**Code**

**1.Yaml\_Code :-**

**import yaml**

**config = {**

**"name": "Deploy Azure Infrastructure with Terraform",**

**"on": {**

**"push": {**

**"branches": ["main"]**

**},**

**"workflow\_dispatch": None # Represents an empty key in YAML**

**},**

**# Commented env variables (you can uncomment and use them dynamically)**

**# "env": {**

**# "ARM\_CLIENT\_ID": "${{ fromJson(secrets.AZURE\_CREDENTIALS).clientId }}",**

**# "ARM\_CLIENT\_SECRET": "${{ fromJson(secrets.AZURE\_CREDENTIALS).clientSecret }}",**

**# "ARM\_SUBSCRIPTION\_ID": "${{ fromJson(secrets.AZURE\_CREDENTIALS).subscriptionId }}",**

**# "ARM\_TENANT\_ID": "${{ fromJson(secrets.AZURE\_CREDENTIALS).tenantId }}"**

**# },**

**"jobs": {**

**"terraform-deploy": {**

**"name": "Terraform Plan & Apply",**

**"runs-on": "ubuntu-latest",**

**"defaults": {**

**"run": {**

**"working-directory": "terraform"**

**}**

**},**

**"steps": [**

**{**

**"name": "📥 Checkout Code",**

**"uses": "actions/checkout@v4"**

**},**

**{**

**"name": "⚙️ Setup Terraform",**

**"uses": "hashicorp/setup-terraform@v3",**

**"with": {**

**"terraform\_version": "1.6.6"**

**}**

**}**

**]**

**}**

**}**

**}**

**# Convert back to YAML if needed**

**yaml\_string = yaml.dump(config, default\_flow\_style=False)**

**print(yaml\_string)**

**2.Terraform\_Code:-**

!pip install pyyaml python-hcl2

import hcl2

import os

def load\_terraform\_file(file\_path):

"""

Load and parse a Terraform (.tf) file from your local PC.

Returns the content as a Python dictionary.

"""

# Check if the provided file path exists

if not os.path.isfile(file\_path):

raise FileNotFoundError(f"❌ File not found: {file\_path}")

# Open and read the file if it exists

with open(file\_path, 'r') as file:

parsed\_data = hcl2.load(file)

return parsed\_data

# Example usage

file\_path1 = "C:\\swathi\\Terraform\\main.tf" # Ensure the path is correct

try:

terraform\_data = load\_terraform\_file(file\_path1)

print("Terraform file loaded successfully:")

print(terraform\_data)

except FileNotFoundError as e:

print(e)

file\_path2 = "C:\\swathi\\Terraform\\output.tf" # Ensure the path is correct

try:

terraform\_data = load\_terraform\_file(file\_path2)

print("Terraform file loaded successfully:")

print(terraform\_data)

except FileNotFoundError as e:

print(e)

file\_path3 = "C:\\swathi\\Terraform\\variable.tf" # Ensure the path is correct

try:

terraform\_data = load\_terraform\_file(file\_path3)

print("Terraform file loaded successfully:")

print(terraform\_data)

except FileNotFoundError as e:

print(e)