

NAME:ELAKKIYA D

ROLL NO:24MCR022

DEVOPS TRAINING

DAY 5 CONFIGURING PIPELINE

Step 1: Create github repository kubernetes and push the cd kubernetes files as frontend ,backend and k8s

```
Select student@CTS-6: ~/kubernetes
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
student@CTS-6:~/kubernetes$ git pull --rebase origin main
fatal: 'origin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
student@CTS-6:~/kubernetes$ ls
README.md  backend  commands-to-stop-instances  frontend  k8s  minikube-linux-amd64
student@CTS-6:~/kubernetes$ cd ..
student@CTS-6:~$ ls
docker-python-app  kubernetes  kubernet  webapp
student@CTS-6:~/kubernet/$ ls
README.md
student@CTS-6:~/kubernet/$ ls
README.md
student@CTS-6:~/kubernet/$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
student@CTS-6:~/kubernet/$ git pull
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 17 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (17/17), 3.06 KiB | 223.00 KiB/s, done.
From https://github.com/53nThil-03/kubernetes
  463ed7d..ea91e00  main    -> origin/main
Updating 463ed7d..ea91e00
Fast-forward
 README.md             | 3 ++
 backend/app.py        | 21 +++++
 backend/dockerfile    | 6 +++++
 backend/products.csv  | 5 +++++
 backend/requirements.txt | 3 +++
 frontend/dockerfile   | 2 ++
 frontend/index.html   | 25 +++++
 k8s/allow-all.yaml   | 9 +++++
 k8s/backend-deployment.yaml | 20 +++++
 k8s/configmap.yaml    | 6 +++++
 k8s/frontend-deployment.yaml | 20 +++++
 k8s/service.yaml      | 25 +++++
 12 files changed, 144 insertions(+), 1 deletion(-)
create mode 100644 backend/app.py
create mode 100644 backend/dockerfile
create mode 100644 backend/products.csv
create mode 100644 backend/requirements.txt
create mode 100644 frontend/dockerfile
create mode 100644 frontend/index.html
```

```

to https://github.com/3nth1x-03/kubernetes
963d1c1-623b0d2 623b0d2 --git --no
student@CTS-6:/kubernetes$ nano Jenkinsfile
student@CTS-6:/kubernetes$ nano Jenkinsfile
student@CTS-6:/kubernetes$ cat Jenkinsfile
pipeline {
    agent any

    environment {
        // Define Image Names
        BACKEND_IMAGE = "senthil73387/docker-backend:latest"
        FRONTEND_IMAGE = "senthil73387/docker-frontend:latest"
        // Define container names
        BACKEND_CONTAINER = "docker-running-backend"
        FRONTEND_CONTAINER = "docker-running-frontend"
    }

    stages {
        stage('Checkout Code') {
            steps {
                withCredentials([usernamePassword(credentialsId: 'github-senthil', usernameVariable: 'GIT_USER', passwordVariable: 'GIT_TOKEN')]) {
                    git url: 'https://$GIT_USER:$GIT_TOKEN@github.com:3nth1x-03/kubernetes.git', branch: 'main'
                }
            }
        }

        stage('Build Backend Docker Image') {
            steps {
                // Use the backend directory as the build context
                sh 'docker build -t $BACKEND_IMAGE -f backend/dockerfile backend'
            }
        }

        stage('Build Frontend Docker Image') {
            steps {
                // Use the frontend directory as the build context
                sh 'docker build -t $FRONTEND_IMAGE -f frontend/dockerfile frontend'
            }
        }

        stage('Login to Docker Registry') {
            steps {
                withCredentials([usernamePassword(credentialsId: 'dockerhub-senthil', usernameVariable: 'DOCKER_USER', passwordVariable: 'DOCKER_PASS')]) {
                    sh 'echo $DOCKER_PASS | docker login -u $DOCKER_USER --password-stdin'
                }
            }
        }

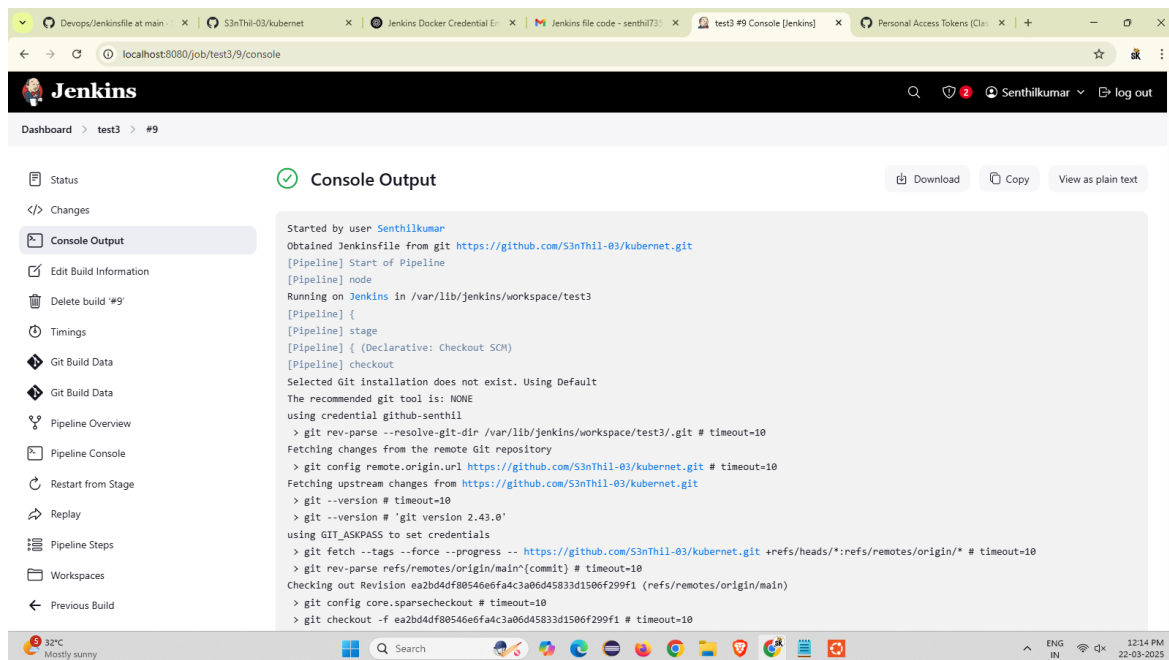
        stage('Push Docker Images') {
            steps {
                sh 'docker push $BACKEND_IMAGE'
                sh 'docker push $FRONTEND_IMAGE'
            }
        }

        stage('Stop & Remove Existing Containers') {
            steps {
                script {
                    sh '''
                    if [ $(docker ps --q --name=$BACKEND_CONTAINER) ]; then
                        docker stop $BACKEND_CONTAINER || true
                        docker rm $BACKEND_CONTAINER || true
                    fi
                    if [ $(docker ps --q --name=$FRONTEND_CONTAINER) ]; then
                        docker stop $FRONTEND_CONTAINER || true
                        docker rm $FRONTEND_CONTAINER || true
                    fi
                    '''
                }
            }
        }
    }
}

```

The screenshot shows the Jenkins web interface in a browser. The address bar indicates the URL is `localhost:8080/job/test3/9/`. The Jenkins logo and name are at the top left. The top navigation bar shows the user `Senthilkumar` and a `log out` button. The main content area displays the build status for `#9` (22 Mar 2025, 06:40:18). The build is in a `Success` state, indicated by a green checkmark. The left sidebar contains a list of navigation items: `Status` (selected), `Changes`, `Console Output`, `Edit Build Information`, `Timings`, `Git Build Data`, `Pipeline Overview`, `Pipeline Console`, `Thread Dump`, `Pause/resume`, `Replay`, `Pipeline Steps`, `Workspaces`, and `Previous Build`. The main content area shows the build details for `#9`, including the user `Senthilkumar`, the repository `https://github.com/S3nThil-03/kubernetes.git`, and the revision `ea2bd4df80546e6fa4c3a06d45833d1506f299f1`. The build is in progress, indicated by a yellow warning icon. The console output shows the command `git checkout -f` and the output `Already up to date.`

Step 4 : open console output and check build is complete or not.



Step 5 : go to dashboard > manage Jenkins > plugins and install the Kubernetes once it all download success will shown.

