Jenkins pipeline for terraform

Goto VSCode >> Create folder Jenkins-pipeline >> file main.tf, variable.tf, provider.tf, backend.tf, terraform.tfvars, .gitignore, Jenkinsfile

>> Start instance JenkinsMasterDevOps in AWS

Goto VSCode>>

>> Add provider {}, terraform {} to provider.tf

>> Add data {}, resource {} to main.tf

>> Add variable {} to variable.tf

>> Add variables arguments and values to terraform.tfvars

>> Add backend {} to backend.tf

* Cd Desktop/june2025-terraformlab/Jenkins-pipeline
* Git init



>> Create a repository in github=”Jenkins-pipeline”

>> Get Jenkinsfile pipeline page from <https://github.com/satyam88/terraform-deployment-aws.git> and add to Jenkinsfile

>> Goto IAM >> Create user=’indigo-aws’ >> Attach administratoraccess >> Next >> Create user >> Security credentials >> Application running on an AWS compute resource >> Access key & Secret access key >> access key=AKIA4RCAN7UKPWSNXQ5C, secret access key=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

>> Jenkins >> update ip-address >> Manage Jenkins >> new credentials >> AWS credentials >> indigo-aws, accesskey, secretaccesskey

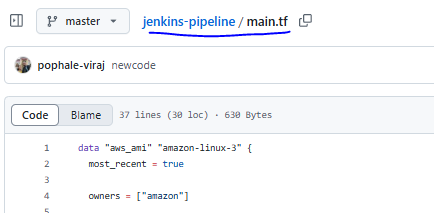
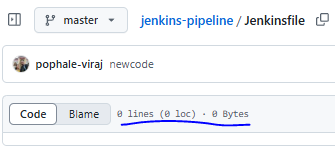
With this, our process has completed, we need to understand that with terraform pipeline, terraform code is built via the pipeline. Unlike Jenkins CI pipeline where java language is in use. Here need to test the code in local and push to github

Goto github >> “Jenkins-pipeline” repo >> take the steps mentioned to push the code as repo in place.

Goto VSCode >>

* git remote add origin <https://github.com/pophale-viraj/jenkins-pipeline.git>
* git branch -M main
* [git branch -m main master
* git fetch origin
* git branch -u origin/master master
* git remote set-head origin –a] ......changes from main to master

If noticed, the code in git includes all files b the new code pushed from VScode is available but does not exist for Jenkins related code.

Add vpc resource block to main.tf

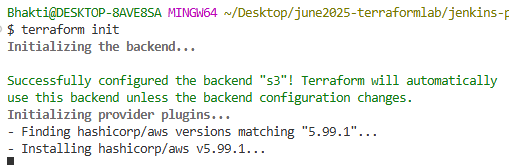


* git diff; git status; ......modified file
* terraform fmt
* terraform init ......bucket unavailable for tfstate upload

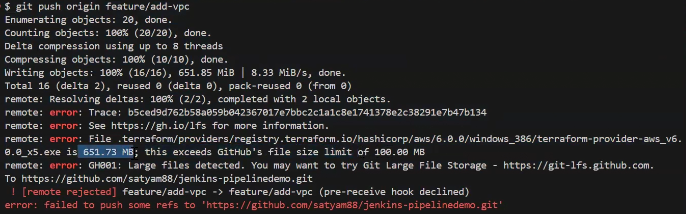
Goto S3 >> Create bucket=”terraform-dev-9325309959” ......empty bucket

Goto VSCode >> file backend.tf >> Add terraform {backend {}} block

* git add –all
* git commit –m “newcodeforbuck”
* git push origin master
* terraform fmt
* terraform init

 ......at this moment we ensured all our files are ready form terraform consumption from git and can be further coded as Jenkins-oriented. We failed earlier as bucket was missing. Now, we can proceed with terraform coding

* git checkout –b feature/add-vpc



Goto <https://github.com/github/gitignore> >> take terraform.gitignore file and add to your .gitignore file

* git add –all
* git commit –m “gitignore”

Still errors are visible for size of terraform file-size. In this case, always try troubleshooting like:

1. Goto Github >> new repository= ”terraform-deployment-aws” >> copy code-url >> goto desktop >> clone repository to desktop >> Add this folder to VSCode as a folder workspace >> Add to master in github using pull request >> add webhook >>

Start from 1.50 with new workspace