

DAY 5

1)Install Maven

Command: sudo apt update

sudo apt install maven -y

To Check Version: mvn -version

2)Pipeline Script

```
pipeline {
```

```
    agent any
```

```
    environment {
```

```
        IMAGE_NAME = "reshma09/devops" // Replace with your Docker Hub username and  
        image name
```

```
        TAG = "latest"
```

```
        CONTAINER_NAME = "my-container"
```

```
        PORT = "3001"
```

```
    }
```

```
    stages {
```

```
        stage('Clone Repository') {
```

```
            steps {
```

```
                echo "Cloning GitHub repository..."
```

```
                git branch:'main', url: 'https://github.com/reshreshu/devops_.git' // Replace with  
                your repo URL
```

```
            }
```

```
        }
```

```
        stage('Build Docker Image') {
```

```
            steps {
```

```
    echo "Building Docker image..."

    sh 'chmod +x build.sh'

    sh './build.sh'

  }
}
```

```
stage('Login to Docker Hub') {
  steps {
    echo "Logging into Docker Hub..."

    withCredentials([usernamePassword(credentialsId: 'docker-hub-creds',
usernameVariable: 'DOCKER_USER', passwordVariable: 'DOCKER_PASS')]) {
      sh 'docker login -u $DOCKER_USER -p $DOCKER_PASS'
    }
  }
}
```

```
stage('Push Docker Image') {
  steps {
    echo "Pushing Docker image to Docker Hub..."

    sh "docker tag $IMAGE_NAME:$TAG $IMAGE_NAME:$TAG"

    sh "docker push $IMAGE_NAME:$TAG"

  }
}
```

```
stage('Deploy Docker Container') {
  steps {
    echo "Deploying Docker container..."

    sh 'chmod +x deploy.sh'

    sh './deploy.sh'

  }
}
```

```

    }
}

post {
    success {
        echo "Deployment Successful!"
    }
    failure {
        echo "Deployment Failed!"
    }
}
}

```

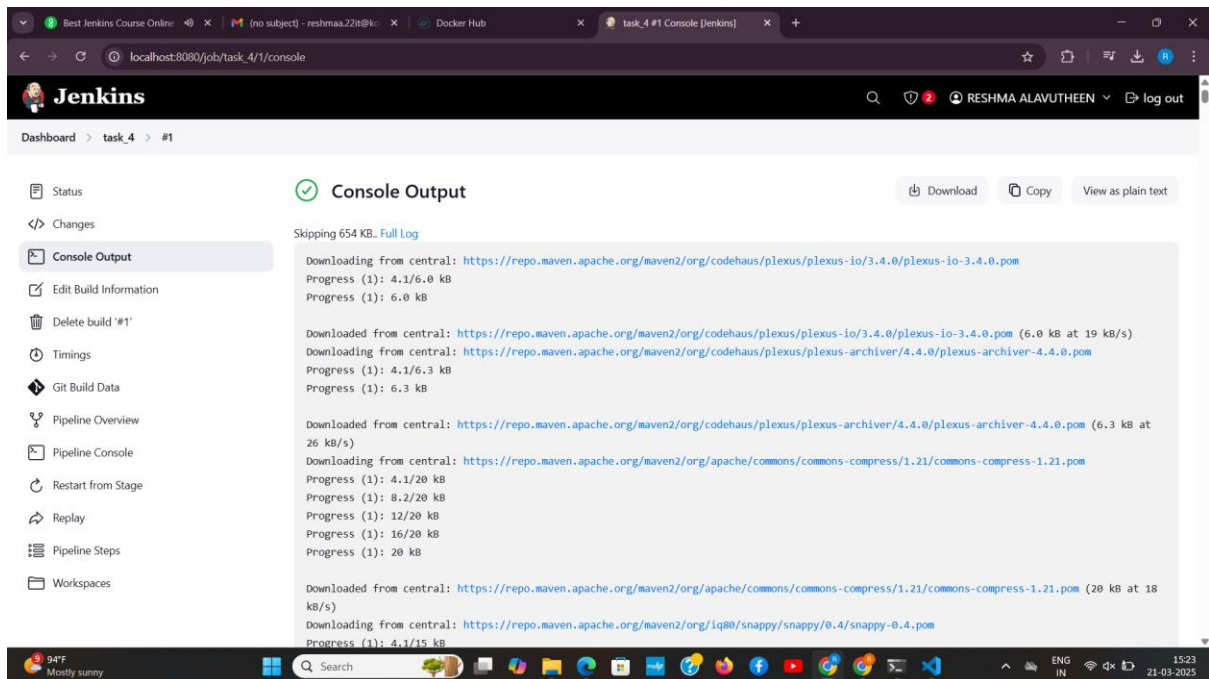
3)Build a new pipeline job

The screenshot displays the Jenkins web interface for a pipeline job named 'devops_git'. The interface includes a sidebar with navigation options like Status, Changes, Build Now, Configure, Delete Pipeline, Full Stage View, Stages, Rename, and Pipeline Syntax. The main area shows the 'Stage View' for the pipeline, which consists of six stages: Clone Repository, Build Docker Image, Login to Docker Hub, Push Docker Image, Deploy Docker Container, and Declarative: Post Actions. Each stage has a duration listed below it. A table summarizes the stage durations for the current build (#11) and the average stage times.

Stage	Clone Repository	Build Docker Image	Login to Docker Hub	Push Docker Image	Deploy Docker Container	Declarative: Post Actions
Average stage times (full run time: ~1min 2s)	2s	3s	8s	46s	1s	169ms
Build #11 (15:10)	2s	3s	8s	46s	1s	169ms

Below the stage view, there are 'Permalinks' for the last build (#11), last stable build (#11), last successful build (#11), and last completed build (#11), all from 10 minutes ago.

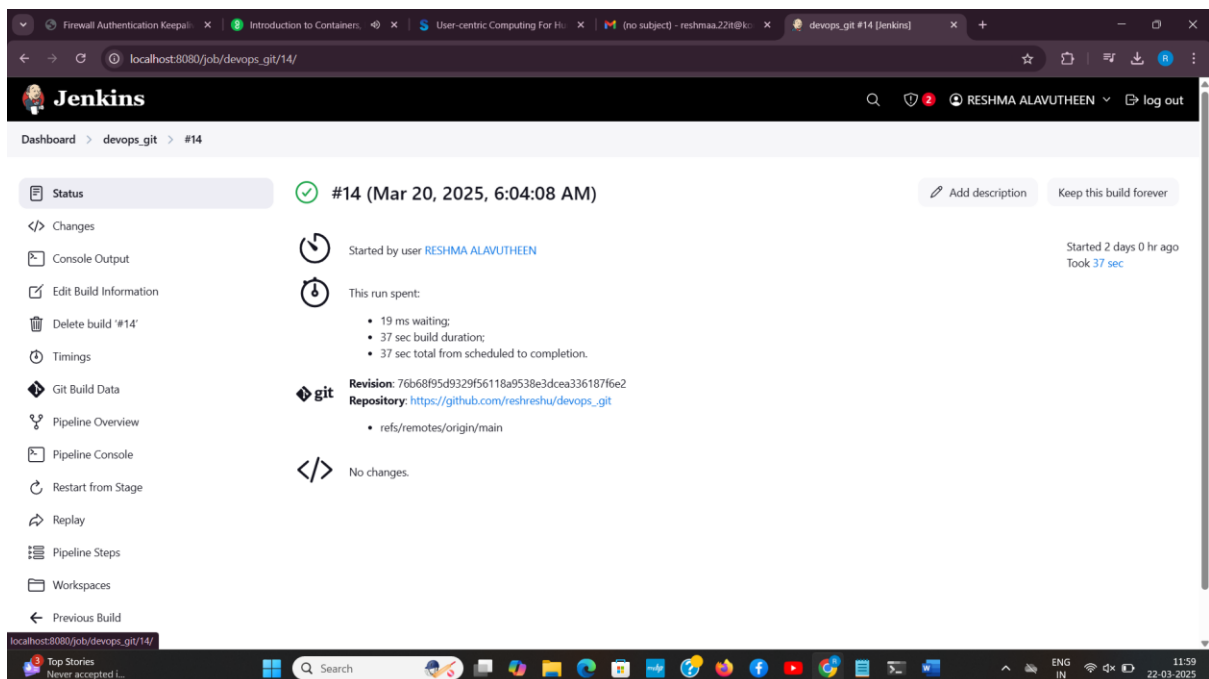
4) Console Output of the pipeline



The screenshot shows the Jenkins web interface for a build named 'task_4 #1'. The left sidebar contains a list of navigation options: Status, Changes, Console Output (selected), Edit Build Information, Delete build '#1', Timings, Git Build Data, Pipeline Overview, Pipeline Console, Restart from Stage, Replay, Pipeline Steps, and Workspaces. The main area displays the 'Console Output' for this build. At the top, there is a green checkmark icon and the text 'Console Output'. Below this, there are buttons for 'Download', 'Copy', and 'View as plain text'. The output text shows the following sequence of events:

- Skipping 654 KB. Full Log
- Downloading from central: <https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/3.4.0/plexus-io-3.4.0.pom>
Progress (1): 4.1/6.0 kB
Progress (1): 6.0 kB
- Downloaded from central: <https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/3.4.0/plexus-io-3.4.0.pom> (6.0 kB at 19 kB/s)
- Downloading from central: <https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/4.4.0/plexus-archiver-4.4.0.pom>
Progress (1): 4.1/6.3 kB
Progress (1): 6.3 kB
- Downloaded from central: <https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/4.4.0/plexus-archiver-4.4.0.pom> (6.3 kB at 26 kB/s)
- Downloading from central: <https://repo.maven.apache.org/maven2/org/apache/commons/commons-compress/1.21/commons-compress-1.21.pom>
Progress (1): 4.1/20 kB
Progress (1): 8.2/20 kB
Progress (1): 12/20 kB
Progress (1): 16/20 kB
Progress (1): 20 kB
- Downloaded from central: <https://repo.maven.apache.org/maven2/org/apache/commons/commons-compress/1.21/commons-compress-1.21.pom> (20 kB at 18 kB/s)
- Downloading from central: <https://repo.maven.apache.org/maven2/org/iq80/snappy/snappy/0.4/snappy-0.4.pom>
Progress (1): 4.1/15 kB

5) Status of the pipeline



The screenshot shows the Jenkins web interface for a build named 'devops_git #14'. The left sidebar contains a list of navigation options: Status (selected), Changes, Console Output, Edit Build Information, Delete build '#14', Timings, Git Build Data, Pipeline Overview, Pipeline Console, Restart from Stage, Replay, Pipeline Steps, and Workspaces. The main area displays the 'Status' for this build. At the top, there is a green checkmark icon and the text '#14 (Mar 20, 2025, 6:04:08 AM)'. Below this, there are buttons for 'Add description' and 'Keep this build forever'. The status information includes:

- Started by user [RESHMA ALAVUTHEEN](#)
- Started 2 days 0 hr ago
- Took 37 sec
- This run spent:
 - 19 ms waiting;
 - 37 sec build duration;
 - 37 sec total from scheduled to completion.
- Revision: [76b68f95d9329f56118a9538e3dcea336187f6e2](https://github.com/reshreshu/devops_git)
- Repository: https://github.com/reshreshu/devops_git
 - refs/remotes/origin/main
- No changes.

6) Pull docker image

```
root@HP: ~/my-docker-app
root@HP:~# mkdir my-docker-app
mkdir: cannot create directory 'my-docker-app': File exists
root@HP:~# cd my-docker-app
root@HP:~/my-docker-app# touch Dockerfile
root@HP:~/my-docker-app# nano Dockerfile
root@HP:~/my-docker-app# npm init -y
Wrote to /mnt/localhost/Ubuntu/root/my-docker-app/package.json:

{
  "name": "my-docker-app",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": ""
}

root@HP:~/my-docker-app# docker pull reshma09/devops:latest
latest: Pulling from reshma09/devops
Digest: sha256:6d662d46ee11378699cf2491db329de1f645d486fec9312c3ee13c7b58d12892
Status: Image is up to date for reshma09/devops:latest
docker.io/reshma09/devops:latest
root@HP:~/my-docker-app# docker build -t reshma09/devops:latest .
[+] Building 12.4s (10/10) FINISHED          docker:default
=> [internal] load build definition from Dockerfile          0.0s
=> => transferring dockerfile: 343B                          0.0s
=> [internal] load metadata for docker.io/library/node:18    7.7s
=> [internal] load .dockerignore                             0.0s
=> => transferring context: 2B                                 0.0s
=> [1/5] FROM docker.io/library/node:18@sha256:7f6bcd8e08a1f81bfb29 0.0s
=> [internal] load build context                             0.0s
=> => transferring context: 613B                               0.0s
=> CACHED [2/5] WORKDIR /app                                0.0s
=> [3/5] COPY package.json ./                               0.1s
=> [4/5] RUN npm install                                     4.3s
=> [5/5] COPY . .                                           0.0s
```

```
=> [5/5] COPY . .                                           0.0s
=> exporting to image                                       0.1s
=> => exporting layers                                         0.1s
=> => writing image sha256:55712ea2b13e7b5b0d1e87cba06009a0ea7992a6d 0.0s
=> => naming to docker.io/reshma09/devops:latest             0.0s
root@HP:~/my-docker-app# docker ps
CONTAINER ID   IMAGE          STATUS      PORTS      COMMAND
CREATED
NAMES
97886ff2fba4   cc8943d6d292   "catalina.sh run"   About an hour ago   Up About an hour   8888/tcp, 0.0.0.0:3002->80/tcp, [::]:3002->80/tcp
fc88758e6df5   gcr.io/k8s-minikube/kicbase:v0.0.46   "/usr/local/bin/entr-"   2 days ago         Up 17 minutes     127.0.0.1:32768->22/tcp, 127.0.0.1:3276
9->2376/tcp, 127.0.0.1:32770->5000/tcp, 127.0.0.1:32771->8443/tcp, 127.0.0.1:32772->32443/tcp
root@HP:~/my-docker-app# minikube start --force
minikube v1.35.0 on Ubuntu 24.04 (amd64)
! minikube skips various validations when --force is supplied; this may lead to unexpected behavior
* Using the docker driver based on existing profile
* The "docker" driver should not be used with root privileges. If you wish to continue as root, use --force.
* If you are running minikube within a VM, consider using --driver=none:
  https://minikube.sigs.k8s.io/docs/reference/drivers/none/
* Tip: To remove this root owned cluster, run: sudo minikube delete
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.46 ...
* Updating the running docker "minikube" container ...
! Failing to connect to https://registry.k8s.io/ from inside the minikube container
! To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking/proxy/
* Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
* Verifying Kubernetes components...
  * Using image gcr.io/k8s-minikube/storage-provisioner:v5
  * Using image docker.io/kubernetesui/dashboard:v2.7.0
  * Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
! Some dashboard features require the metrics-server addon. To enable all features please run:
  minikube addons enable metrics-server
* Enabled addons: storage-provisioner, default-storageclass, dashboard
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
root@HP:~/my-docker-app# sudo nano nginx-deployment2.yaml
root@HP:~/my-docker-app# kubectl apply -f nginx-deployment2.yaml
deployment.apps/mvn-app created
```

```
root@HP: ~/my-docker-app x + v
root@HP:~/my-docker-app# minikube start --force
minikube v1.35.0 on Ubuntu 24.04 (amd64)
! minikube skips various validations when --force is supplied; this may lead to unexpected behavior
* Using the docker driver based on existing profile
* The "docker" driver should not be used with root privileges. If you wish to continue as root, use --force.
* If you are running minikube within a VM, consider using --driver=none:
  https://minikube.sigs.k8s.io/docs/reference/drivers/none/
* Tip: To remove this root owned cluster, run: sudo minikube delete
Starting "minikube" primary control-plane node in "minikube" cluster
Pulling base image v0.0.46 ...
Updating the running docker "minikube" container ...
Failing to connect to https://registry.k8s.io/ from inside the minikube container
To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking/proxy/
Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
Verifying Kubernetes components...
  * Using image gcr.io/k8s-minikube/storage-provisioner:v5
  * Using image docker.io/kubernetesui/dashboard:v2.7.0
  * Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
Some dashboard features require the metrics-server addon. To enable all features please run:

    minikube addons enable metrics-server

* Enabled addons: storage-provisioner, default-storageclass, dashboard
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
root@HP:~/my-docker-app# sudo nano nginx-deployment2.yaml
root@HP:~/my-docker-app# kubectl apply -f nginx-deployment2.yaml
deployment.apps/mvn-app created
service/mvn-app-service created
root@HP:~/my-docker-app# kubectl get pods
NAME                                READY   STATUS              RESTARTS   AGE
mvn-app-59b7764985-2h5cj             0/1     ContainerCreating   0           24s
mvn-app-59b7764985-kfpv9             0/1     ContainerCreating   0           24s
my-app-7d8b8f5d5d-h5szn             1/1     Running             2 (4m53s ago)  26h
my-nginx-768f959b59-dsznh           1/1     Running             5 (4m53s ago)  2d6h
root@HP:~/my-docker-app#
```

94°F Mostly sunny 16:00 21-03-2025