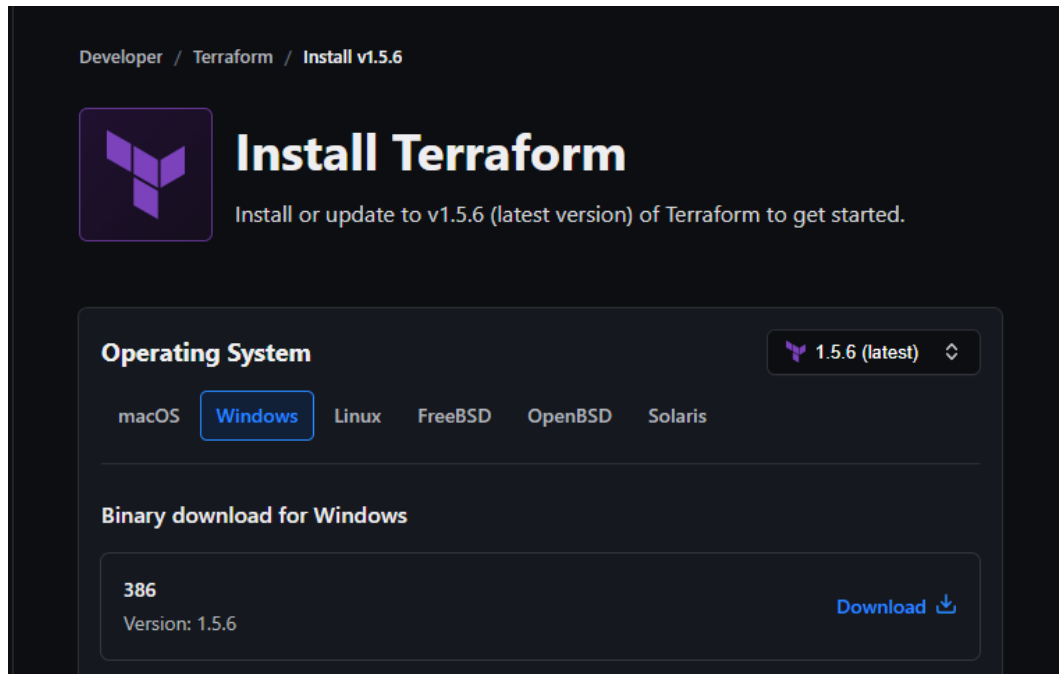
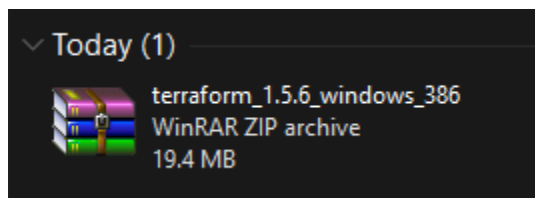


7. Laboratory Exercise

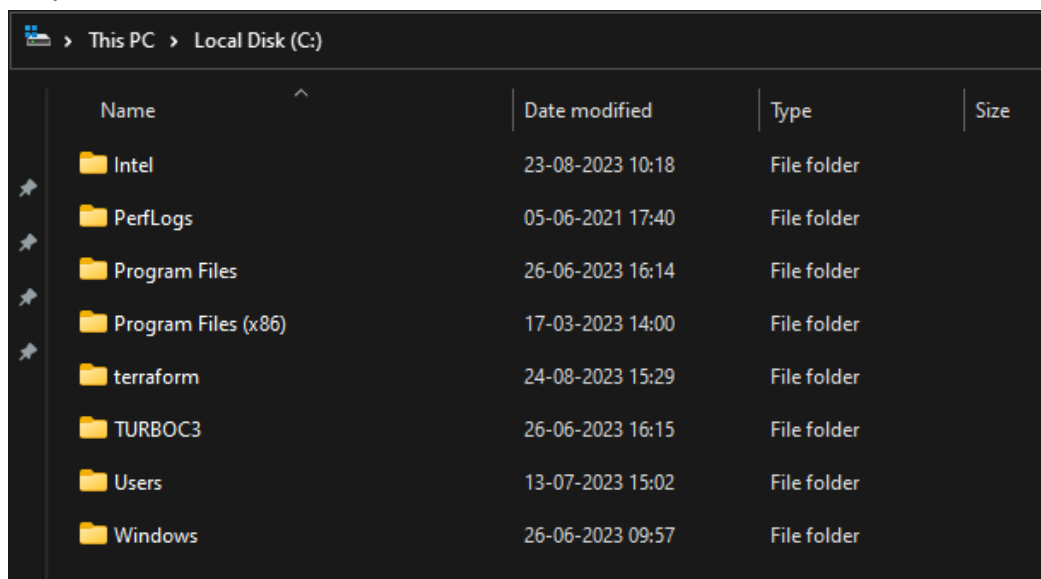
Step 1 : Download appropriate terraform package(.zip) from terraform.io/downloads.html for Windows.



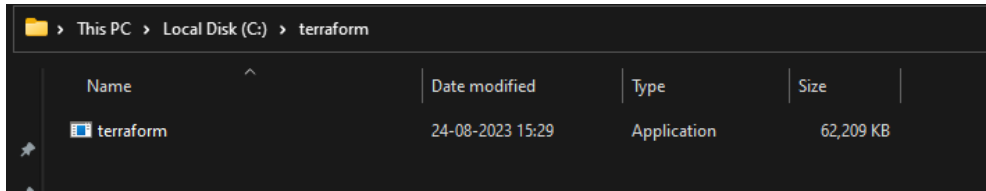
Step 2: Download Terraform for Windows 64-bit / (32-bit).



Step 3: Create a folder 'terraform' in drive C .

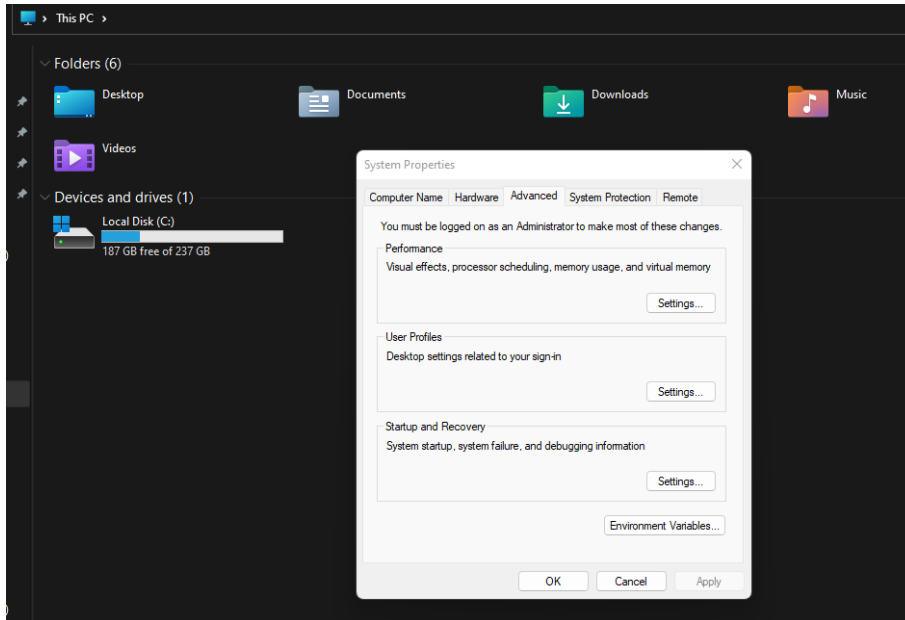


Step 4 : Extract downloaded zip in to this c:/terraform folder

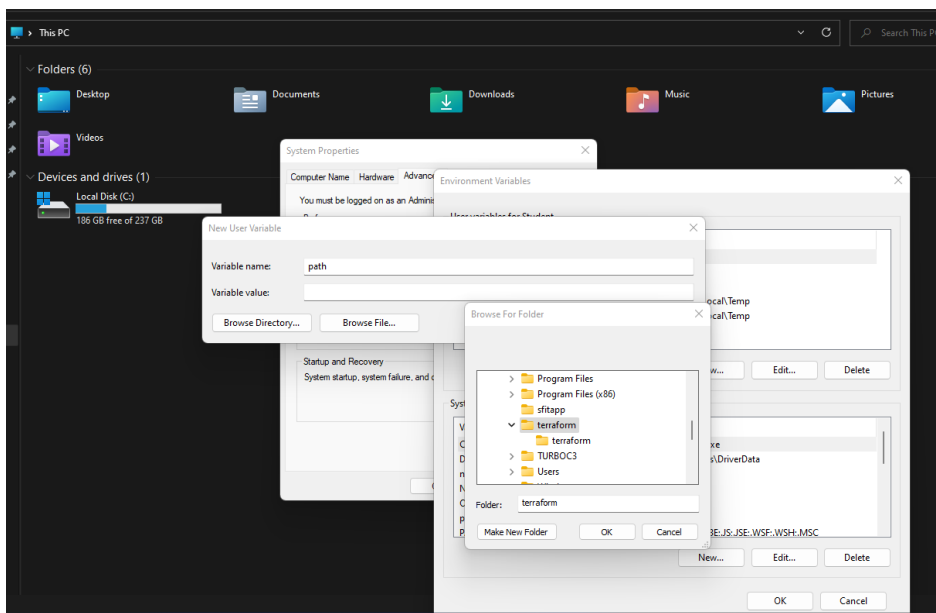


Step 5: Now we need to set a path for terraform. Go to My computer/ This PC, right click, select properties, go to advanced system settings.

Step 6: click on environment variable

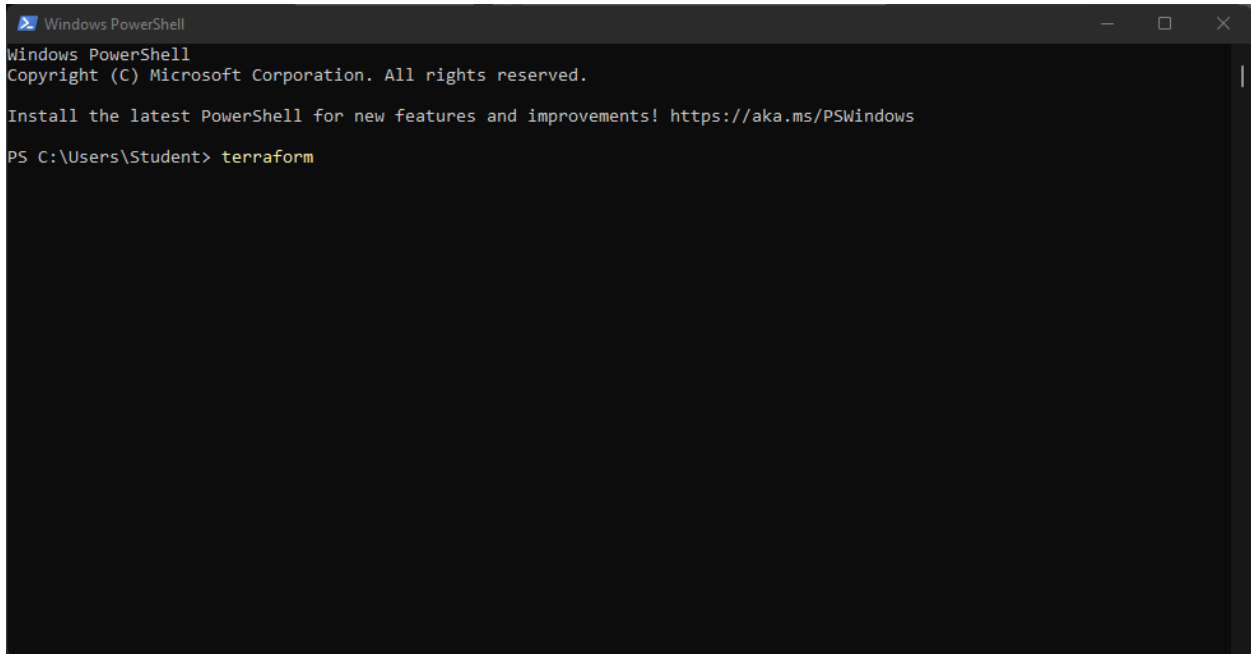


Step 7: click on New, give variable name = Path, click on browse directory, select c:/terraforms/terra....exe, OK



Step 8: Cross verify terraform installed properly or not . go to MS Powershell, run as a administrator

Step 9: Type terraform

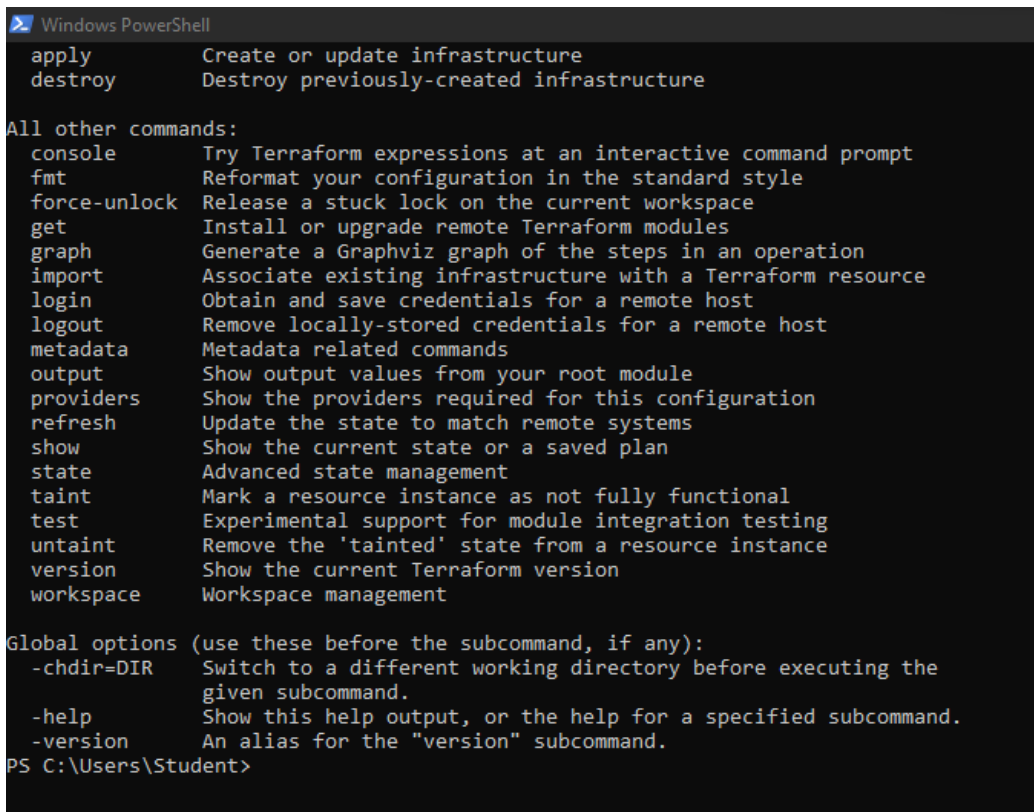


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Student> terraform
```

Step 10: You will find init, validate, plan, apply and destroy options means you have installed terraform successfully.



```
Windows PowerShell

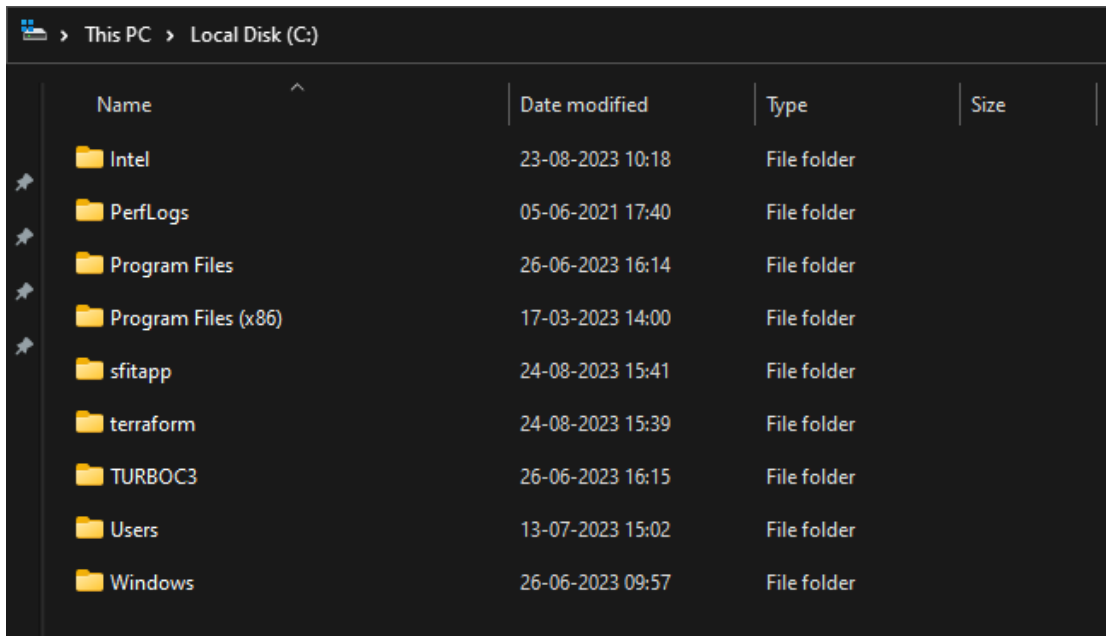
  apply      Create or update infrastructure
  destroy    Destroy previously-created infrastructure

All other commands:
  console    Try Terraform expressions at an interactive command prompt
  fmt        Reformat your configuration in the standard style
  force-unlock Release a stuck lock on the current workspace
  get        Install or upgrade remote Terraform modules
  graph      Generate a Graphviz graph of the steps in an operation
  import     Associate existing infrastructure with a Terraform resource
  login      Obtain and save credentials for a remote host
  logout     Remove locally-stored credentials for a remote host
  metadata   Metadata related commands
  output     Show output values from your root module
  providers  Show the providers required for this configuration
  refresh    Update the state to match remote systems
  show       Show the current state or a saved plan
  state      Advanced state management
  taint      Mark a resource instance as not fully functional
  test       Experimental support for module integration testing
  untaint    Remove the 'tainted' state from a resource instance
  version    Show the current Terraform version
  workspace  Workspace management

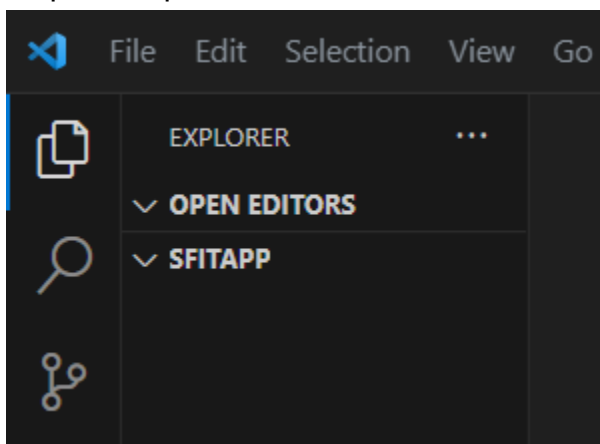
Global options (use these before the subcommand, if any):
  -chdir=DIR  Switch to a different working directory before executing the
              given subcommand.
  -help       Show this help output, or the help for a specified subcommand.
  -version    An alias for the "version" subcommand.

PS C:\Users\Student>
```

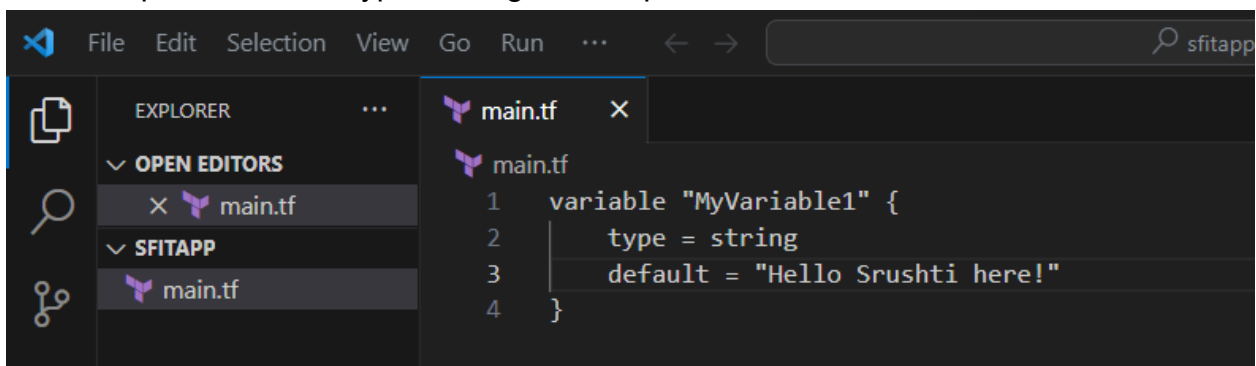
Step 11: Create a folder c:\SfitApp



Step 12 : Open Atom/VS CODE Editor ...Open folder SfitApp



Step 13: write main.tf file with input variables. The input variables, like the one above, use a couple of different types: string, list, map, and Boolean.

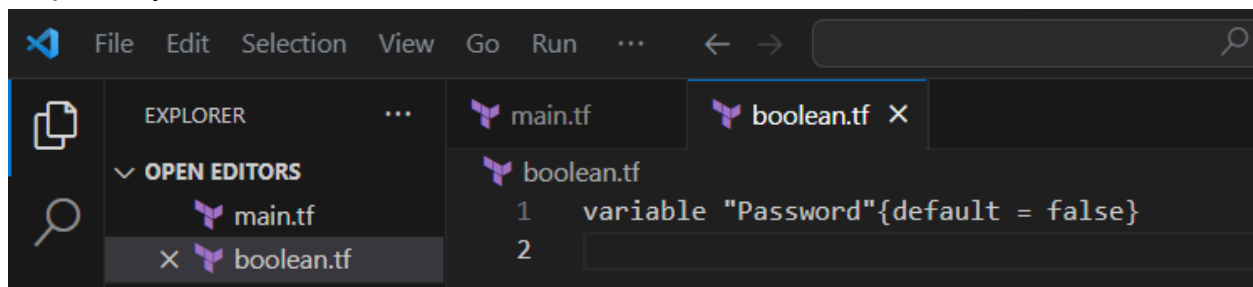


Step 14: Check the output on command Prompt...Go to C:\SfitApp, Type Terraform Console

You will get terraform prompt, run the .tf with var.MyVariable1, You will get welcome to SFIT msg.

```
PS C:\Users\Student> cd ..
PS C:\Users> cd ..
PS C:\> cd sfitapp
PS C:\sfitapp> terraform console
> var.MyVariable1
"Hello Srushti here!"
>
```

Step 15: try Boolean variable...Create Boolean.tf



```
PS C:\sfitapp> terraform console
> var.Password
false
>
```