Dotnet-without-stage

trigger:

-

pool:

  vmImage: 'windows-latest'

steps:

- task: UseDotNet@2

  displayName: 'Use .NET Core sdk'

  inputs:

    packageType: 'sdk'

    version: '3.x.x'

- script: dotnet restore

  displayName: 'Restore NuGet Packages'

- script: dotnet build --configuration Release

  displayName: 'Build Solution'

- task: DotNetCoreCLI@2

  displayName: 'Publish Artifact'

  inputs:

    command: 'publish'

    publishWebProjects: true

    arguments: '--configuration Release --output $(Build.ArtifactStagingDirectory)'

- task: AzureRmWebAppDeployment@4

  displayName: 'Azure App Service Deploy'

  inputs:

    ConnectionType: 'AzureRM'

    azureSubscription: 'Free Trial(2f637d63-48d6-4ad0-9ff8-95f3614e6914)'

    appType: 'webApp'

    WebAppName: 'test'

    packageForLinux: '$(Build.ArtifactStagingDirectory)/\*\*/\*.zip'

dotnet-with-stage

trigger:

-

stages:

- stage: Build

  displayName: 'Build Stage'

  jobs:

  - job: Build

    displayName: 'Build Job'

    pool:

      vmImage: 'windows-latest'

    steps:

    - task: UseDotNet@2

      displayName: 'Use .NET Core sdk'

      inputs:

        packageType: 'sdk'

        version: '3.x.x'

    - script: dotnet restore

      displayName: 'Restore NuGet Packages'

    - script: dotnet build --configuration Release

      displayName: 'Build Solution'

- stage: Publish

  displayName: 'Publish Stage'

  dependsOn: Build

  jobs:

  - job: Publish

    displayName: 'Publish Job'

    pool:

      vmImage: 'windows-latest'

    steps:

    - task: DotNetCoreCLI@2

      displayName: 'Publish Artifact'

      inputs:

        command: 'publish'

        publishWebProjects: true

        arguments: '--configuration Release --output $(Build.ArtifactStagingDirectory)'

- stage: Deploy

  displayName: 'Deploy Stage'

  dependsOn: Publish

  jobs:

  - job: Deploy

    displayName: 'Deploy Job'

    pool:

      vmImage: 'windows-latest'

    steps:

    - task: AzureRmWebAppDeployment@4

      displayName: 'Azure App Service Deploy'

      inputs:

        ConnectionType: 'AzureRM'

        azureSubscription: 'Free Trial(2f637d63-48d6-4ad0-9ff8-95f3614e6914)'

        appType: 'webApp'

        WebAppName: 'test'

        packageForLinux: '$(Build.ArtifactStagingDirectory)/\*\*/\*.zip'

aks-python-without-stage

trigger:

-

pool:

  vmImage: ubuntu-latest

steps:

- task: Docker@2

  inputs:

    containerRegistry: 'docker'

    repository: 'dockerimage'

    command: 'buildAndPush'

    Dockerfile: 'webapp/dockerfile'

    tags: 'latest'

- task: KubernetesManifest@1

  inputs:

    action: 'createSecret'

    connectionType: 'kubernetesServiceConnection'

    kubernetesServiceConnection: 'aks'

    namespace: 'default'

    secretType: 'dockerRegistry'

    secretName: 'acr-secret'

    dockerRegistryEndpoint: 'docker'

- task: KubernetesManifest@1

  inputs:

    action: 'deploy'

    connectionType: 'kubernetesServiceConnection'

    kubernetesServiceConnection: 'aks'

    namespace: 'default'

    manifests: |

       webapp/deployment.yml

       webapp/service.yml

    containers: 'avnishdocker.azurecr.io/dockerimage:latest'

    imagePullSecrets: 'acr-secret'

Assignment:

Convert this aks pipeline stage wise.