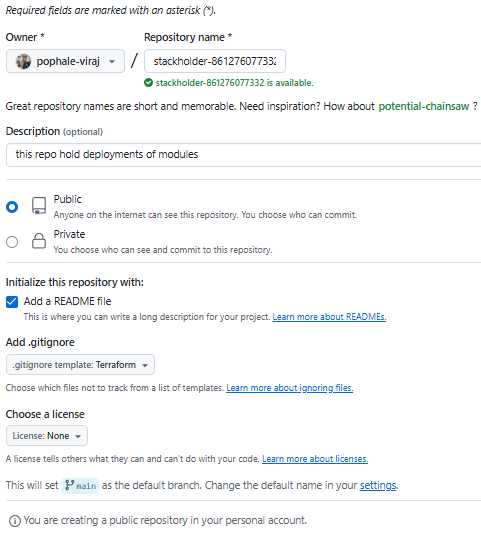
Modules in Terraform

By using terraform modules, we don’t want to repeat the code 100 times; avoiding graphical, use code, and use modules to reduce code repetition. Always a standard of implementation of resource

Create two github repositories using settings: tf\_aws\_module, stackholder\_861276077332



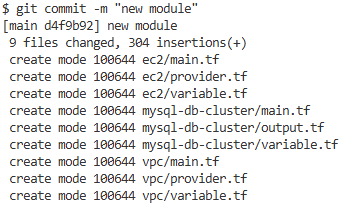
Clone the repositories into local machine at Desktop

Add the folder workspace to the VSC >> add modules ec2, vpc, mysql-db-cluster (I want to create ec2-et-al module in terraform in chatgpt) to tf\_aws\_module

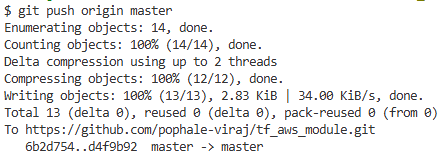
Add resource, data {} to main.tf. Add provider, terraform {} to provider.tf. Add variable {} to variable to tf. Add output {} to output.tf for db cluster module for check DB related arguments & values

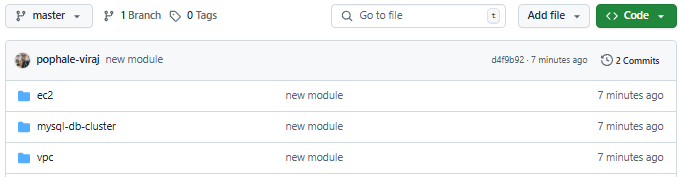
Push the code to git

* Move to tf\_aws\_module
* Git pull origin master
* Git add –all
* Git commit –m “new module”

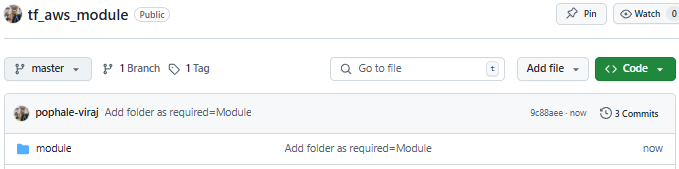


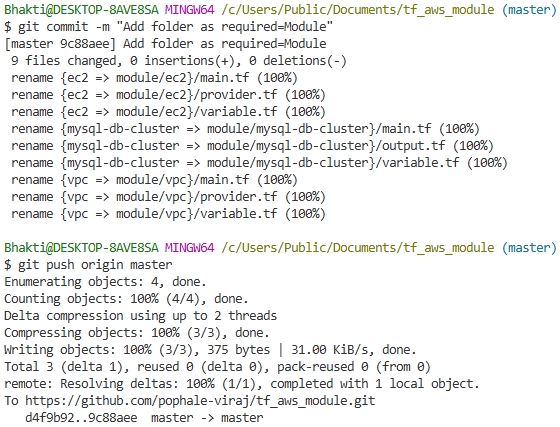
* Git push origin master





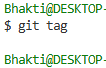
If not done, mention folder='module" in VSC as per Sir to clear follow-ups



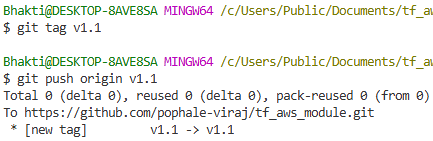
......IMP

Start repo versioning by adding tag to source code.

* Git tag ......checks the tag in repo



* Git tag v1.1
* Git push origin v1.1

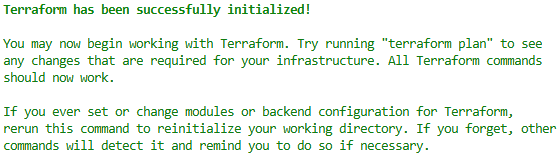


Use separate workspace stackholder to deploy modules

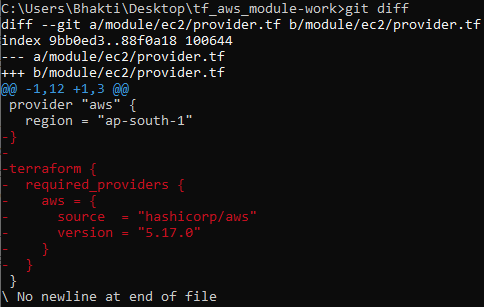
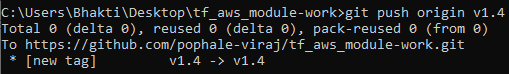
Add the folder workspace to the VSC=” stackholder-861276077332” >> Add modules main.tf, variable.tf, terraform.tfvars

Before running, understand this: We are not supposed to hardcode anything in codes. Infra deployment happens on the specified modules which are provided to the deployment in tfvars. Elaborate the sentence keeping in mind the picture of the module(tf\_aws\_module) and deployment(stackholder\_861276077332)

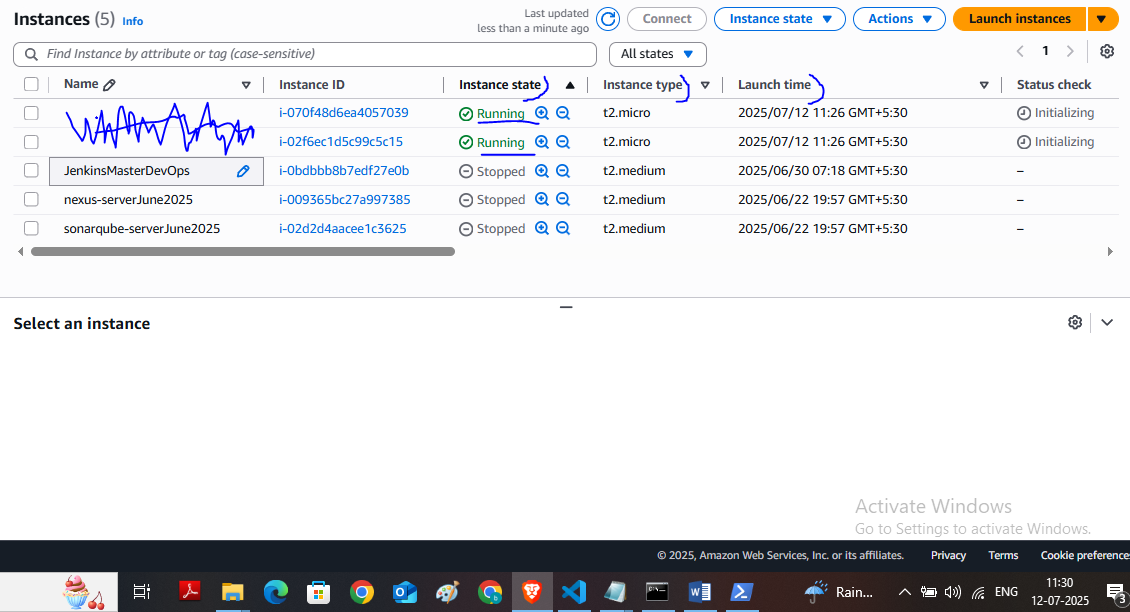
* Cd C:\Users\Bhakti\Desktop\stackholder-861276077332-work\ec2-instance
* Terraform fmt
* Terraform init ......gets the modules in .terraform from module workspace. To explain, Resource block is in terraform module. In stackholder workspace only, we are passing arguments & variables and calling the module workspace



>> If you make changes to module, push to the repo from cmd. Do git tag versioning tags, change source version tag, do terraform init & plan to update

* Terraform fmt
* Terraform init
* Terraform plan
* Terraform apply –auto-approve



* Terraform destroy

