**Explain steps for modify Azure-Pipelines.yaml file**

**Introduction to Azure Pipelines**

* Azure Pipelines is a service in **Azure DevOps** that allows you to **automate build, test, and deployment** of your code using **CI/CD pipelines**.
* The pipeline instructions are written in a YAML file called azure-pipelines.yml.

**Why Modify the YAML File?**

You modify the YAML file when you need to:

* Add new build or deployment steps
* Change triggers (like branch name)
* Add tasks for testing, building, or deploying
* Include conditions, variables, or environments

**Steps to Modify azure-pipelines.yml File**

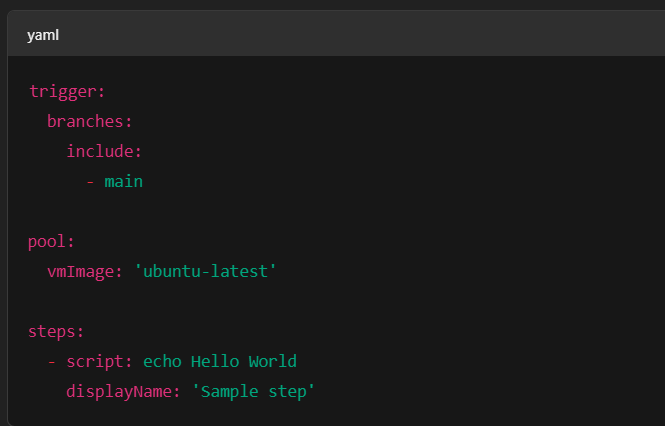
**1. Locate the YAML File**

* Go to your **Azure DevOps project**.
* Open the repository (code section).
* Look for azure-pipelines.yml at the root of the project.

**2. Open the YAML File**

* Click on the file to edit it.
* You can also clone the repo and open it in **VS Code** or any text editor.
* YAML files use indentation (spaces) to organize steps, so edit carefully.

**3. Understand the Current Structure**



* trigger: Defines when the pipeline runs
* pool: Defines the VM or environment to run
* steps: List of tasks to perform

**4. Make Your Modifications**

**a. Change the Trigger Branch**

trigger:

branches:

include:

- develop

Runs pipeline only when develop branch is updated.

**b. Add a New Task (e.g., Run Tests)**

- script: npm install

displayName: 'Install packages'

Adds a new step to run your project’s tests.

**c. Use a Different Agent/Pool**

pool:

vmImage: 'windows-latest'

Runs the pipeline on a Windows virtual machine.

**d. Add a Deployment Job**

- task: AzureWebApp@1

inputs:

appName: 'myapp'

package: '$(System.DefaultWorkingDirectory)/\*\*/\*.zip'

Deploys your app to Azure Web App.

**e. Add Variables**

variables:

nodeVersion: '18.x'

You can reuse this variable in steps using $(buildConfiguration).

**f. Use Conditions**

- script: echo "Deploy step"

condition: succeeded()

Only runs if the previous steps succeed.

**5. Validate YAML Format**

* YAML is sensitive to spaces.
* Use **2 spaces** (not tabs) for indentation.
* You can use online tools or Azure DevOps built-in validator to check errors.

**6. Save and Commit the Changes**

* Save the changes and commit them to your branch.
* Azure Pipeline will automatically run (if trigger is set properly).

**7. Monitor the Pipeline**

* Go to **Pipelines > Runs** in Azure DevOps.
* Check the output logs for each step.
* Fix errors if any step fails.

**Conclusion**

Modifying azure-pipelines.yml involves updating tasks, triggers, variables, or environments based on your project’s needs. Understanding YAML structure and Azure DevOps syntax is key to successfully customizing the CI/CD pipeline