

EXP NO: 5	Create Maven Build pipeline in Azure
DATE:	

AIM: -

To Create Maven Build pipeline in Azure

ALGORITHM / PROCEDURE: -

1. Create a New Azure Pipeline: In Azure DevOps, go to the Pipelines section and click New Pipeline.
2. Choose Repository: Select the repository where your Maven project is stored, such as GitHub or Azure Repos.
3. Select Pipeline Template: Choose Maven from the available templates or select Starter Pipeline if you want to manually configure it.
4. Configure YAML File: If you selected Starter Pipeline, configure the pipeline by editing the YAML file to define the build process.
5. Add Maven Build Task: In the YAML file, use the maven task to specify the goals like clean install or clean package to build the project.
6. Set Up Build Triggers: Configure triggers to run the pipeline on code changes, such as when code is pushed to a specific branch.
7. Run and Monitor: Save the pipeline, then run it manually or based on triggers, and monitor the build process through Azure DevOps.

CODING: -

trigger:

- main # Change to your branch name, e.g., 'master' or 'develop'

pool:

vmImage: 'ubuntu-latest' # Or any other VM image like 'windows-latest'

steps:

- task: MavenAuthenticate@0 # Authenticate with a repository if necessary

inputs:

mavenFeed: 'your-feed-name'

- task: Maven@3 # Maven task to build the project

inputs:

mavenPomFile: 'pom.xml' # Path to the Maven pom.xml file

options: '-X' # Optional: extra Maven options, e.g., verbose mode

goals: 'clean install' # Or 'clean package' depending on the goal

javaHomeOption: 'JDKVersion' # Choose the JDK version (e.g., '1.8', '11')

jdkVersionOption: '1.8'

mavenVersionOption: 'Default'

- task: PublishBuildArtifacts@1 # Optional: Publish build artifacts

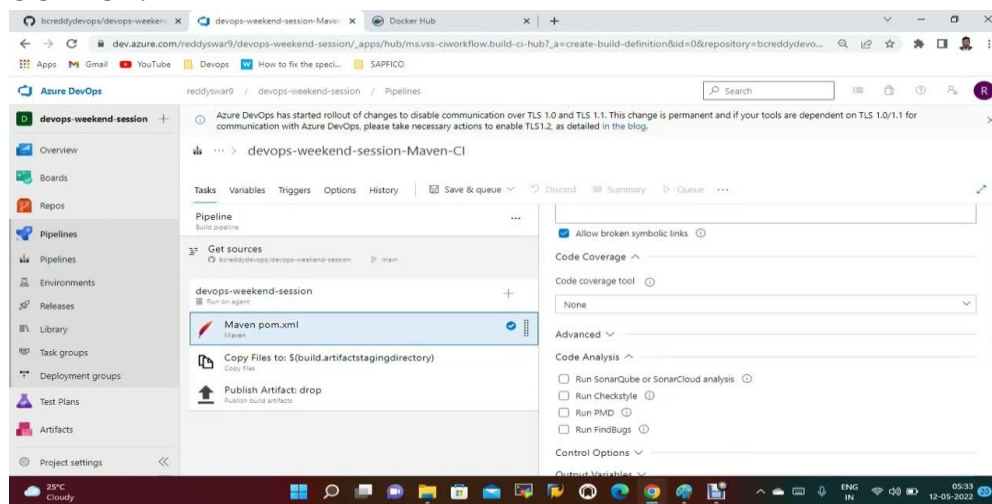
inputs:

PathtoPublish: '\$(Build.ArtifactStagingDirectory)'

ArtifactName: 'drop' # Name of the artifact

publishLocation: 'Container'

OUTPUT: -



RESULT: -

Thus we successfully created Maven Build pipeline in Azure.