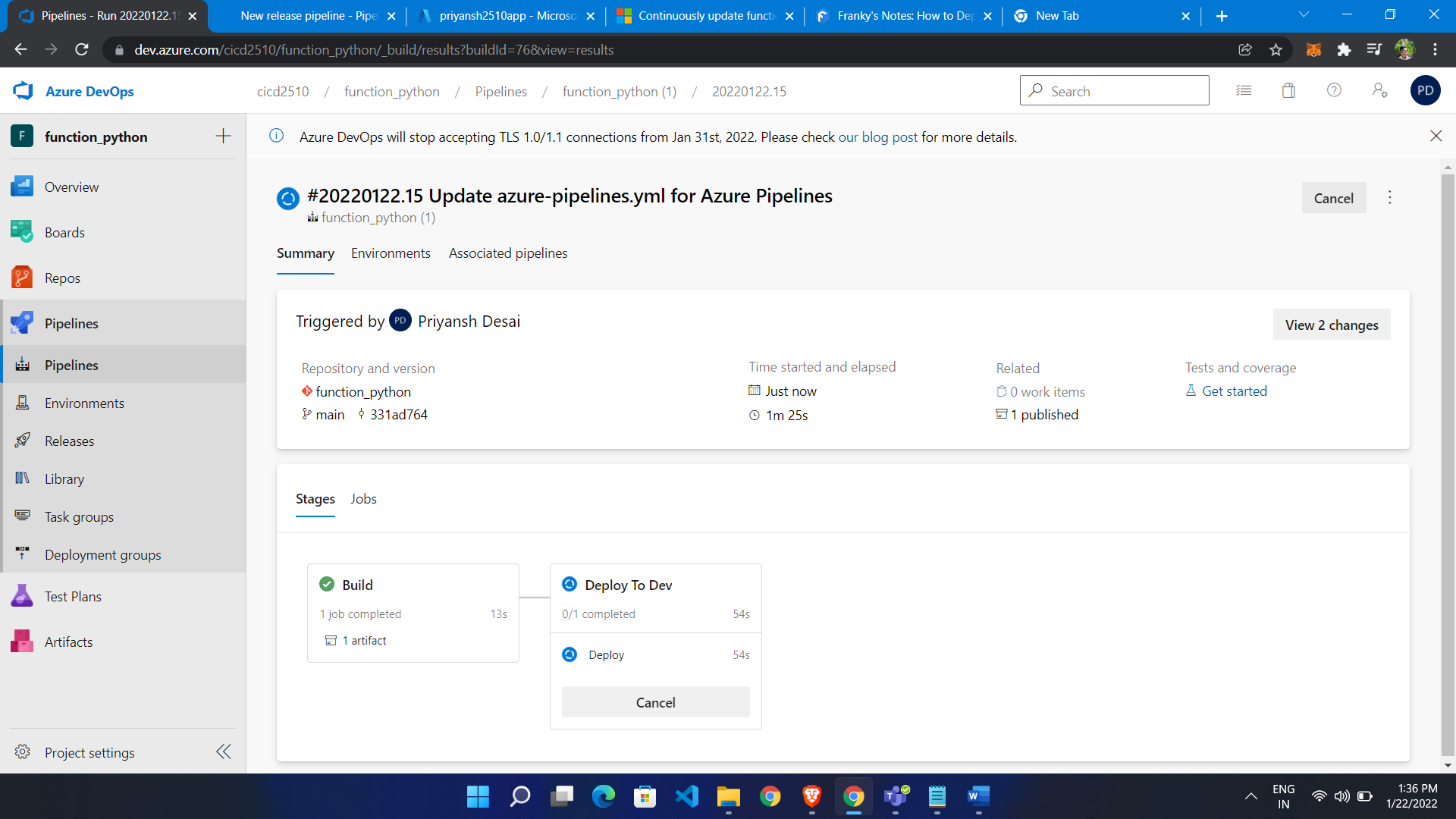
**YAML\_FUNCTION**

YAML:

|  |
| --- |
| # Starter pipeline  # Start with a minimal pipeline that you can customize to build and deploy your code.  # Add steps that build, run tests, deploy, and more:  # https://aka.ms/yaml  trigger:  - main  stages:    - stage: Build      jobs:        - job: build          pool:           vmImage: ubuntu-latest          steps:          - task: Bash@3            inputs:              targetType: 'inline'              script: |                if [ -f extensions.csproj ]                then                    dotnet build extensions.csproj --output ./bin                fi          - task: UsePythonVersion@0            inputs:              versionSpec: '3.6'              addToPath: true              architecture: 'x64'          - task: Bash@3            inputs:              targetType: 'inline'              script: |                python3.6 -m venv worker\_venv                source worker\_venv/bin/activate                pip3.6 install setuptools                pip3.6 install -r requirements.txt          - task: CopyFiles@2            inputs:              SourceFolder: 'ArmTemplate'              Contents: '\*\*'              TargetFolder: '$(build.artifactstagingdirectory)/Drop'          - task: ArchiveFiles@2            inputs:              rootFolderOrFile: '$(Build.BinariesDirectory)'              includeRootFolder: true              archiveType: 'zip'              archiveFile: '$(Build.ArtifactStagingDirectory)/$(Build.BuildId).zip'              replaceExistingArchive: true          - task: PublishBuildArtifacts@1            inputs:              PathtoPublish: '$(Build.ArtifactStagingDirectory)'              ArtifactName: 'drop'              publishLocation: 'Container'      - stage: DeployToDev      displayName: 'Deploy To Dev'      variables:           - group: 'YAML-Releases-Dev'      jobs:        - deployment:          pool:           vmImage: ubuntu-latest          environment: 'DEV'          strategy:           runOnce:             deploy:               steps:                 - task: AzureResourceManagerTemplateDeployment@3                   inputs:                     deploymentScope: 'Resource Group'                     azureResourceManagerConnection: 'my connection'                     subscriptionId: 'af173fdc-9c6c-4fa7-bab2-69d025b4142a'                     action: 'Create Or Update Resource Group'                     resourceGroupName: 'Demo'                     location: 'East US'                     templateLocation: 'Linked artifact'                     csmFile: '$(Pipeline.Workspace)/drop/Drop/function.json'                     overrideParameters: '-appName $(appName)'                     deploymentMode: 'Incremental'                 - task: AzureFunctionApp@1                   inputs:                     azureSubscription: 'my connection'                     appType: 'functionApp'                     appName: 'priyansh2510app'                     package: '$(Pipeline.Workspace)/\*\*/\*.zip'                     deploymentMethod: 'auto' |



Build Successfully

