name: $(BuildDefinitionName)\_$(date:yyyyMMdd)$(rev:.r)

trigger: none

pr: none

pool:

  vmImage: Ubuntu-latest

variables:

  TF\_VAR\_RG\_NAME: ${{ parameters.RGNAME }}

  TF\_VAR\_LOCATION: ${{ parameters.LOCATION }}

  TF\_VAR\_ENVIRONMENT: ${{ parameters.ENVIRONMENT }}

  TF\_VAR\_ADB\_NAME: ${{ parameters.ADB\_NAME }}

  TF\_VAR\_ASA\_NAME: ${{ parameters.ASA\_NAME }}

  TF\_VAR\_ADF\_NAME: ${{ parameters.ADF\_NAME }}

  TF\_VAR\_AKV\_NAME: ${{ parameters.AKV\_NAME }}

  TF\_VAR\_CONTACT: ${{ parameters.CONTACT }}

parameters:

- name: ENVIRONMENT

  displayName: Subscription Environment

  type: string

  values:

  - dev

  - tst

  - prd

  - All

- name: RGNAME

  displayName: Please provide the Resource Group Name

  type: object

  default: <Resource Group Name>

- name: LOCATION

  displayName: Please provide the Location for the resource deployment

  type: string

  default: < West Europe (or)  North Europe>

- name: ADB\_NAME

  displayName: Please specify the data bricks resource name

  type: string

- name: ASA\_NAME

  displayName: Please specify the name of the App Storage account name

  type: string

- name: ADF\_NAME

  displayName: Please specify the name of the Azure data factory Service name

  type: string

- name: AKV\_NAME

  displayName: Please specify the name of the Azure Key vault name

  type: string

- name: CONTACT

  displayName: Contact Person for any issues with the azure platform or devops support

  type: string

stages :

  - stage: terraform\_plan\_publish

    jobs:

    - job: terraform\_plan\_publish

      steps:

        - task: ms-devlabs.custom-terraform-tasks.custom-terraform-installer-task.TerraformInstaller@0

          displayName: Installing Terraform

          inputs:

            terraformVersion: 'latest'

        - task: AzurePowerShell@5

          displayName: 'Terraform Initialize - ${{ parameters.ENVIRONMENT }}'

          inputs:

            azureSubscription: 'Azure-Deployment-${{ parameters.ENVIRONMENT }} '

            ScriptType: 'InlineScript'

            Inline: 'terraform init -backend=false'

            azurePowerShellVersion: 'LatestVersion'

            workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

        - task: TerraformTaskV4@4

          displayName: 'Terraform Validate - ${{ parameters.ENVIRONMENT }}'

          inputs:

            workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

            provider: 'azurerm'

            command: 'validate'

        - task: CmdLine@2

          displayName: 'Setting Workspace - ${{ parameters.ENVIRONMENT }}'

          inputs:

            workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

            script: terraform workspace select ${{ parameters.ENVIRONMENT }} || terraform workspace new ${{ parameters.ENVIRONMENT }}

        - task: TerraformTaskV4@4

          displayName: 'Terraform Plan - ${{ parameters.ENVIRONMENT }}'

          inputs:

            workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

            provider: 'azurerm'

            command: 'plan'

            CommandOptions: "--out=${{ parameters.ENVIRONMENT }}-tfplan"

            environmentServiceNameAzureRM: 'Azure-Deployment-${{ parameters.ENVIRONMENT }}'

            PublishPlanResults: '${{ parameters.ENVIRONMENT }}-tfplan'

  - stage: terraform\_plan\_build

    jobs:

    - job: terraform\_plan\_build

      steps:

        - task: ms-devlabs.custom-terraform-tasks.custom-terraform-installer-task.TerraformInstaller@0

          displayName: Installing Terraform

          inputs:

            terraformVersion: 'latest'

        - task: AzurePowerShell@5

          displayName: 'Terraform Initialize - ${{ parameters.ENVIRONMENT }}'

          inputs:

            azureSubscription: 'Azure-Deployment-${{ parameters.ENVIRONMENT }} '

            ScriptType: 'InlineScript'

            Inline: 'terraform init -backend=false'

            azurePowerShellVersion: 'LatestVersion'

            workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

        - task: TerraformTaskV4@4

          displayName: 'Terraform Validate - ${{ parameters.ENVIRONMENT }}'

          inputs:

            workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

            provider: 'azurerm'

            command: 'validate'

        - task: CmdLine@2

          displayName: 'Setting Workspace - ${{ parameters.ENVIRONMENT }}'

          inputs:

            workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

            script: terraform workspace select ${{ parameters.ENVIRONMENT }} || terraform workspace new ${{ parameters.ENVIRONMENT }}

        - task: TerraformTaskV4@4

          displayName: 'Terraform Plan - ${{ parameters.ENVIRONMENT }}'

          inputs:

            workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

            provider: 'azurerm'

            command: 'plan'

            CommandOptions: "--out=${{ parameters.ENVIRONMENT }}-tfplan"

            environmentServiceNameAzureRM: 'Azure-Deployment-${{ parameters.ENVIRONMENT }}'

        - task: ArchiveFiles@2

          inputs:

            rootFolderOrFile: '$(Build.SourcesDirectory)'

            includeRootFolder: false

            archiveType: 'tar'

            tarCompression: 'gz'

            archiveFile: '$(Build.ArtifactStagingDirectory)/$(Build.BuildId).tgz'

            replaceExistingArchive: true

            displayName: 'Create Plan Artifact'

        - task: PublishBuildArtifacts@1

          displayName: 'Publish Build Artifacts'

          inputs:

            PathtoPublish: '$(Build.ArtifactStagingDirectory)'

            ArtifactName: '$(Build.BuildId)-tfplan'

            publishLocation: 'Container'

            displayName: 'Publish Plan Artifact'

        - task: PowerShell@2

          displayName: 'Degug parameters'

          inputs:

            targetType: Inline

            script: |

              Write-Host "$(Build.ArtifactStagingDirectory)"

              Write-Host "$(System.DefaultWorkingDirectory)"

              Write-Host "$(System.ArtifactsDirectory)"

              Write-Host "$(Pipeline.Workspace)"

              Write-Host "$(System.ArtifactsDirectory)"

        - task: DeleteFiles@1

          displayName: 'Remove unneeded files'

          inputs:

            contents: |

              .terraform

              tfplan

  - stage: terraform\_plan\_apply

    dependsOn: [terraform\_plan\_build]

    #trigger: manual

    condition: succeeded('terraform\_plan\_build')

    jobs:

    - deployment: terraform\_apply #changing job to deployment

      timeoutInMinutes: 0

      environment:

        name: aztf-appservices-${{ parameters.ENVIRONMENT }}

      strategy:

        runOnce:

          deploy:

            steps:

            - checkout: none

            - task: DownloadPipelineArtifact@2     #Replaced DownloadBuildArtifacts@0

              inputs:

                artifactName: '$(Build.BuildId)-tfplan'

                displayName: 'Download Plan Artifact'

                path: $(System.DefaultWorkingDirectory)/

            - task: ExtractFiles@1

              inputs:

                archiveFilePatterns: '$(System.DefaultWorkingDirectory)/$(Build.BuildId).tgz' #'$(System.ArtifactsDirectory)/$(Build.BuildId)-tfplan/$(Build.BuildId).tgz'

                destinationFolder: '$(System.DefaultWorkingDirectory)/' #'$(System.DefaultWorkingDirectory)/'

                cleanDestinationFolder: false

                overwriteExistingFiles: false

                displayName: 'Extract Terraform Plan Artifact'

            - task: ms-devlabs.custom-terraform-tasks.custom-terraform-installer-task.TerraformInstaller@0

              displayName: Installing Terraform

              inputs:

                terraformVersion: 'latest'

            - task: AzurePowerShell@5

              displayName: 'Terraform Initialize - ${{ parameters.ENVIRONMENT }}'

              inputs:

                azureSubscription: 'Azure-Deployment-${{ parameters.ENVIRONMENT }} '

                ScriptType: 'InlineScript'

                Inline: 'terraform init -backend=false'

                azurePowerShellVersion: 'LatestVersion'

                workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

            - task: TerraformTaskV4@4

              displayName: 'Terraform Plan - ${{ parameters.ENVIRONMENT }}'

              inputs:

                workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

                provider: 'azurerm'

                command: 'plan'

                CommandOptions: "--out=${{ parameters.ENVIRONMENT }}-tfplan"

                environmentServiceNameAzureRM: 'Azure-Deployment-${{ parameters.ENVIRONMENT }}'

            - task: TerraformTaskV4@4

              displayName: 'Terraform Apply - ${{ parameters.ENVIRONMENT }}'

              inputs:

               workingDirectory: '$(System.DefaultWorkingDirectory)/App\_Deployment/vlk-adb/${{ parameters.ENVIRONMENT }}'

               provider: 'azurerm'

               command: 'apply'

               commandOptions: '-auto-approve'

               environmentServiceNameAzureRM: 'Azure-Deployment-${{ parameters.ENVIRONMENT }}' #App\_Deployment\web-app\ifte