Lab Exercise 20
Creating a Pipeline Script

Objective: To create a pipeline script for automating build processes in Jenkins
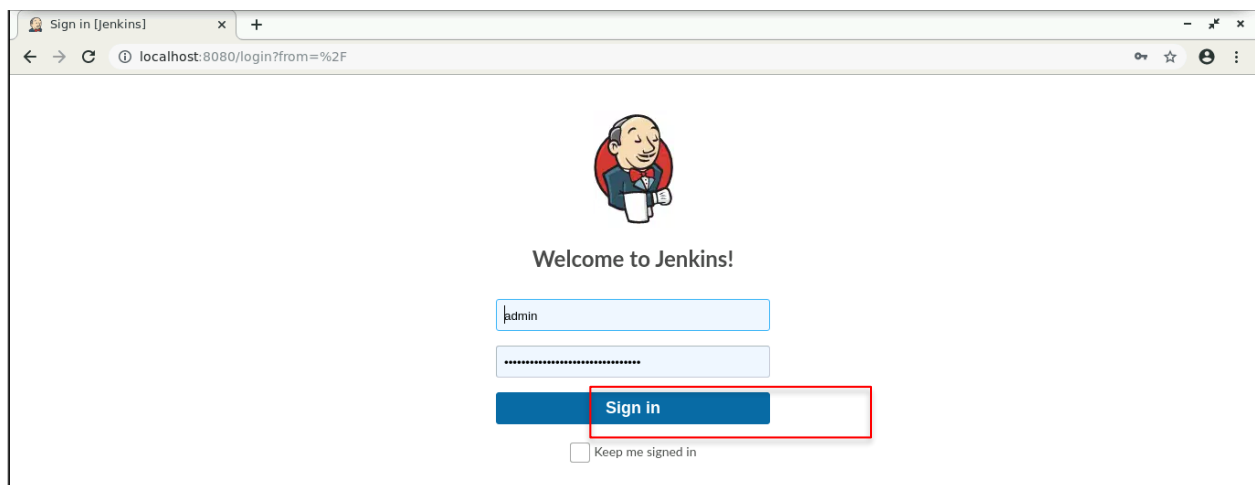
Tools required: Jenkins

Prerequisites: None

Steps to be followed:

1. Log in to the Jenkins CI tool and create a pipeline script

Step 1: Log in to the Jenkins CI tool and create a pipeline script

1. Open the browser, go to the Jenkins **Dashboard** by typing **localhost:8080** in your browser, provide the credentials, and click the **Sign in** button

Note: The credentials for accessing Jenkins in the lab are Username: **admin** and Password: **admin**.

2. Click on the **New Item** option as shown in the screenshot below:

OUTPUT :-

3. Enter a desired name for the project, select **Pipeline**, and then click on **OK** as shown in the screenshot below:

OUTPUT :-

4. Click on **Pipeline** as shown in the screenshot below:
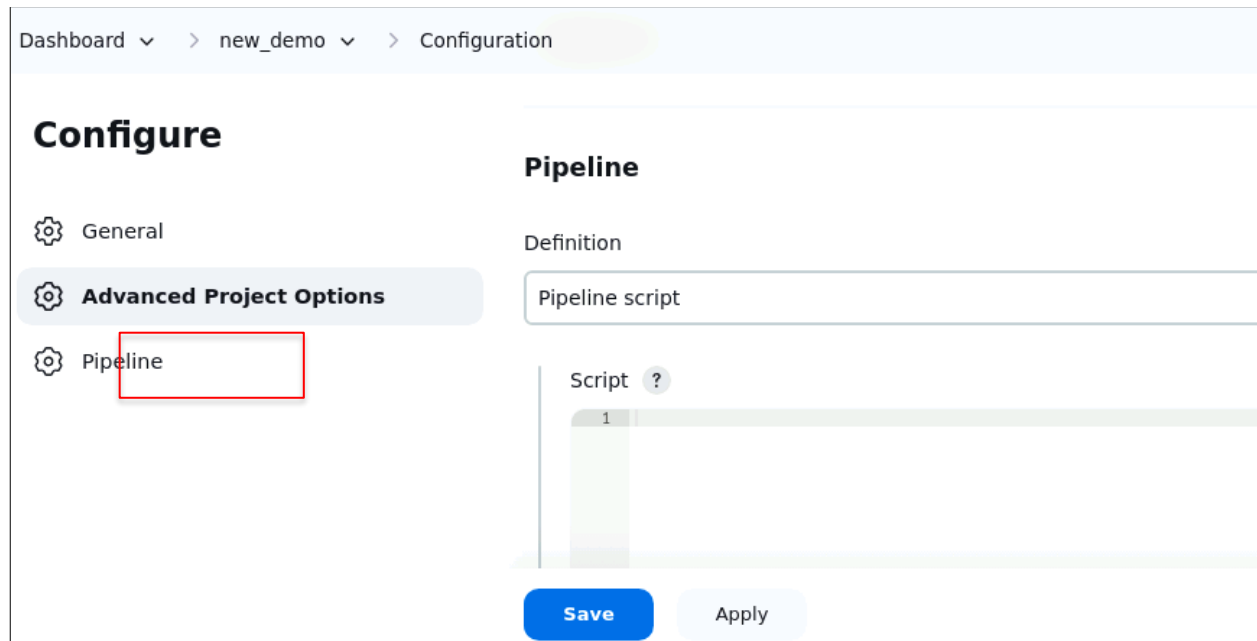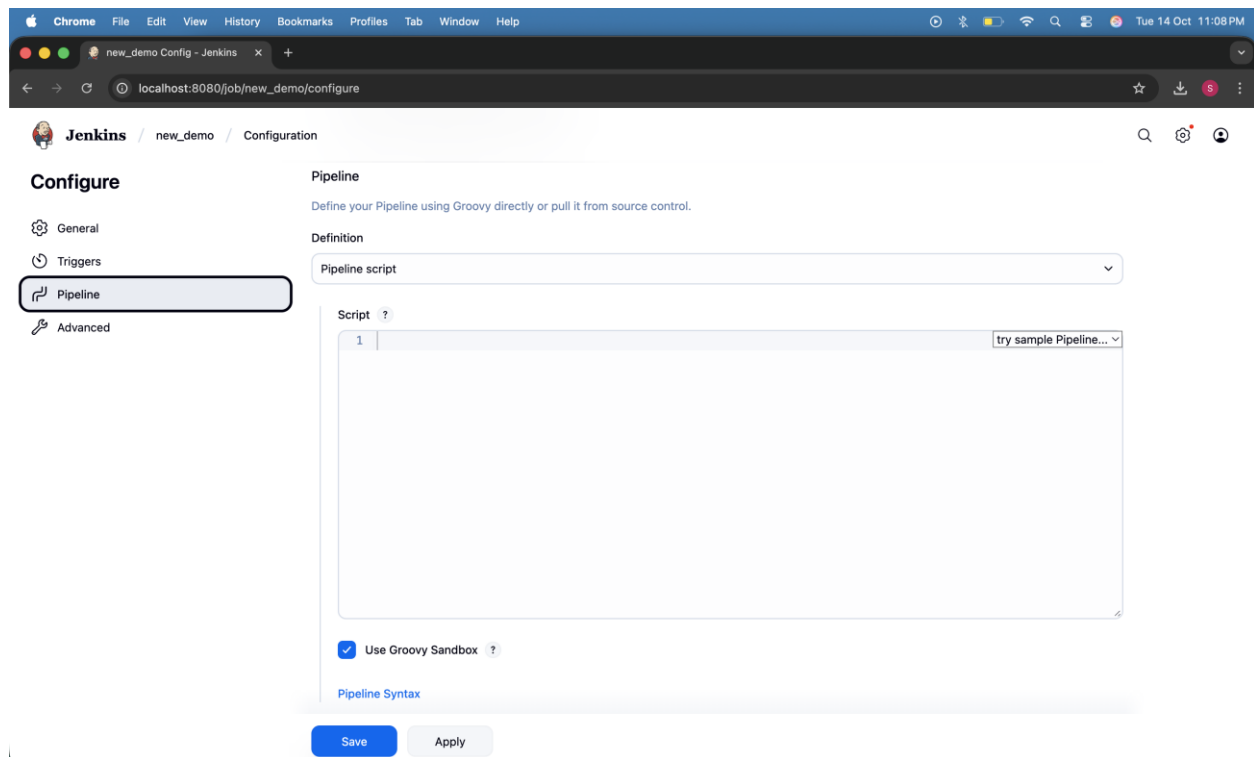
OUTPUT :-

5. Enter the following pipeline script in the script editor and click on **Save** as shown in the screenshot below:

pipeline {
  agent any
  stages {
    stage("hello") {
      steps{
        echo"welcome to Jenkins pipeline"
      }
    }
  }
}

Configure

Definition

Pipeline script

General

Advanced Project Options

Pipeline

Script ?

```
1 ▾ pipeline {
2       agent any
3 ▾     stages {
4 ▾         stage("hello") {
5 ▾             steps{
6                   echo"welcome to jenkins pipeline"
7               }
8           }
9       }
10  }|
```

Save   Apply
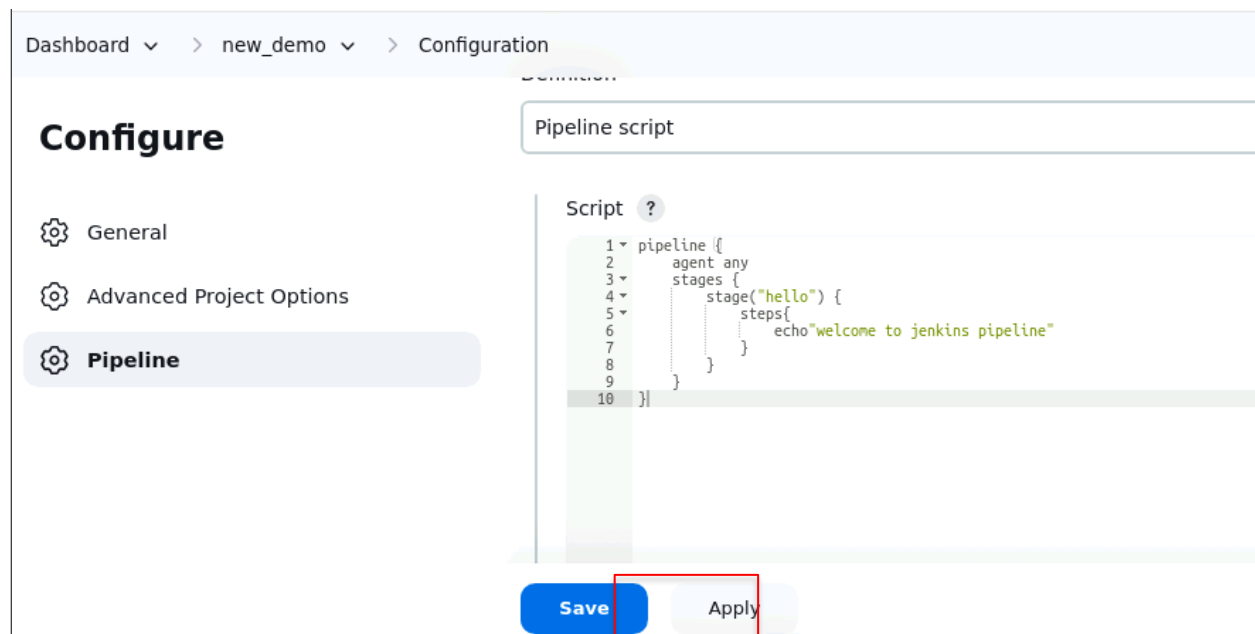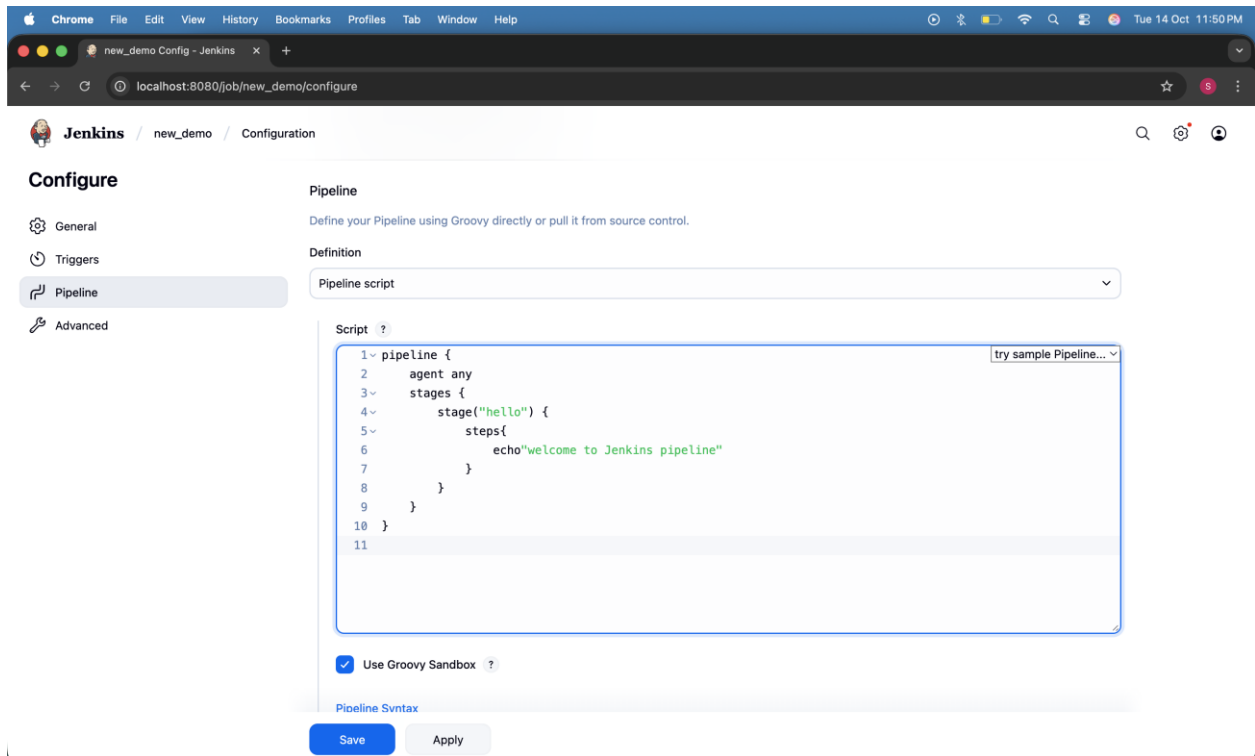
OUTPUT :-

6. Click on **Build Now** to run the pipeline script as shown in the screenshot below:
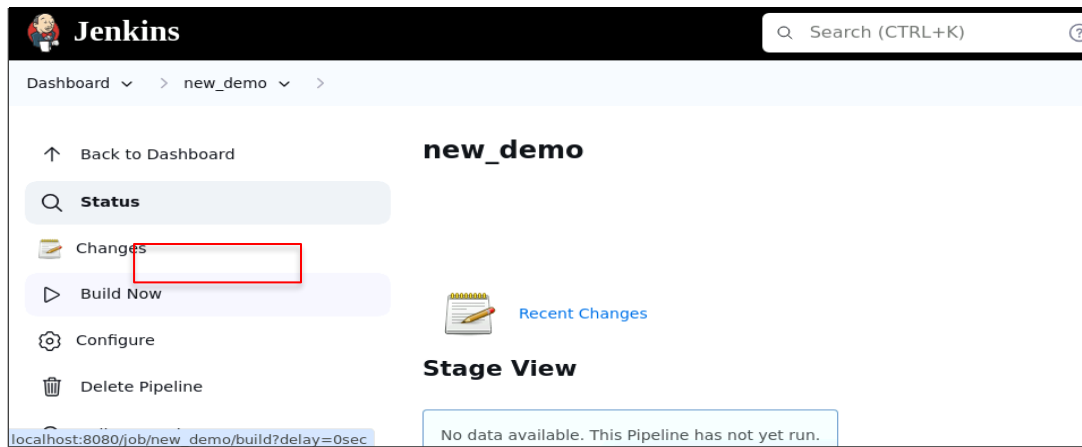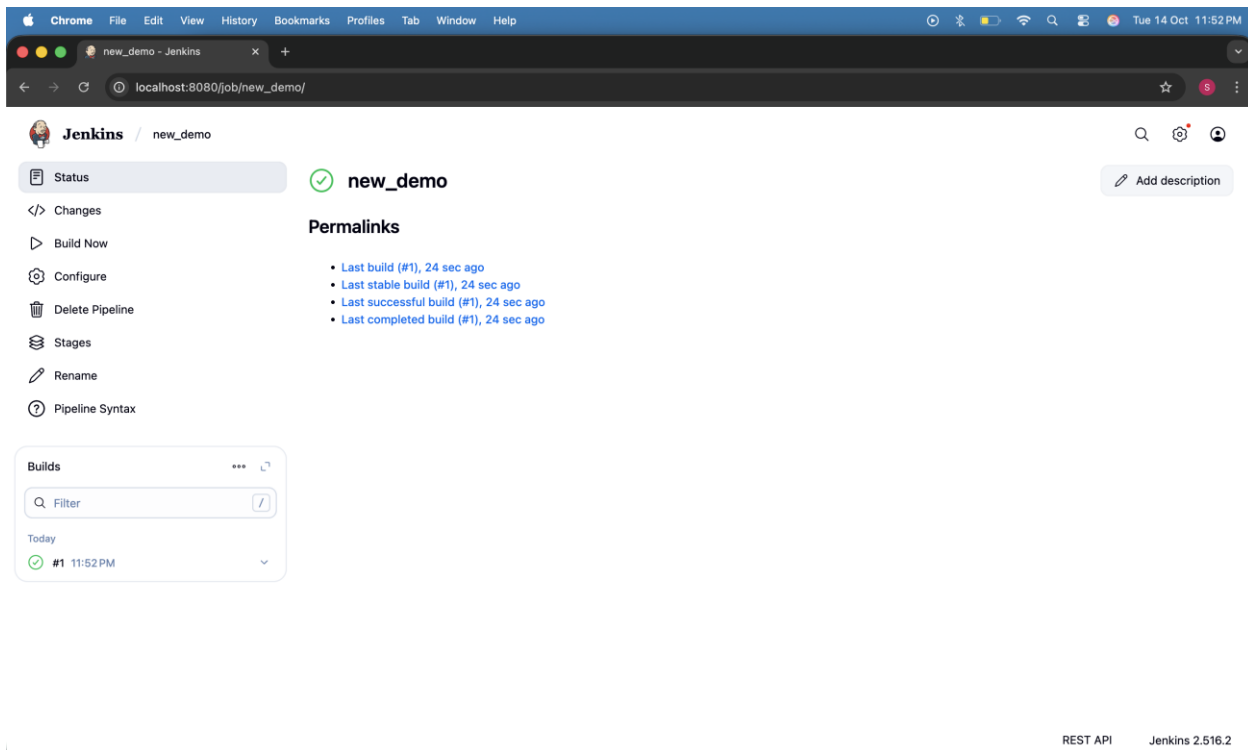
OUTPUT :-

7. Hover over the milliseconds number next to the build stage name as shown in the screenshot below:
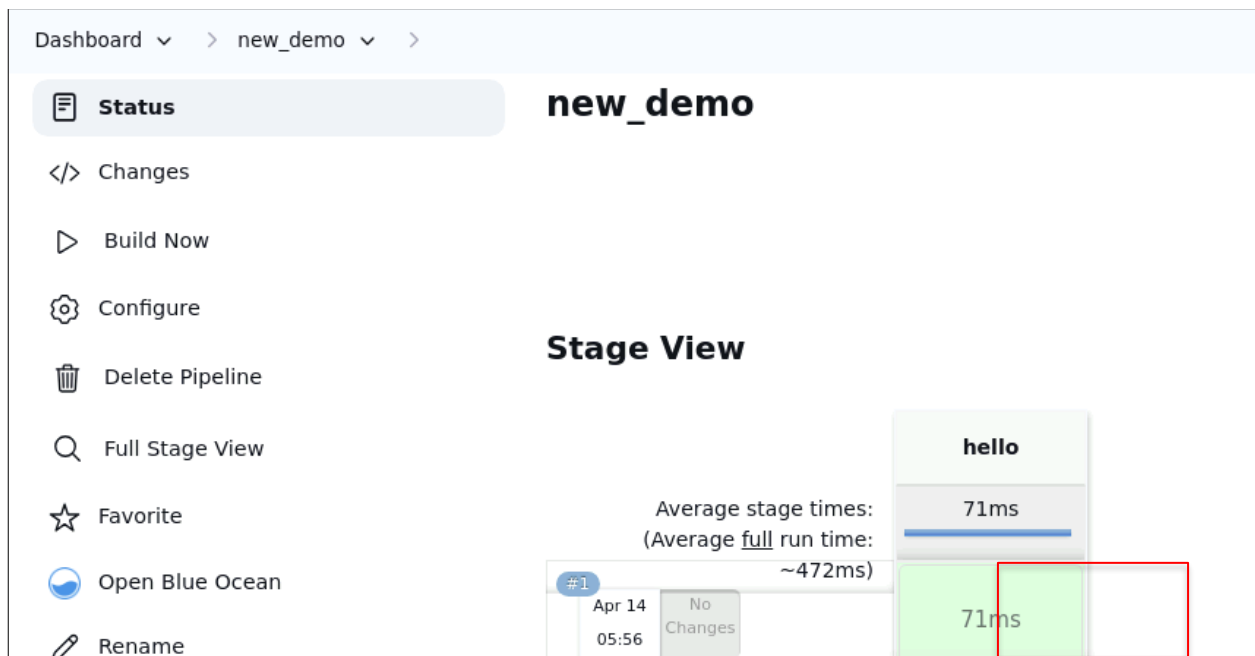
Note: Ensure that you hover the cursor over the milliseconds number without clicking on it

OUTPUT :-

8. Click on **Logs** as shown in the screenshot below:

OUTPUT :-

Check for the message in the top-left corner to confirm the successful execution of the pipeline stage as shown in the screenshot below:
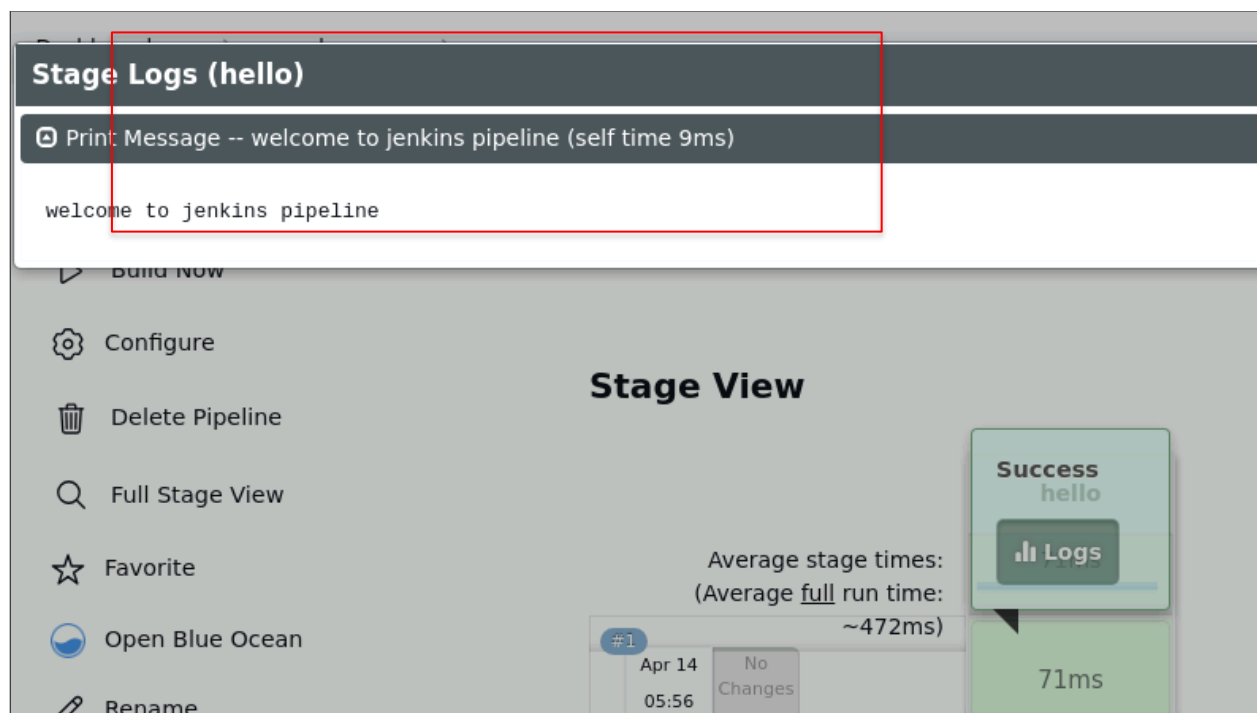
By following these steps, you have successfully created a pipeline script for automating build processes in Jenkins.
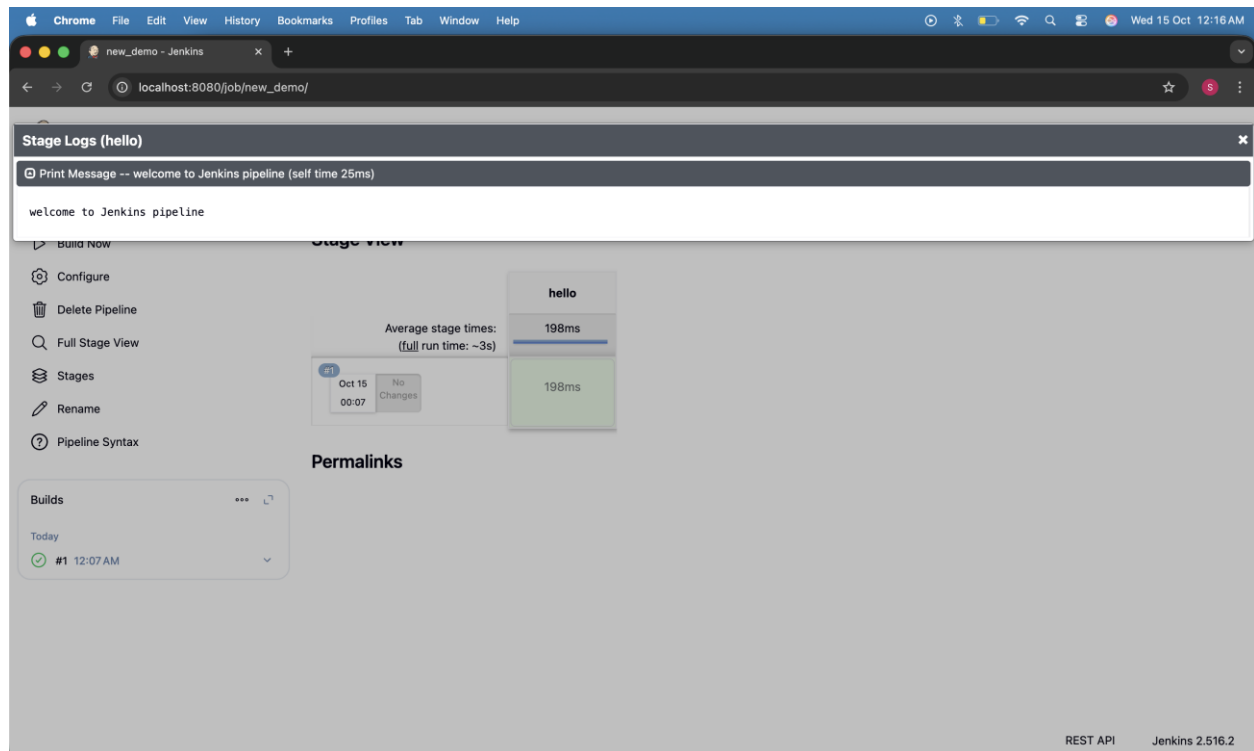
OUTPUT :-

Creating a pipeline for a project

1) Create a built using pipeline
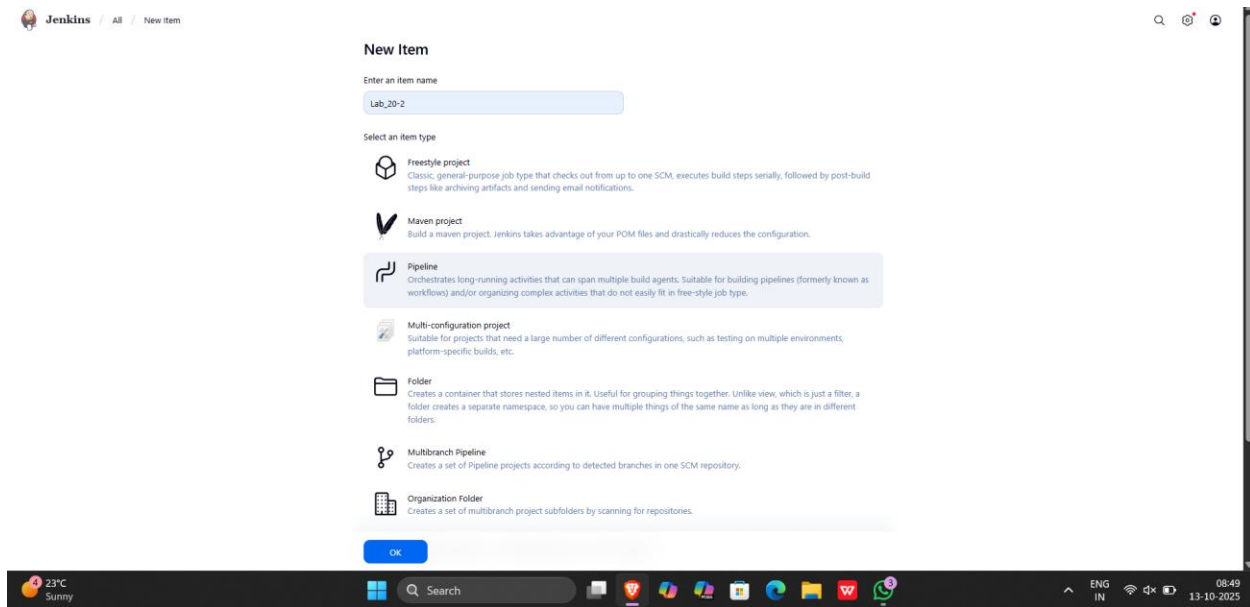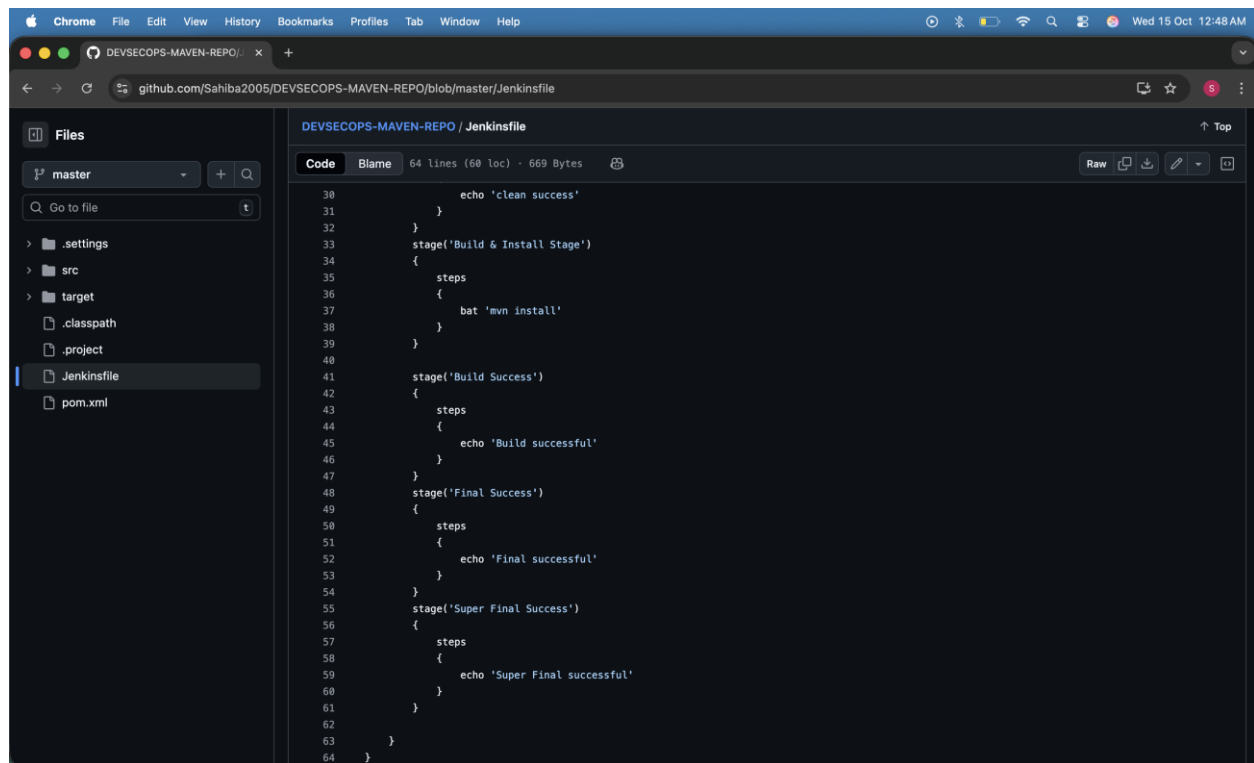
OUTPUT :-
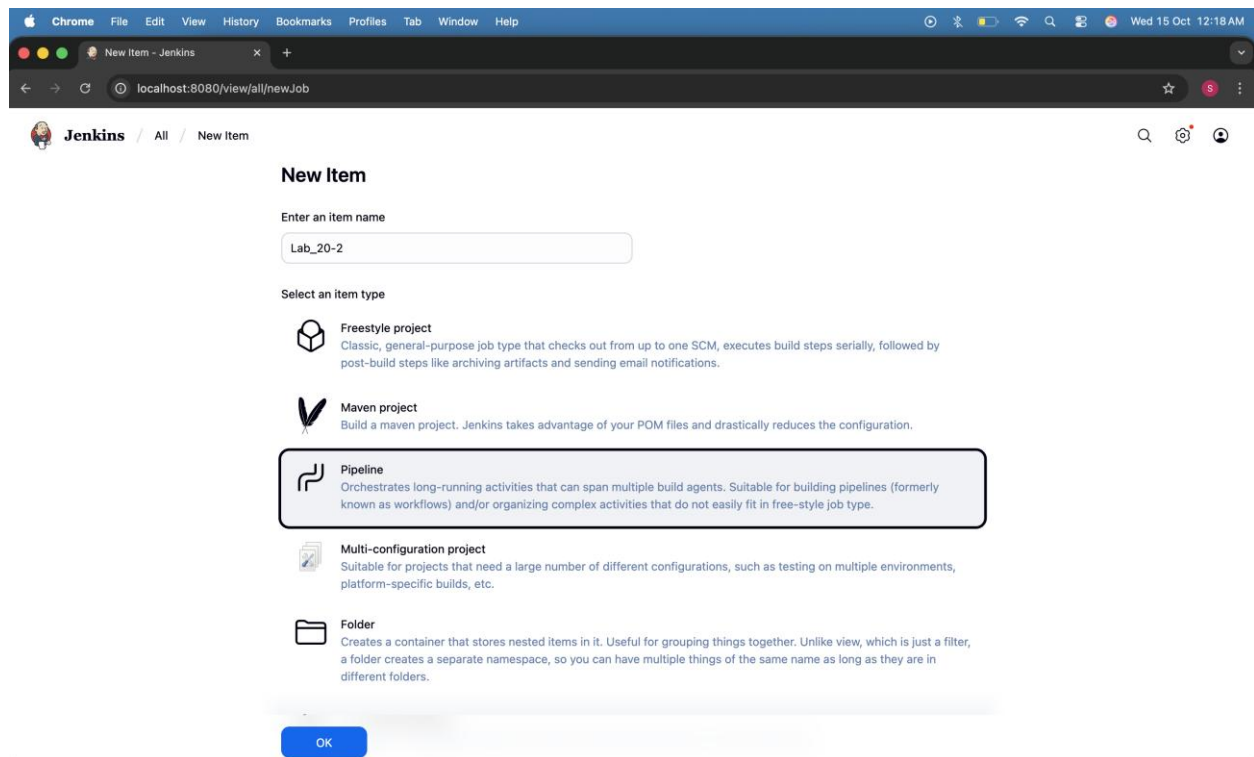
2) write the script for the pipeline in jenkinsfile in github

Files

master

Go to file

> .settings
> src
> target
  .classpath
  .project
  Jenkinsfile
  pom.xml

DEVSECOPS-MAVEN-REPO / Jenkinsfile

Sahiba2005 Update Jenkinsfile   0bbe8d6 · now   History

Code   Blame   64 lines (60 loc) · 669 Bytes   Raw

```
 1    pipeline
 2    {
 3        agent any
 4        tools
 5        {
 6            maven 'MAVEN'
 7        }
 8
 9        stages
10        {
11            stage('Welcome Stage')
12            {
13                steps
14                {
15                    echo 'Welcome to Jenkins Pipeline'
16                }
17            }
18
19            stage('Clean Stage')
20            {
21                steps
22                {
23                    bat 'mvn clean'
24                }
25            }
26            stage('Clean Success Stage')
27            {
28                steps
29                {
30                    echo 'clean success'
31                }
32            }
```

3) Add the repository url in github project and in pipeline

Jenkins   /   Lab_20-2   /   Configuration

Configure

- ⚙ General
- ⏱ Triggers
- Pipeline
- 🔧 Advanced

☐ Do not allow the pipeline to resume if the controller restarts

☑ GitHub project

Project url   ?

https://github.com/Sahiba2005/DEVSECOPS-MAVEN-REPO

Advanced ⌄

☐ Pipeline speed/durability override   ?

☐ Preserve stashes from completed builds   ?

☐ This project is parameterized   ?

☐ Throttle builds   ?

Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

☐ Build after other projects are built   ?

☐ Build periodically   ?

☐ GitHub hook trigger for GITScm polling   ?

☑ Poll SCM   ?

Save   Apply
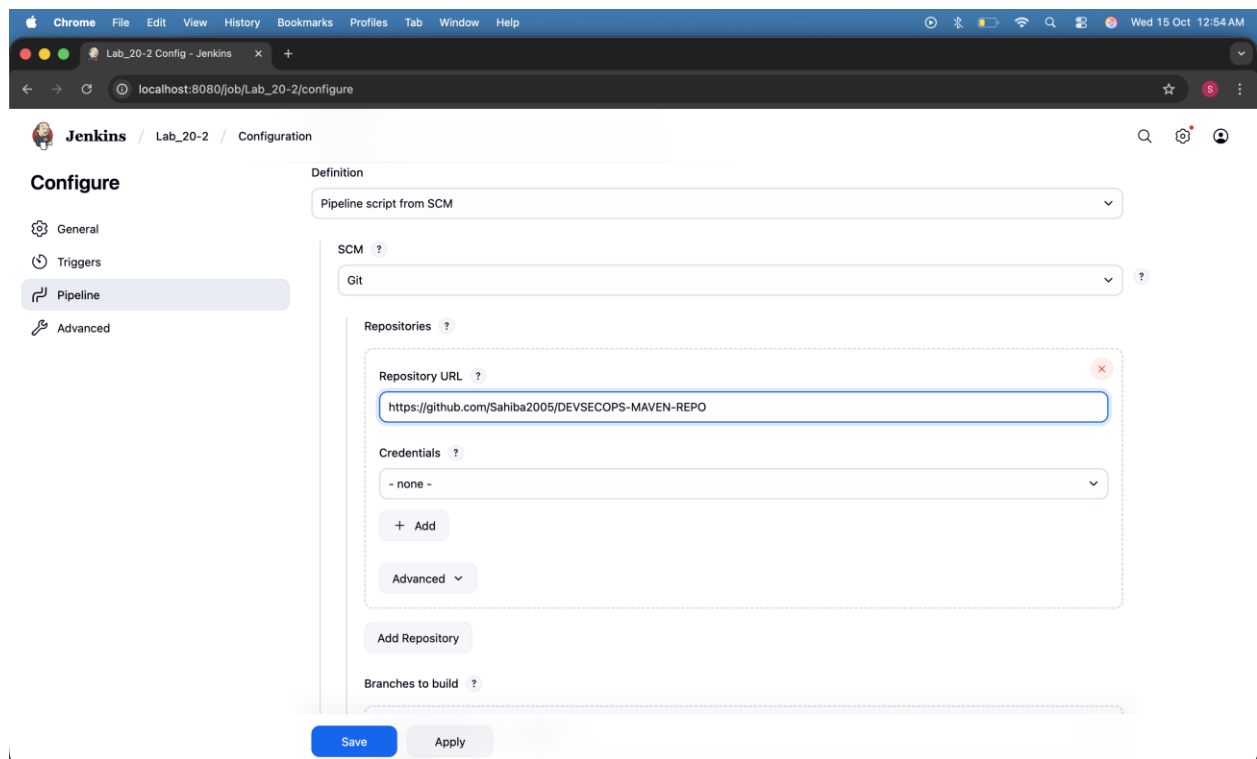
4) check poll scm option
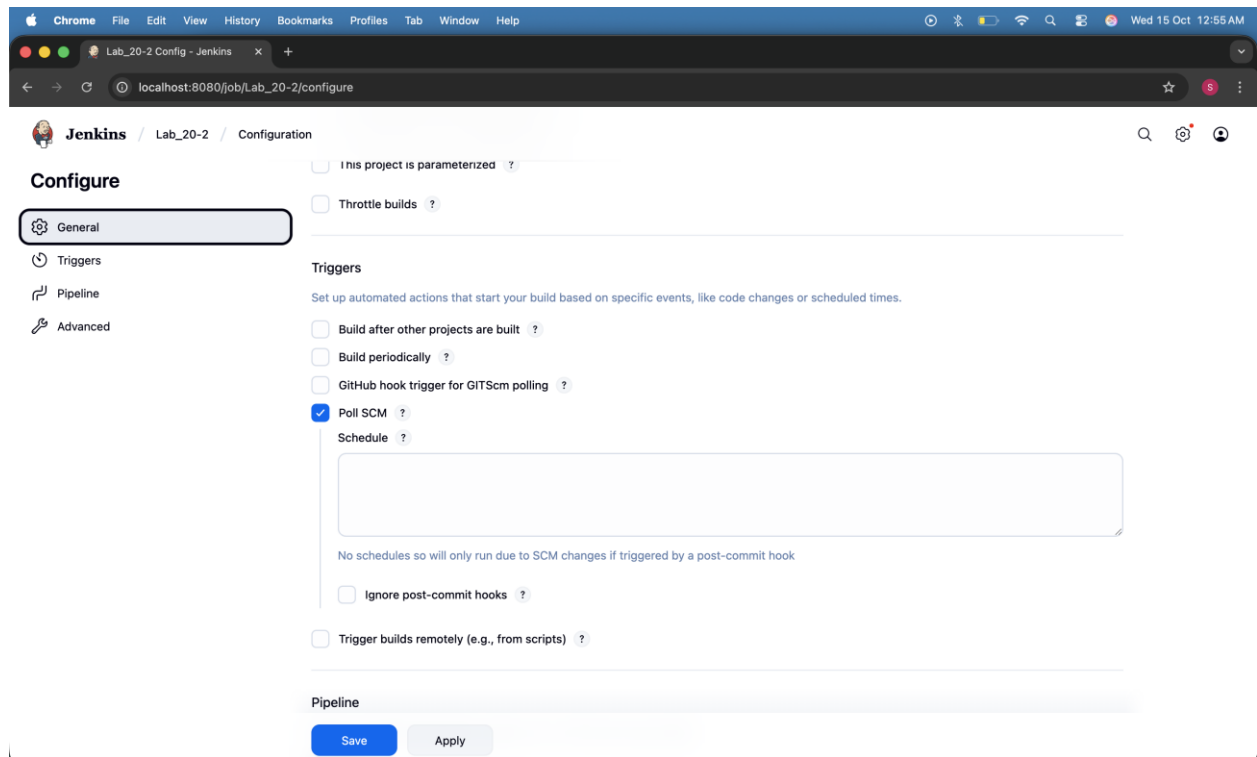
OUTPUT :-

Stage Logs (Super Final Success)   ✕

⊕ Use a tool from a predefined Tool Installation -- MAVEN_HOME (self time 34ms)

⊕ Fetches the environment variables for a given tool in a list of 'FOO=bar' strings suitable for the withEnv step. (self time 39ms)

⊕ Print Message -- Super Final successful (self time 20ms)

		Declarative: Checkout SCM	Declarative: Tool Install	Welcome Stage	Clean Stage	Clean Success Stage	Build & Install Stage	Build Success	Final Success	Super Final Success
⚙ Configure										
🗑 Delete Pipeline										
🔍 Full Stage View	Average stage times: (full run time: ~12s)	1s	150ms	247ms	2s	221ms	4s	248ms	216ms	215ms
🐙 GitHub										
🗂 Stages	#1 Oct 15 01:02 No Changes	1s	150ms	247ms	2s	221ms	4s	248ms	216ms	215ms
✏ Rename										
❓ Pipeline Syntax										
📋 Git Polling Log										

Permalinks

Builds   ••• ⤢

Today

✅ #1 1:02 AM   ⌄

REST API   Jenkins 2.516.2

REST API   Jenkins 2.516.2

Pipeline script

```
pipeline

{

agent any

tools

{

maven 'Maven'

}
```

```
stages
{
stage('Welcome Stage')
{
steps
{
echo 'Welcome to Jenkins Pipeline'
}
}

stage('Clean Stage')
{
steps
{
sh 'mvn clean'
}
}
        stage('Clean Success Stage')
{
steps
{
echo 'clean success'
}
}
stage('Build & Install Stage')
{
steps
{
Sh 'mvn install'
}
}

stage('Build Success')
{
```

```
steps
{
echo 'Build successful'
}
}
stage('Final Success')
{
steps
{
echo 'Final successful'
}
}

}
}
```