CI/CD Pipeline

##
Full CI/CD Pipeline
##

-----*** Using Git ***-----

$ sudo apt update
$ sudo apt install git
$ git --version

>> Initialise local git repository
$ mkdir gitdemo
$ cd gitdemo
$ git init
$ nano file1
this is first line of code

$ git status

>> Put code to staging Area
$ git add file1

$ git status

>> Initialise Commit
$ git config user.name "Aasem Quazi"
$ git config user.email "aasem@databinaries.com"
$ git commit

>> Make changes
$ nano file1
this is second line of code

$ git status

$ git add file1
$ git status
$ git commit -m "added second line of code"

>> Check log

$ git log

----*** Connecting to remote repo ***----

$ ssh-keygen -t rsa -b 4096
$ cd .ssh
$ cat id_rsa.pub
(Copy content)
>> Add key in github account
$ cd
$ mkdir gitdemo2
$ cd gitdemo2
$ git clone git@github.com:aasemquazi/test1.git
$ cd test1
$ git status

$ nano pqr
first line of code

$ git status
$ git add pqr
$ git status
$ git commit -m "added"
$ git status
$ git push
>> Verify on Github.com

##

----*** Installing Jenkins ***----

$ sudo apt install openjdk-17-jdk -y

$ curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \
  /usr/share/keyrings/jenkins-keyring.asc > /dev/null

$ echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null

$ sudo apt-get update

$ sudo apt-get install jenkins -y

$ sudo service jenkins status

>> Connect to jenkins on browser on port 8080

----*** Build a Jenkins Job ***----

>> Click New
>> Freestyle Project
>> Build Steps - Execute Shell
        id
        pwd
        ping -c 1 google.com

>> Click New
>> Freestyle Project
>> Build Steps - Execute Shell
        echo "Job is runnning" >> myjob.txt
        echo "See text live"
        cat myjob.txt

Build from Git repo

>> Click New
>> Freestyle Project
>> Source Code Management - Git

Repository URL - https://github.com/databinaries001/ci-cd-demo.git
Save

----*** Triggering build with Git Hooks ***----

>> Generate GitHub API token
Your account - settings - developer tools - Personal access tokens - admin:repo:hook

>> Add API token in Jenkins
Manage Jenkins -> System -> Github -> Add github server -> Give a name -> Add credentials -> Kind : secret text -> Add
Click on Manage Hooks Checkbox
Test Connection

Create Jenkins Job

>> Click New
>> Freestyle Project
>> Source Code Management - Git
https://github.com/aasemquazi/ci-cd-demo-1.git
Build Triggers
GitHub hook trigger for GITScm polling - Check the checkbox
(Note : Add port 8080 in security group if required)

----*** Build a Jenkins Pipeline ***----

>>> New Item -> Pipeline -> Try sample - Hello world

>>> New Item -> Pipeline ->

```
pipeline {
   agent any
   stages {
      stage('Build') {
         steps {
            echo 'Build the job'
         }
      }
      stage('Test') {
         steps {
            echo 'Test the job'
         }
```

```
        }
        stage('Deploy') {
            steps {
                echo 'Deploy the job'
            }
        }
    }
}
```

----*** Build a Jenkins Test Pipeline ***----

Install Go plugin

>> Configure Go plugin
Manage Jenkins -> Go -> Tools -> Add Go -> Version Go 1.17

Build from the repo
>> Click New -> Freestyle Project -> Git -> Repository URL -> https://github.com/databinaries001/ci-cd-demo.git

Now start the test pipeline

>>> New Item -> Pipeline ->

```
pipeline {
  agent any
  tools {
     go 'gotest'
  }
  environment {
     GO111MODULE='on'
  }

  stages {
    stage('Test') {
      steps {
        git 'https://github.com/databinaries001/ci-cd-demo.git'
        sh 'go test ./...'
      }
    }

  }
}
```

>> Install golang on os

```
sudo apt  install golang-go -y
```

----*** Build a Full Pipeline ***----

>>> New Item -> Pipeline ->

```
pipeline {
 agent any
 tools {
    go 'gotest'
 }
 environment {
    GO111MODULE='on'
 }
 stages {
  stage('Test') {
   steps {
    git 'https://github.com/databinaries001/ci-cd-demo.git'
    sh 'go test ./...'
   }
  }
  stage('Build') {
     steps {
     git 'https://github.com/databinaries001/ci-cd-demo.git'
     sh 'go build .'
     }
  }
  stage('Run') {
     steps {
        sh 'cd /var/lib/jenkins/workspace/full-cicd-go && go-webapp-sample &'
     }
  }
 }
}
```

----*** Build a Full CI/CD Pipeline ***----

>>> New Item -> Pipeline ->

>> Build Triggers - GitHub hook trigger for GITScm polling (Check it)

>> Pipeline -> Pipeline script from SCM

SCM - Git
Repository URL - https://github.com/aasemquazi/ci-cd-demo-1.git

>> Credentials - Add - Username and password (Give uname and pass of Github)
Save

>>Now go under github and update jenkins file with below code and commit on github itself

```
pipeline {
 agent any
 tools {
    go 'gotest'
 }
 environment {
    GO111MODULE='on'
 }

 stages {
  stage('Test') {
   steps {
    git 'https://github.com/databinaries001/ci-cd-demo.git'
    sh 'go test ./...'
   }
  }
  stage('Build') {
     steps {
     git 'https://github.com/databinaries001/ci-cd-demo.git'
     sh 'go build .'
     }
  }
  stage('Run') {
     steps {
        sh 'cd /var/lib/jenkins/workspace/full-cicd-go && go-webapp-sample &'
     }
  }

 }
}
```

>>Now check in Jenkins