

## ASSIGNMENT -2

Course: DevOps

Module: TOMCAT

Topic: Deploy in tomcat

Trainer Name: Mr. Madhukar

Name: Lakshmi Priyanka Chinta

Mail Id: chlakshmipriyanka9@gmail.com

Batch No: 115 (9 am)

Assignment: 2

Date of

Submission: 25-11-23

### TOPIC :

By the below URL Please deploy the application in Tomcat

<https://github.com/venna12/dockerjenkin.git>

### AWS:

Two instances are created as Jenkins and Tomcat. The Jenkins instance is used to create JENKINS and the Tomcat instance is used to create the Tomcat server.

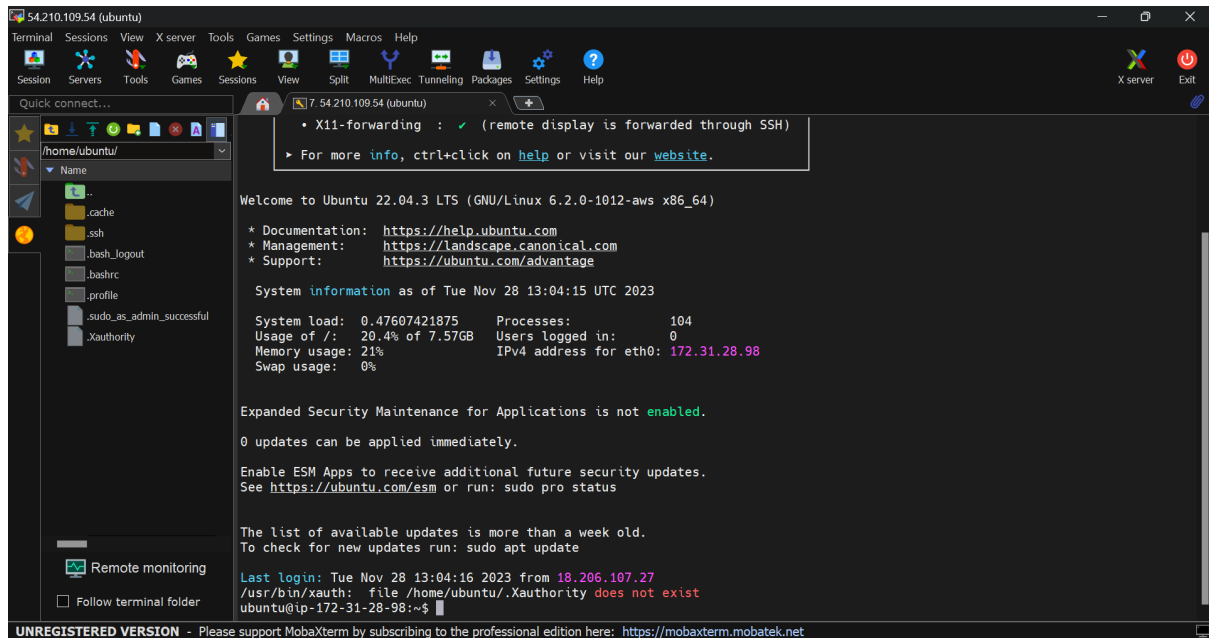
The screenshot displays the AWS Management Console interface. On the left, the navigation menu shows 'Instances' selected. The main panel shows a list of two EC2 instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
jenkins	i-0615efb7af229b73a	Running	t2.micro	Initializing	No alarms
tomcat	i-069e7cbac2d739242	Running	t2.micro	Initializing	No alarms

The 'jenkins' instance details are expanded, showing the following information:

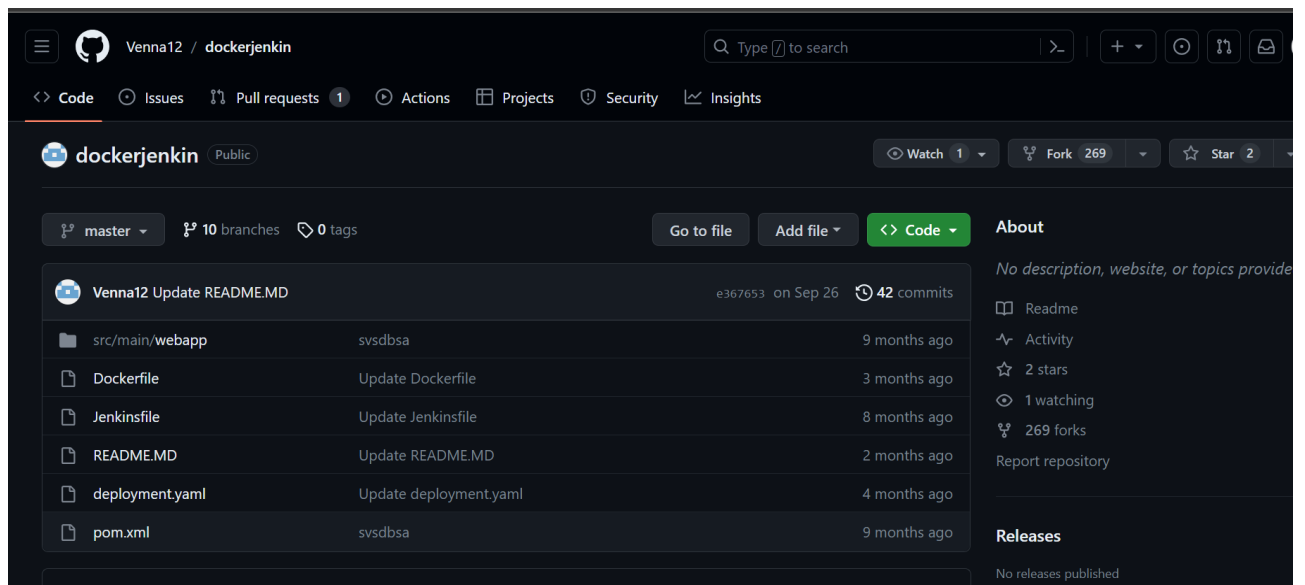
- Instance ID: i-0615efb7af229b73a (jenkins)
- Public IPv4 address: 54.226.196.31
- Private IPv4 addresses: 172.31.26.170
- Instance state: Running
- Public IPv4 DNS: 54-226-196-31.compute-1.amazonaws.com

## SERVER CONNECTION:



It shows the connections of instances in mobaxterm, which is used to install Jenkins, tomcat, and required packages, and more.

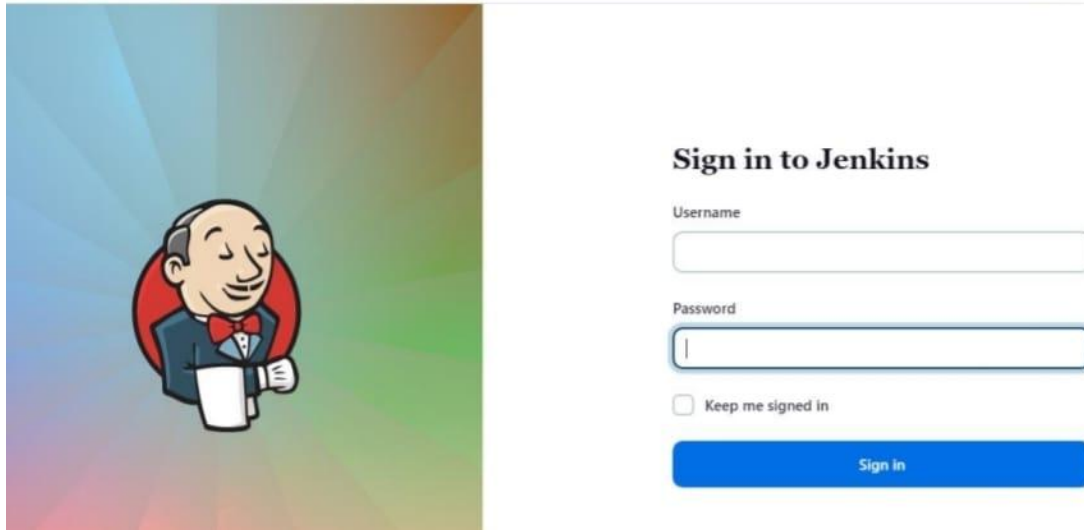
## GITHUB:



It shows that the git repository that which is used to deploy the

webapp in the Tomcat server. It is forked by the <https://github.com/Venna12/dockerjenkin.git>.

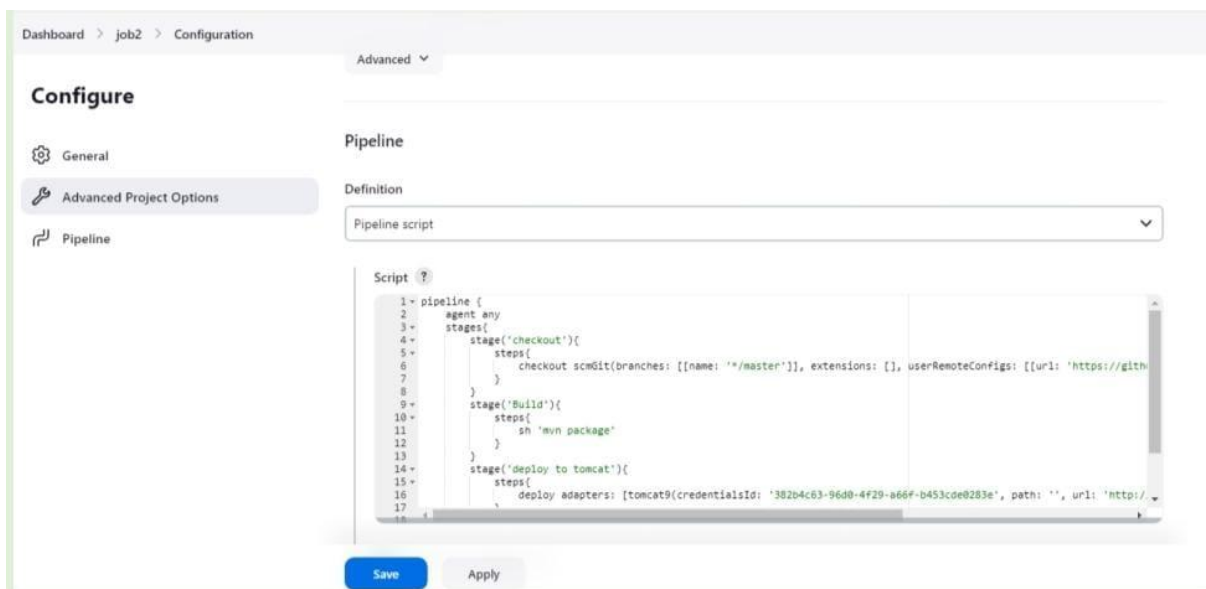
## JENKINS:



The image shows the Jenkins login interface. On the left is a large illustration of the Jenkins mascot, a man in a blue suit and red bow tie, holding a white cup. On the right, the heading "Sign in to Jenkins" is displayed above a form. The form includes a "Username" field, a "Password" field, a checkbox for "Keep me signed in", and a blue "Sign in" button.

This is the Sign page for Jenkins.

## PIPELINE:



The image displays the Jenkins "Configure" page for a job named "job2". The "Pipeline" tab is selected in the left sidebar. The "Definition" section shows "Pipeline script" selected. A code editor displays the following pipeline script:

```
1 pipeline {
2   agent any
3   stages{
4     stage('checkout'){
5       steps{
6         checkout scmGit(branches: [[name: '**/master']], extensions: [], userRemoteConfigs: [[url: 'https://github.com:Venna12/dockerjenkin.git']])
7       }
8     }
9     stage('Build'){
10      steps{
11        sh 'mvn package'
12      }
13    }
14    stage('deploy to tomcat'){
15      steps{
16        deploy adapters: [tomcat9(credentialsId: '382b4c63-96d0-4f29-a66f-b453cde0283e', path: '', url: 'http://localhost:8080/')]
17      }
18    }
19  }
20 }
```

At the bottom of the page are "Save" and "Apply" buttons.

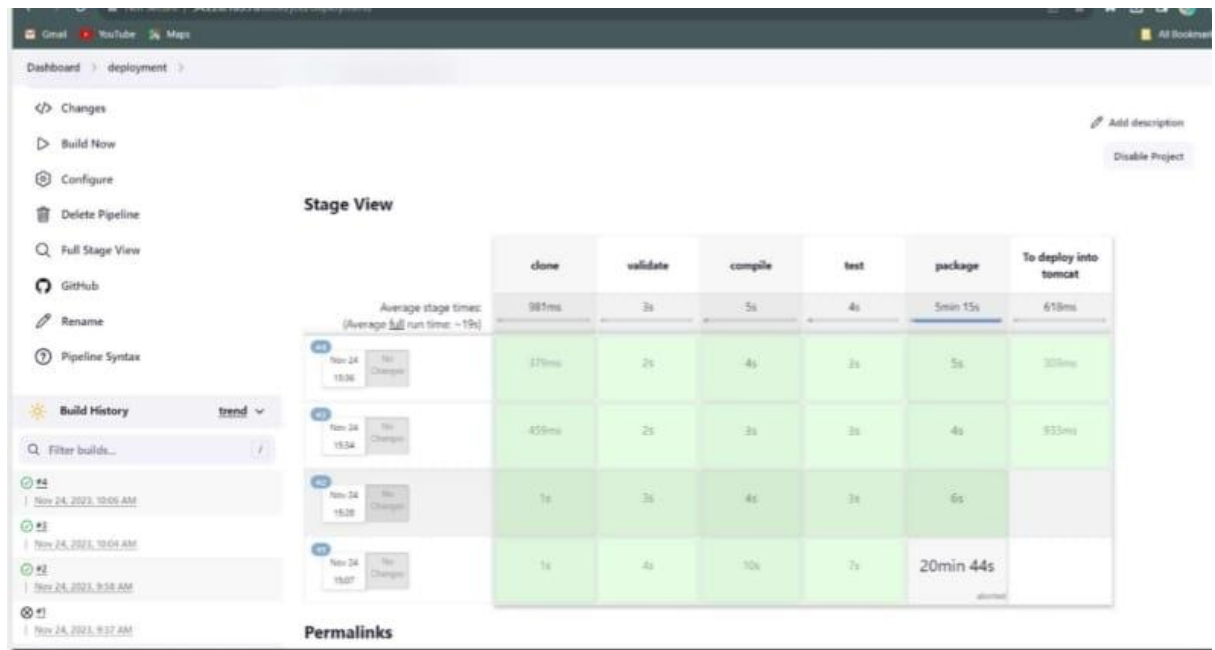
pipeline job creation in the Jenkins.

## — Execution of Application —

< We have to write the script for the pipeline to build the program, the script is Script

```
line syntax. Shown below. }
}
stage('mvn validate'){
  steps{
    sh 'mvn validate' pipeline {
      agent any
      stages {
        stage('Clone the Project'){
          steps{
            we get code from the pipe
          }
        }
      }
    }
    stage('mvn compile'){
      steps{
        sh 'mvn compile'
      }
    }
    stage('mvn test'){
      steps{
        sh 'mvn test'
      }
    }
    stage('mvn package'){
      steps{
        sh 'mvn package'
      }
    }
  }
}
```

Now, build the code and run the code.



Application successfully build. Now, Install plugin for the deployment in the tomcat.

## Deploy Container

To get tomcat script, we have to use pipeline syntax.

**Pipeline Syntax**

Steps

Sample Step

deploy: Deploy war/ear to a container

deploy

WAR/EAR files

\*\*/\*.war

War/ear files to deploy. Relative to the workspace root. You can also specify Ant-style GLOBs, like "\*\*/\*.war" (from Deploy to container Plugin)

Context path

/dockerjenkins

Containers

Add Container

☒ Deploy on failure

Generate Pipeline Script

Tomcat and the host manager



## Tomcat Web Application Manager

Message:	OK				
<b>Manager</b>					
<a href="#">List Applications</a>	<a href="#">HTML Manager Help</a>	<a href="#">Manager Help</a>	<a href="#">Server Status</a>		
<b>Applications</b>					
Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/test	None specified	Tomcat Test Manager Application	true	0	Start Stop Reload Undeploy

Now, after build the application . We will see the application in the host manager  
It is named as docker file.Now we can see the console output as shown below

## OUTPUT:

