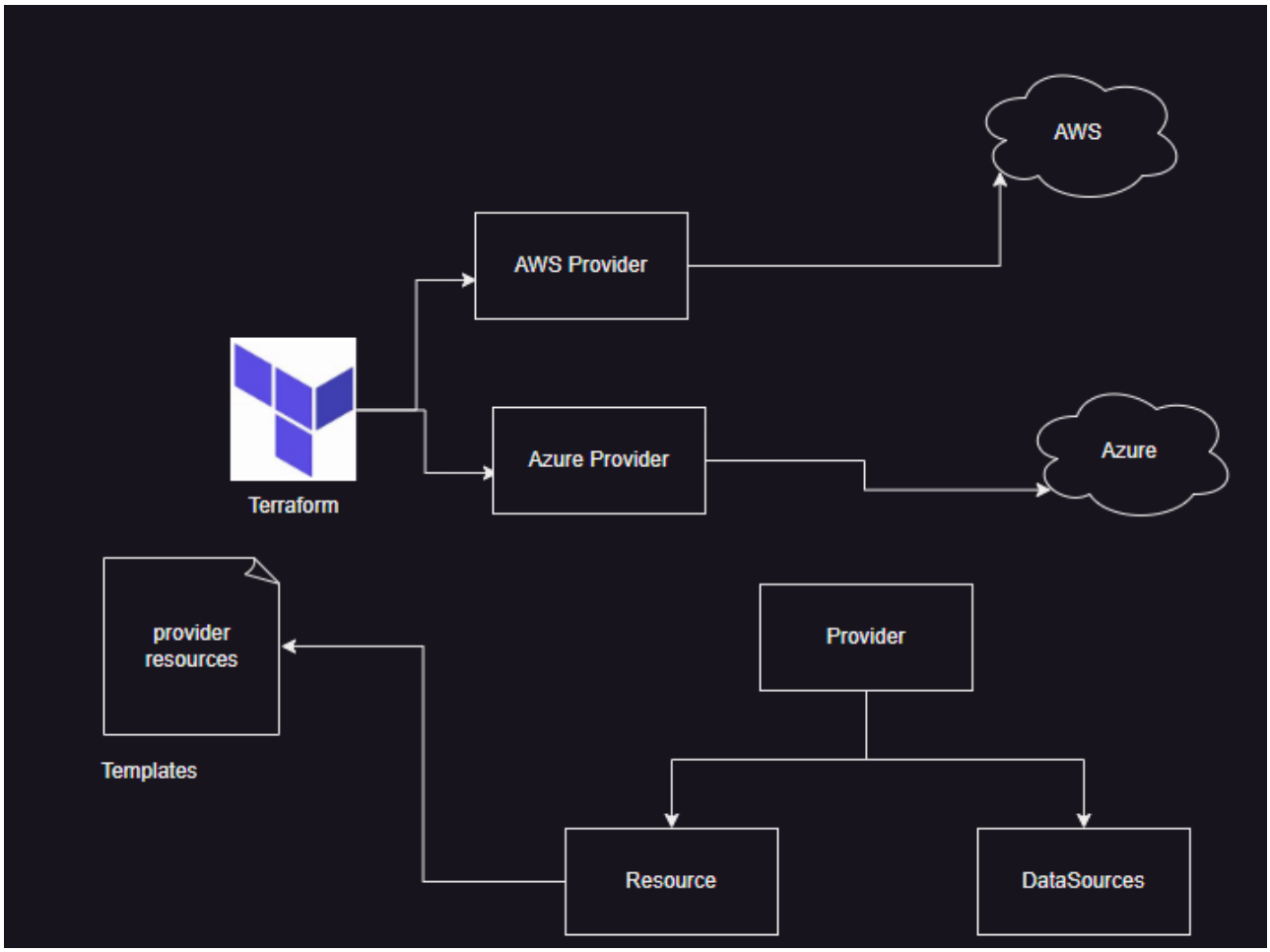
How Terraform Works

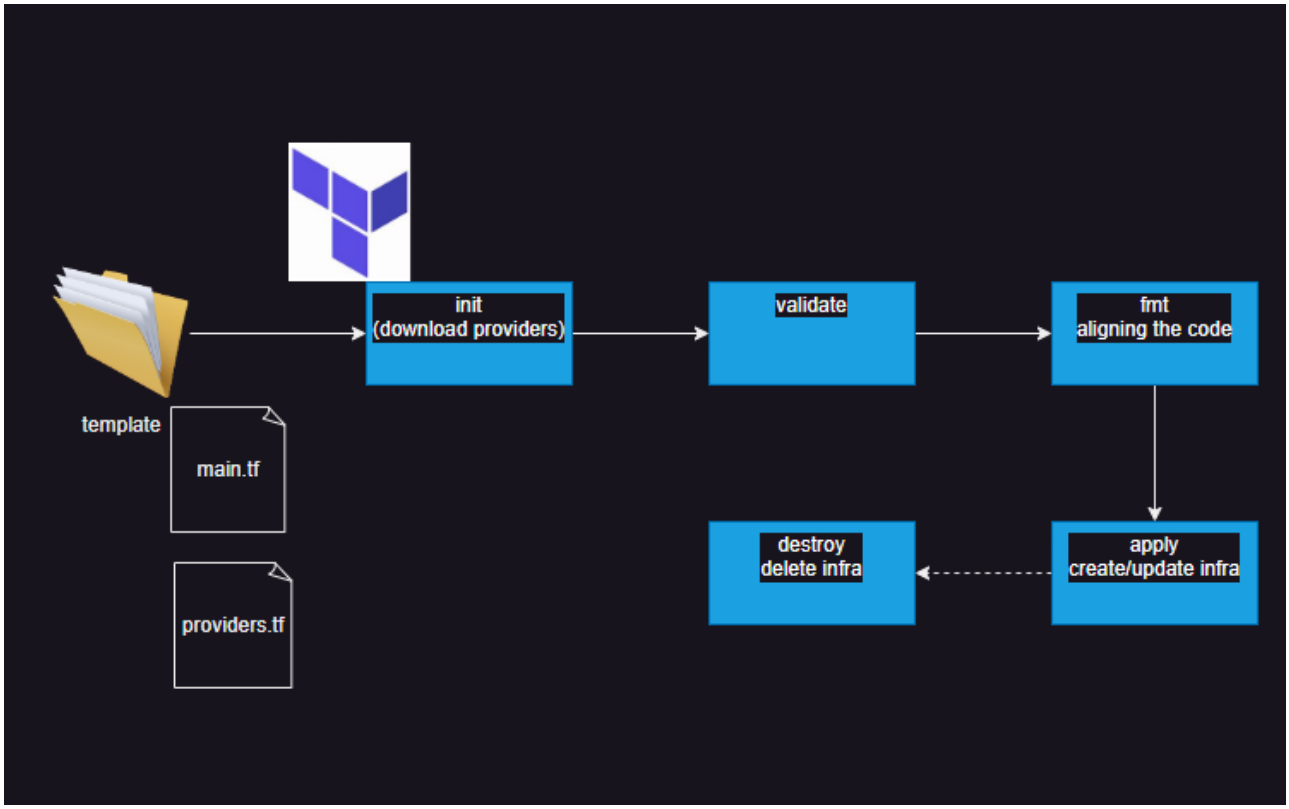
- Overview



- Terraform is a single executable.
- Terraform has the capability to communicate with multiple clouds or virtualization platforms to create infra via providers
- Each provdier has
  - Resources: What can be created
  - Datasources: query the cloud/virtualization platform
- We would be creating templates where we define
  - provider
  - resources
- For authoring Templates, Terraform uses declarative approach via a language called as Hashicorp Configuration Language (HCL).

Workflow

- Terraform workflow

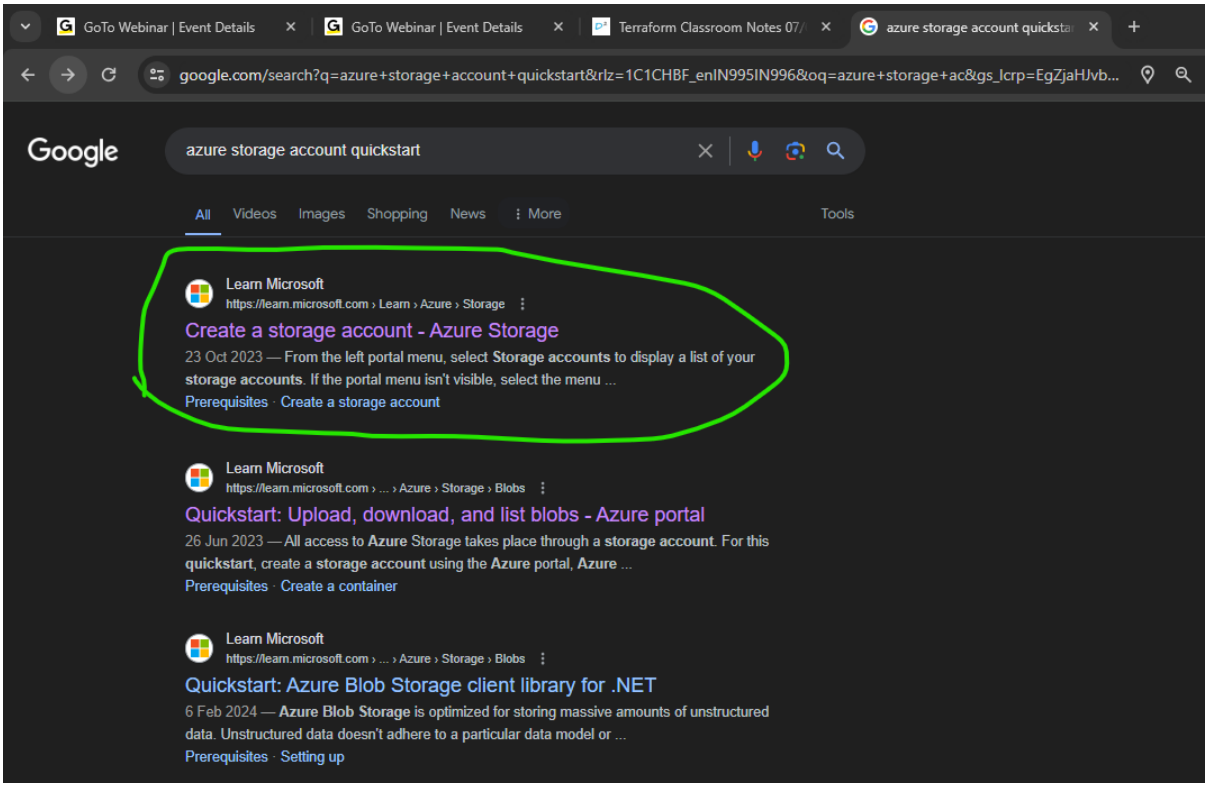


Setting up Terraform

- Downloading terraform Refer Here

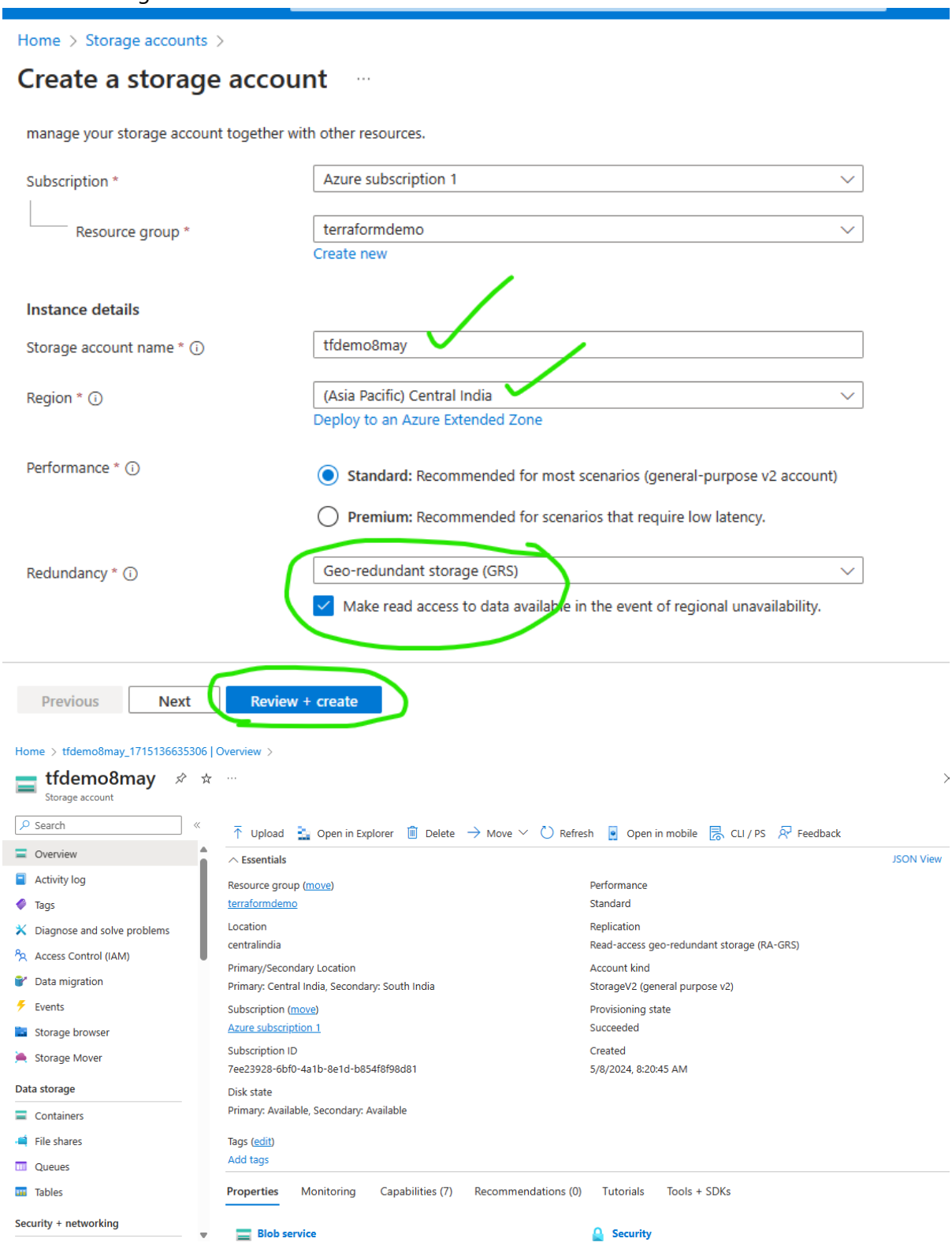
Activity 1: Creating a storage account in Azure

- First Step:
  - Figure out manual steps Refer Here



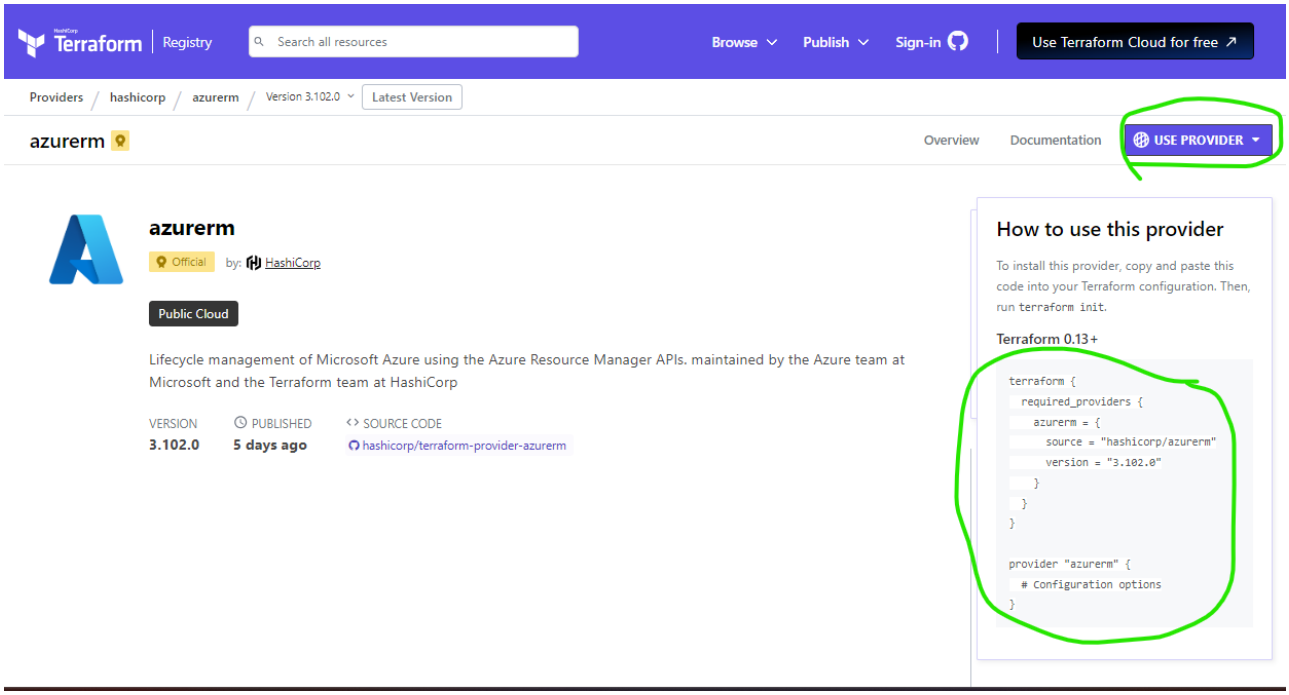
  - Azure needs a resource group

- Create a storage account



- Roughly our inputs
  - resource group name
  - location
  - storage account name
  - Redundancy level
- Create a folder `storageaccount` and create a file called as `main.tf`.

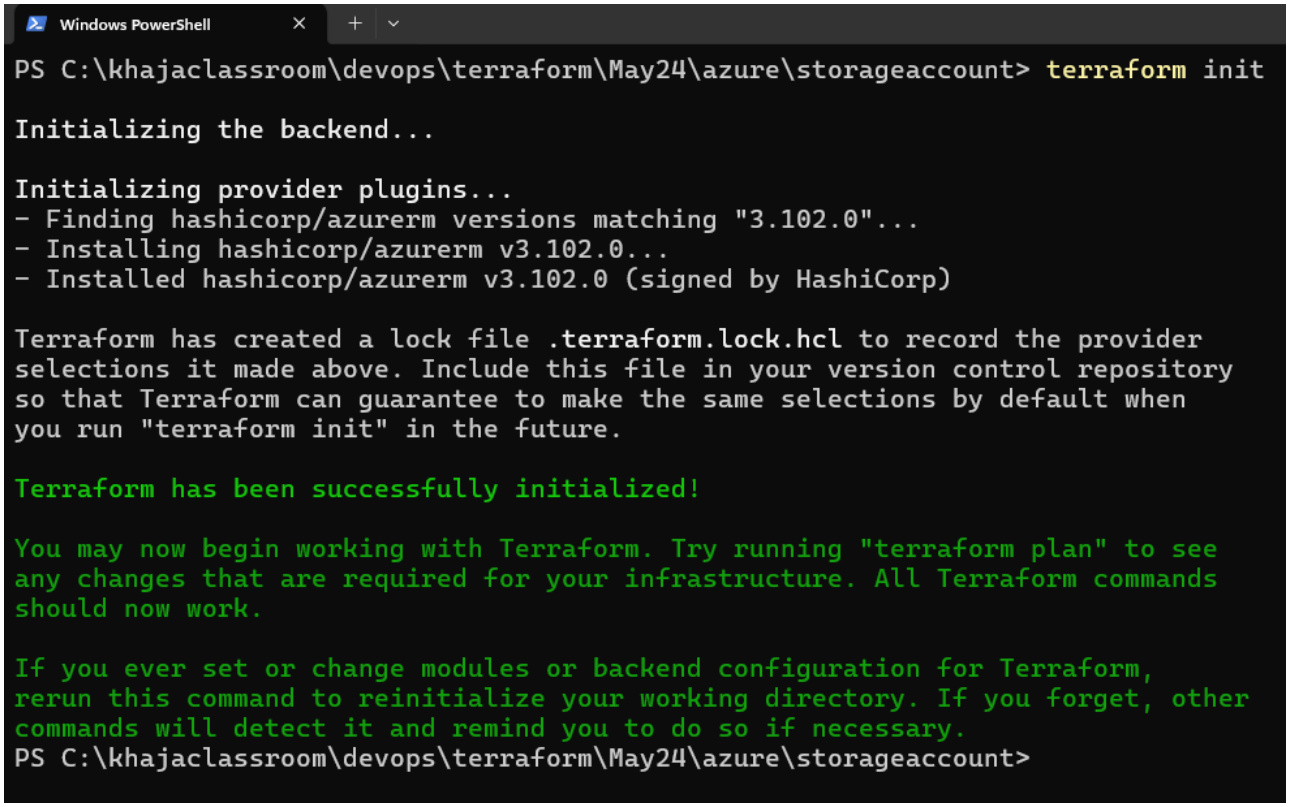
- Now lets configure provider Refer Here



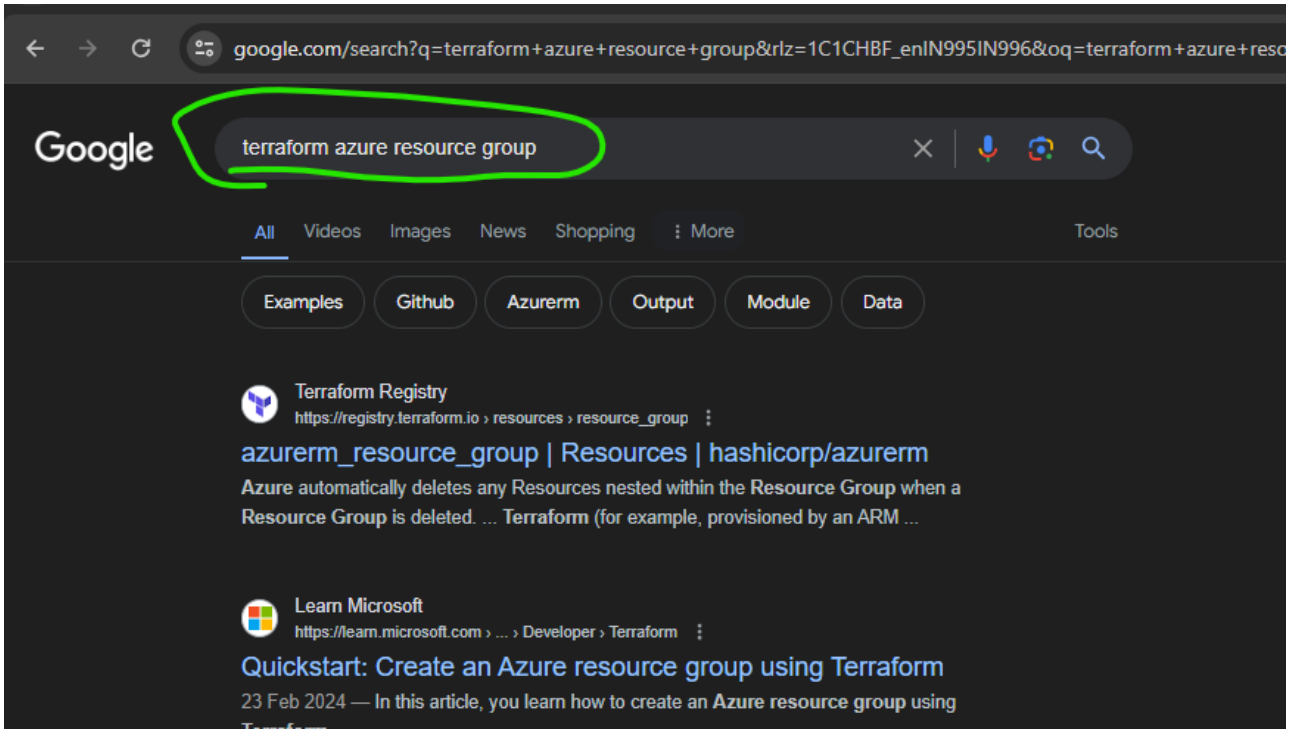
- The template will be as shown below

```
terraform {
  required_providers {
    azurerm = {
      source = "hashicorp/azurerm"
      version = "3.102.0"
    }
  }
}

provider "azurerm" {
  features {

  }
}
```

- Now execute `terraform init`



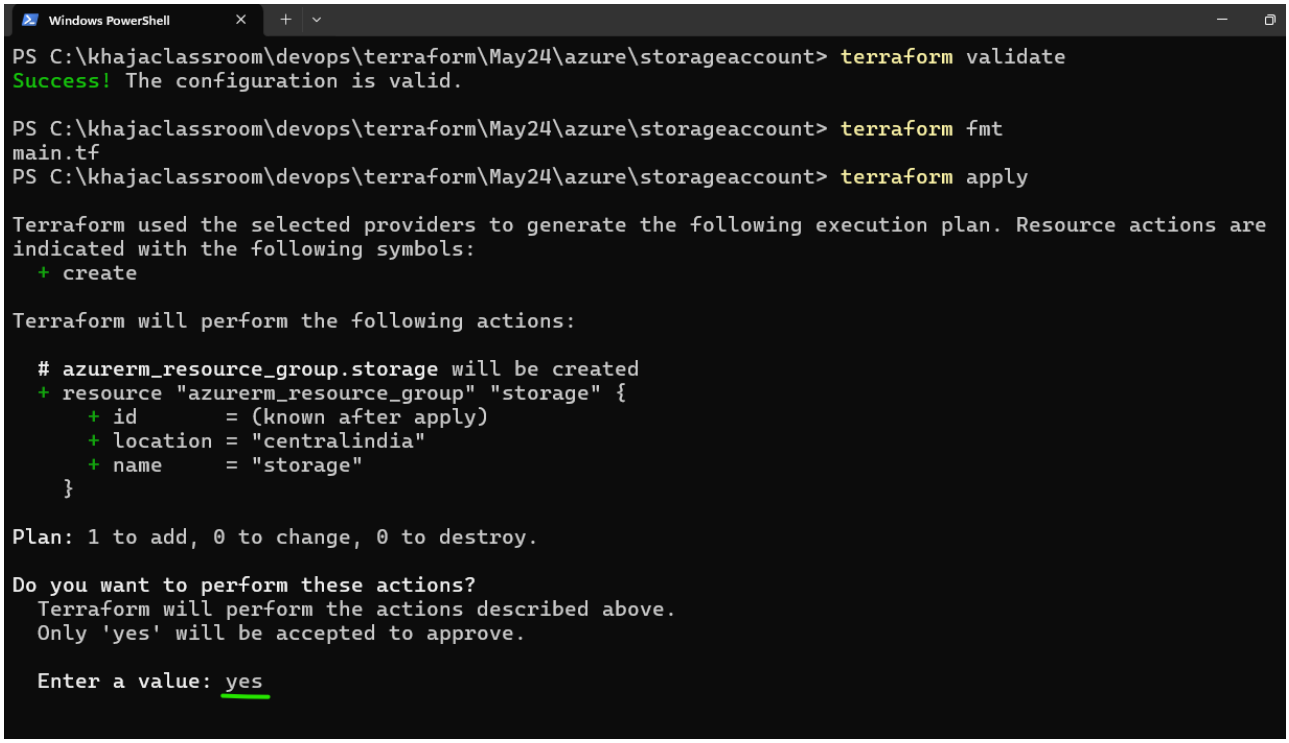
- Terraform provider is downloaded, we need to configure credentials, easiest way to configure credentials is download azure cli Refer Here
- Once azure cli is installed, launch terminal and execute `az login`
- lets write a resource for resource group



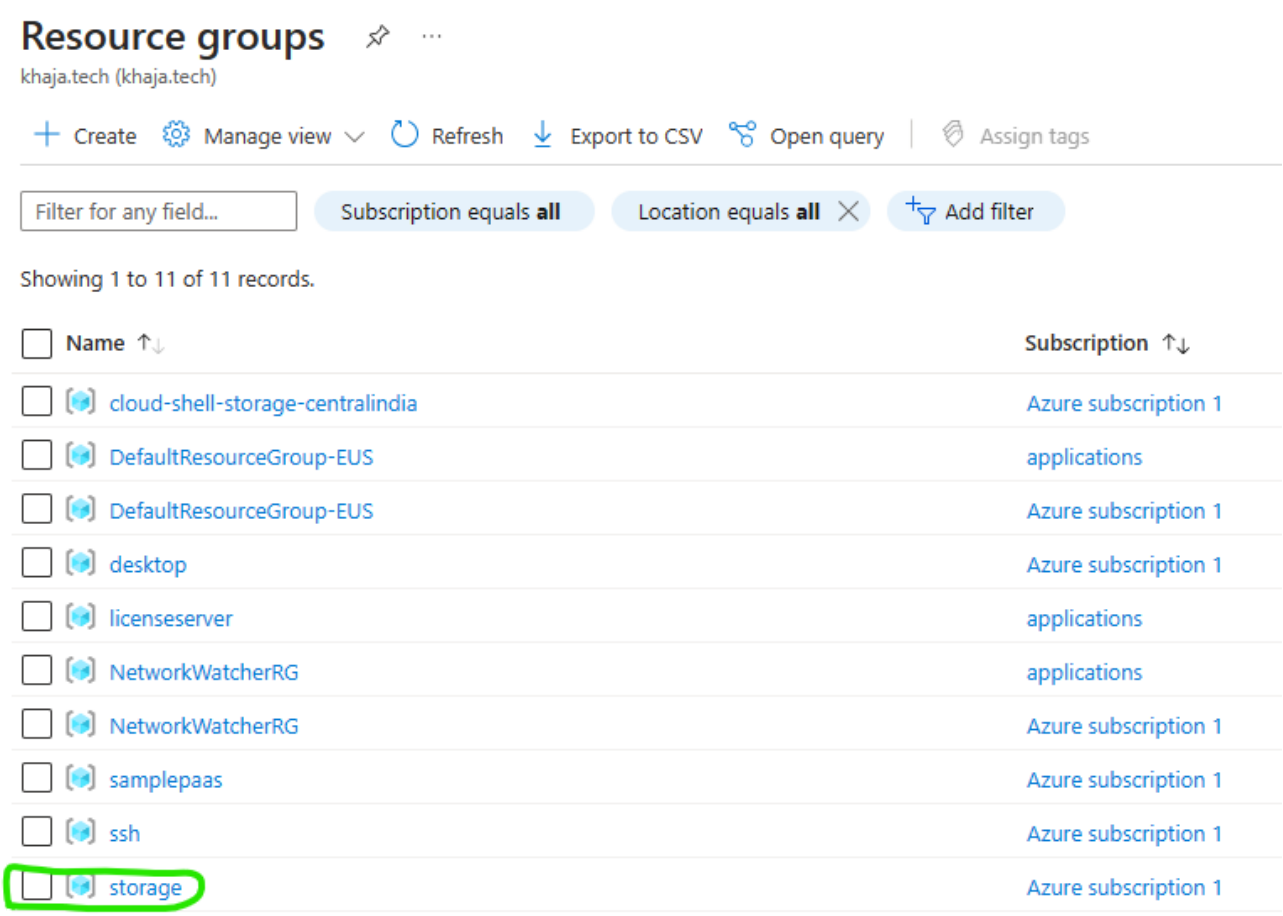
- Now declare the resource

```
resource "azurerm_resource_group" "storage" {
  name     = "storage"
  location = "Central India"
}
```

- Execute the workflow



```
main.tf
PS C:\khajaclassroom\devops\terraform\May24\azure\storageaccount> terraform apply

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_resource_group.storage will be created
  + resource "azurerm_resource_group" "storage" {
      + id       = (known after apply)
      + location = "centralindia"
      + name     = "storage"
    }

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_resource_group.storage: Creating...
azurerm_resource_group.storage: Creation complete after 9s [id=/subscriptions/7ee23928-6bf0-4a1b-8e1d-b8
54f8f98d81/resourceGroups/storage]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PS C:\khajaclassroom\devops\terraform\May24\azure\storageaccount>
```



- To be continued in next session