacted]\TerraformHandsOn3>
```

Before creating the resource through terraform, azure portal is verified for any VMs exist

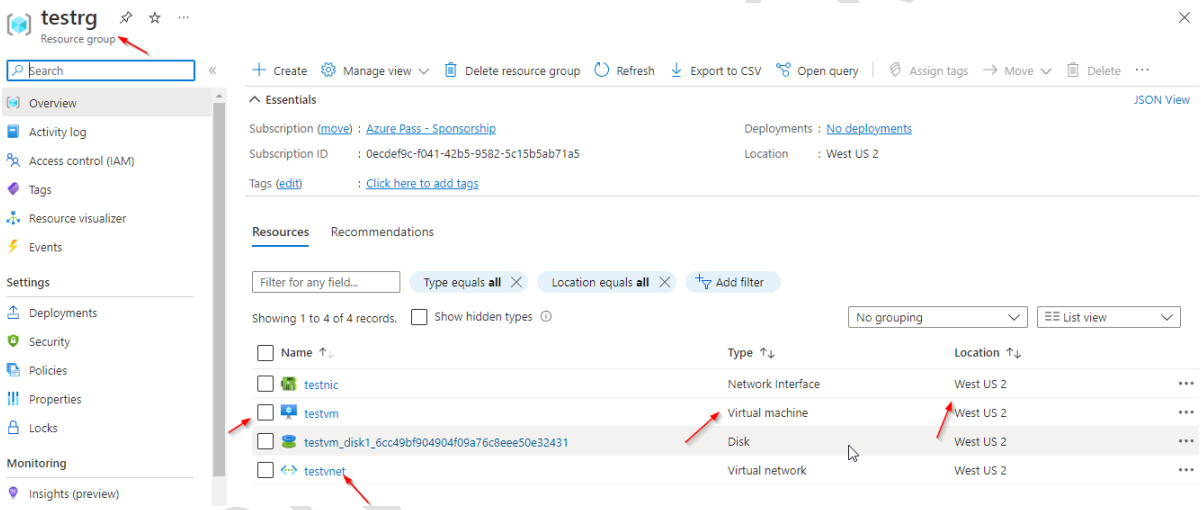


Now the next command is to execute is the terraform apply –auto-approve to actually create in the azure platform

```
PS [redacted]\TerraformHandsOn3> terraform apply -auto-approve

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_network_interface.testnic will be created
  + resource "azurerm_network_interface" "testnic" {
      + applied_dns_servers           = (known after apply)
      + dns_servers                   = (known after apply)
      + enable_accelerated_networking = false
      + enable_ip_forwarding          = false
      + id                            = (known after apply)
```

```
Plan: 5 to add, 0 to change, 0 to destroy.
azurerm_resource_group.testrg: Creating...
azurerm_resource_group.testrg: Creation complete after 4s [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups
/testrg]
azurerm_virtual_network.testvnet: Creating...
azurerm_virtual_network.testvnet: Still creating... [10s elapsed]
azurerm_virtual_network.testvnet: Creation complete after 14s [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGr
oups/testrg/providers/Microsoft.Network/virtualNetworks/testvnet]
azurerm_subnet.testsubnet: Creating...
azurerm_subnet.testsubnet: Creation complete after 9s [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/tes
trg/providers/Microsoft.Network/virtualNetworks/testvnet/subnets/testsubnet]
azurerm_network_interface.testnic: Creating...
azurerm_network_interface.testnic: Still creating... [10s elapsed]
azurerm_network_interface.testnic: Creation complete after 11s [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceG
roups/testrg/providers/Microsoft.Network/networkInterfaces/testnic]
azurerm_windows_virtual_machine.testvm: Creating...
azurerm_windows_virtual_machine.testvm: Still creating... [10s elapsed]
azurerm_windows_virtual_machine.testvm: Still creating... [20s elapsed]
azurerm_windows_virtual_machine.testvm: Still creating... [30s elapsed]
azurerm_windows_virtual_machine.testvm: Still creating... [40s elapsed]
azurerm_windows_virtual_machine.testvm: Still creating... [50s elapsed]
azurerm_windows_virtual_machine.testvm: Creation complete after 51s [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/reso
urceGroups/testrg/providers/Microsoft.Compute/virtualMachines/testvm]

Apply complete! Resources: 5 added, 0 changed, 0 destroyed.
```

Verified the changes applied in the azure portal as highlighted below



Once when the resource is not required, terraform destroy –auto-approve command is executed to remove all resources from azure account

```
PS [redacted]\TerraformHandsOn3> terraform destroy -auto-approve
azurerm_resource_group.testrg: Refreshing state... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
]
azurerm_virtual_network.testvnet: Refreshing state... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/tes
trg/providers/Microsoft.Network/virtualNetworks/testvnet]
azurerm_subnet.testsubnet: Refreshing state... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg/pro
viders/Microsoft.Network/virtualNetworks/testvnet/subnets/testsubnet]
azurerm_network_interface.testnic: Refreshing state... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/te
strg/providers/Microsoft.Network/networkInterfaces/testnic]
azurerm_windows_virtual_machine.testvm: Refreshing state... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGrou
ps/testrg/providers/Microsoft.Compute/virtualMachines/testvm]

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

Terraform will perform the following actions:

  # azurerm_network_interface.testnic will be destroyed
  - resource "azurerm_network_interface" "testnic" {
      - applied_dns_servers           = [] -> null
      - dns_servers                   = [] -> null
      - enable_accelerated_networking = false -> null
      - enable_ip_forwarding          = false -> null
      - id                            = "/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg/providers/Mic
```

```
Plan: 0 to add, 0 to change, 5 to destroy.
azurerm_windows_virtual_machine.testvm: Destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/tes
trg/providers/Microsoft.Compute/virtualMachines/testvm]
azurerm_windows_virtual_machine.testvm: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-...crosoft.Compute/virtua
lMachines/testvm, 10s elapsed]
azurerm_windows_virtual_machine.testvm: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-...crosoft.Compute/virtua
lMachines/testvm, 20s elapsed]
azurerm_windows_virtual_machine.testvm: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-...crosoft.Compute/virtua
lMachines/testvm, 30s elapsed]
azurerm_windows_virtual_machine.testvm: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-...crosoft.Compute/virtua
lMachines/testvm, 40s elapsed]
azurerm_windows_virtual_machine.testvm: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-...crosoft.Compute/virtua
lMachines/testvm, 50s elapsed]
azurerm_windows_virtual_machine.testvm: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-...crosoft.Compute/virtua
lMachines/testvm, 1m0s elapsed]
azurerm_windows_virtual_machine.testvm: Destruction complete after 1m1s
azurerm_network_interface.testnic: Destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg/p
roviders/Microsoft.Network/networkInterfaces/testnic]
azurerm_network_interface.testnic: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-...soft.Network/networkInterfa
ces/testnic, 10s elapsed]
azurerm_network_interface.testnic: Destruction complete after 13s
azurerm_subnet.testsubnet: Destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg/providers
/Microsoft.Network/virtualNetworks/testvnet/subnets/testsubnet]
```

```
azurerm_virtual_network.testvnet: Destruction complete after 13s
azurerm_resource_group.testrg: Destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg]
azurerm_resource_group.testrg: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
, 10s elapsed]
azurerm_resource_group.testrg: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
, 20s elapsed]
azurerm_resource_group.testrg: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
, 30s elapsed]
azurerm_resource_group.testrg: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
, 40s elapsed]
azurerm_resource_group.testrg: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
, 50s elapsed]
azurerm_resource_group.testrg: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
, 1m0s elapsed]
azurerm_resource_group.testrg: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
, 1m10s elapsed]
azurerm_resource_group.testrg: Still destroying... [id=/subscriptions/0ecdef9c-f041-42b5-9582-5c15b5ab71a5/resourceGroups/testrg
, 1m20s elapsed]
azurerm_resource_group.testrg: Destruction complete after 1m23s

Destroy complete! Resources: 5 destroyed.
```

Below is the exract from the **resource group visualizer** tool to see the entire architecture that was created though terraform code as above