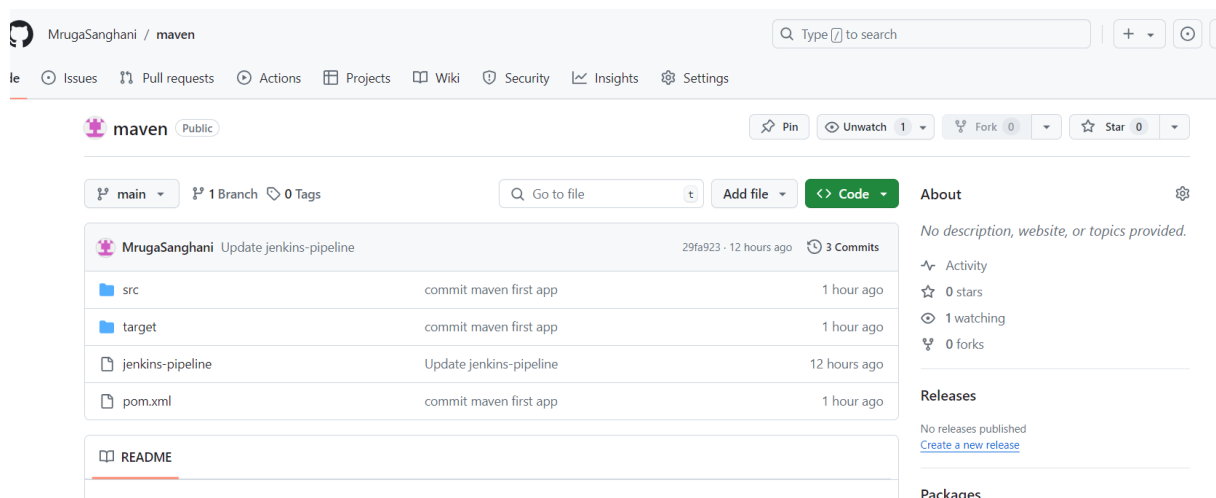


3. Create Jenkins job pipeline it from GitHub repo to run using build

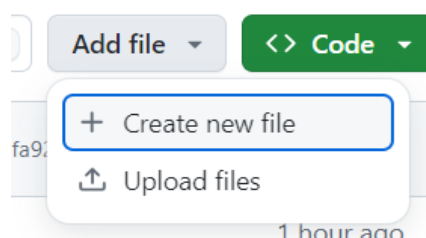
➤ Push the maven application on GitHub

- o Goto 2nd practical maven application folder
- o Open CMD
- o `$git init`
- o `$git add .`
- o `$git commit -m "commit maven first app"`
- o `$git remote add origin https://github.com/MrugaSanghani/maven.git`
- o `$git branch -M main`
- o `$git push -u origin main`

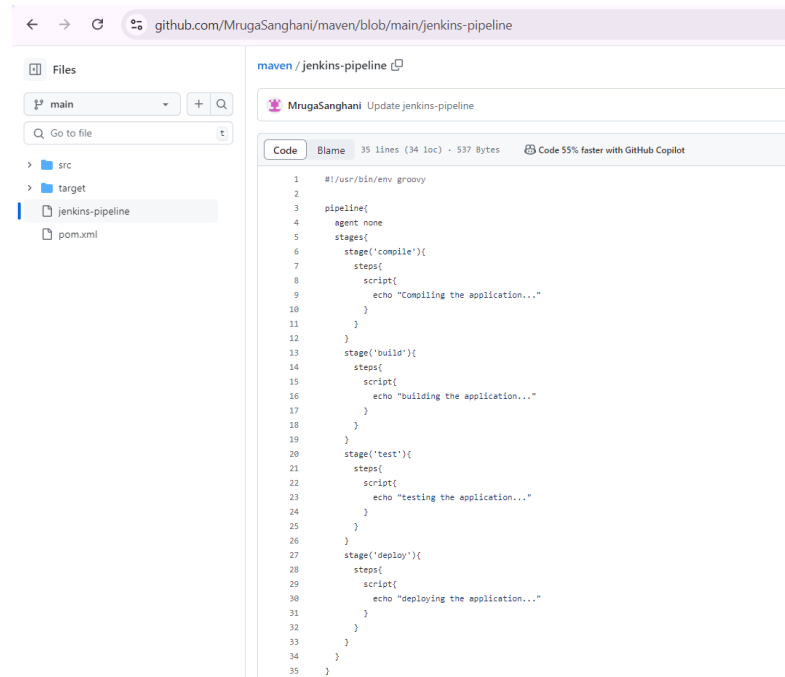


➤ Pipeline Script

- o Create a new-file in git repo



- o Give filename "Jenkins-pipeline"
- o Write following script in that file



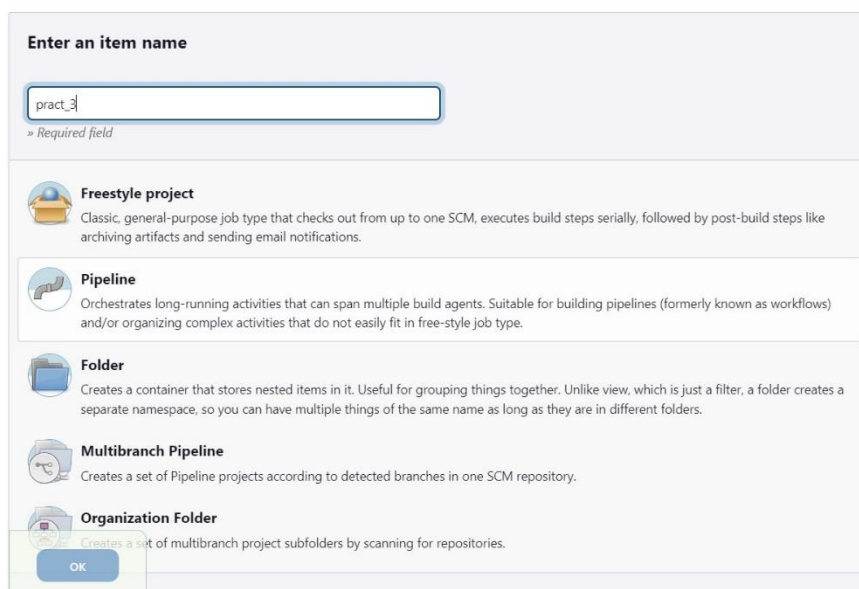
The screenshot shows a GitHub repository for 'maven' by user 'MrugaSanghani'. The file 'jenkins-pipeline' is selected, showing its content in a code editor. The file is 35 lines long, 34 loc, and 537 bytes. It contains a Jenkins pipeline script in Groovy.

```

1  #!/usr/bin/env groovy
2
3  pipeline{
4      agent none
5      stages{
6          stage('compile'){
7              steps{
8                  script{
9                      echo "Compiling the application..."
10                 }
11             }
12         }
13         stage('build'){
14             steps{
15                 script{
16                     echo "building the application..."
17                 }
18             }
19         }
20         stage('test'){
21             steps{
22                 script{
23                     echo "testing the application..."
24                 }
25             }
26         }
27         stage('deploy'){
28             steps{
29                 script{
30                     echo "deploying the application..."
31                 }
32             }
33         }
34     }
35 }
    
```

➤ **Click on the Jenkins Dashboard**

- Click on "new item"
- Give the project Name



The screenshot shows the 'Enter an item name' dialog in Jenkins. The input field contains 'pract_3'. Below the input field, there are several options for creating a new item:

- Freestyle project**: Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
- Pipeline**: Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Folder**: Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
- Multibranch Pipeline**: Creates a set of Pipeline projects according to detected branches in one SCM repository.
- Organization Folder**: Creates a set of multibranch project subfolders by scanning for repositories.

An 'OK' button is visible at the bottom left of the dialog.

Pipeline

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://github.com/MrugaSanghani/maven.git

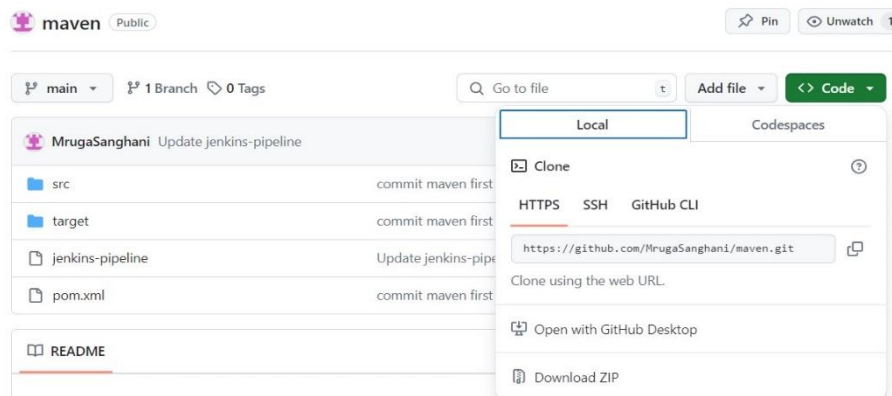
Credentials ?

- none -

+ Add

Advanced

Save Apply



- o Click on "Apply" and "save"

Branches to build ?

Branch Specifier (blank for 'any') ?

*/main

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add

Script Path ?

jenkins-pipeline

☒ Lightweight checkout ?

Pipeline Syntax

Save Apply

➤ Go to Jenkins and Build now in pract_3

The screenshot shows the Jenkins dashboard for a pipeline named 'pract_3'. The pipeline is in a 'Success' state, indicated by a green checkmark. The 'Stage View' shows four stages: 'compile', 'build', 'test', and 'deploy'. The 'Average stage times' are 447ms, 137ms, 114ms, and 131ms respectively. The 'Average full run time' is approximately 14s. The 'Build History' shows two builds: #2 (Aug 14, 2024, 12:10 PM) and #1 (Aug 14, 2024, 12:09 PM). The 'Permalinks' section provides links to the last build and the last stable build.

Stage	compile	build	test	deploy
Average stage times	447ms	137ms	114ms	131ms
#2 (Aug 14, 2024, 12:10)	447ms	137ms	114ms	131ms
#1 (Aug 14, 2024, 12:09)				

Permalinks

- Last build (#2), 5 hr 48 min ago
- Last stable build (#2), 5 hr 48 min ago

➤ Go to the Tools

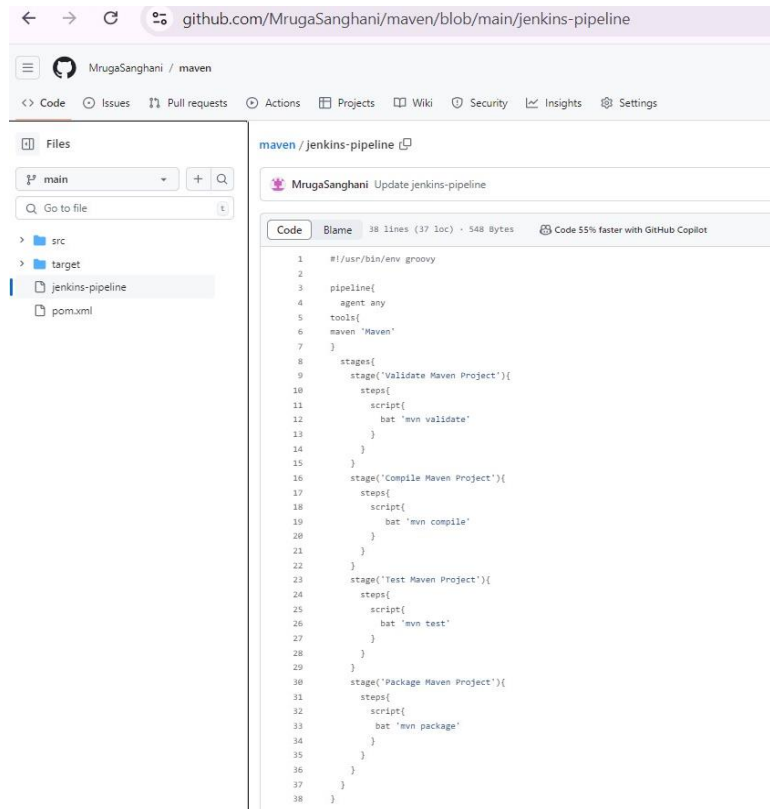
- Add Maven

The screenshot shows the Jenkins 'Tools' configuration page for Maven. The 'Maven installations' section is expanded, showing a table with one entry: 'Maven'. The 'Name' field is 'Maven', and the 'MAVEN_HOME' field is 'C:\Users\Lenovo\Desktop\SEM 7\devops lab\apache-maven-3.9.8'. The 'Install automatically' checkbox is unchecked. The 'Save' button is highlighted in blue, and a green 'Saved' notification is visible at the bottom.

Name	MAVEN_HOME	Install automatically
Maven	C:\Users\Lenovo\Desktop\SEM 7\devops lab\apache-maven-3.9.8	<input type="checkbox"/>

➤ Pipeline Script

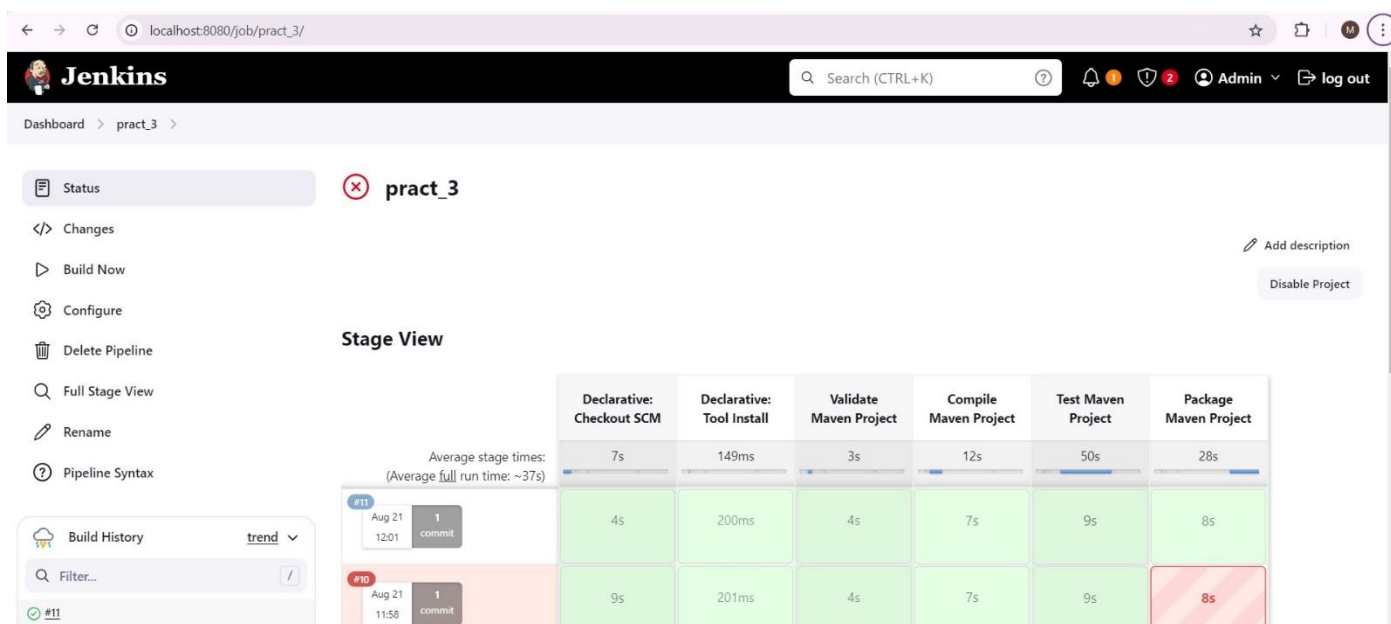
- o Write following script in "Jenkins-pipeline"



The screenshot shows a GitHub repository for 'maven' by 'MrugaSanghani'. The file 'jenkins-pipeline' is selected, displaying the following Groovy script:

```
1  #!/usr/bin/env groovy
2
3  pipeline{
4    agent any
5    tools{
6      maven 'Maven'
7    }
8    stages{
9      stage('Validate Maven Project'){
10       steps{
11         script{
12           bat 'mvn validate'
13         }
14       }
15     }
16     stage('Compile Maven Project'){
17       steps{
18         script{
19           bat 'mvn compile'
20         }
21       }
22     }
23     stage('Test Maven Project'){
24       steps{
25         script{
26           bat 'mvn test'
27         }
28       }
29     }
30     stage('Package Maven Project'){
31       steps{
32         script{
33           bat 'mvn package'
34         }
35       }
36     }
37   }
38 }
```

➤ Go to Jenkins and Build now in pract_3



The screenshot shows the Jenkins web interface for the 'pract_3' pipeline. The pipeline is in a failed state, indicated by a red 'X' icon. The 'Stage View' shows the following stages and their durations:

Stage	Declarative: Checkout SCM	Declarative: Tool Install	Validate Maven Project	Compile Maven Project	Test Maven Project	Package Maven Project
Avg	7s	149ms	3s	12s	50s	28s
#11 (Aug 21 12:01)	4s	200ms	4s	7s	9s	8s
#10 (Aug 21 11:58)	9s	201ms	4s	7s	9s	8s

The 'Package Maven Project' stage in the #10 build is highlighted in red, indicating a failure. The 'Build History' section shows the last two builds, both of which failed.