

13.Demonstrate how Postman is used with Jenkins.

To use Postman in Jenkins, you can use the Postman Integration Plugin. This plugin allows you to run Newman, the Postman command-line runner, from within Jenkins.

To install the Postman Integration Plugin, go to Manage Jenkins > Manage Plugins and search for "Postman Integration". Click the Install button to install the plugin.

Once the plugin is installed, you can create a new Jenkins job to run your Postman collection. To do this, go to New Item and select Postman Job.

In the Job Name field, enter a name for your job. In the Collection field, select the Postman collection that you want to run. In the Environment field, select the Postman environment that you want to use.

You can also configure other options for your job, such as the Exit Code handling and the Post-build Actions.

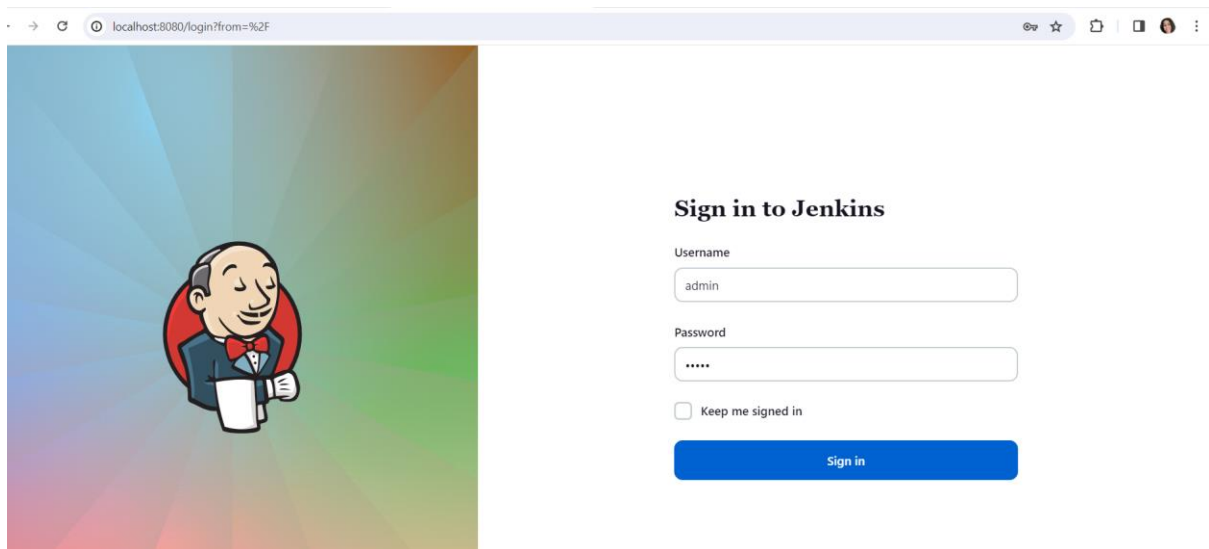
Once you have configured your job, click the Save button.

To run your job, click the Build Now button. Jenkins will run Newman to execute your Postman collection.

You can view the results of the job in the Jenkins console. The console will show you the status of each request in the collection.

You can also use the Postman Integration Plugin to generate reports from your Postman collection runs. To do this, click the Generate Reports button in the Jenkins console.

The Postman Integration Plugin is a powerful tool that can help you to automate your API testing with Postman and Jenkins.

A screenshot of a web browser showing the Jenkins login page. The browser's address bar displays 'localhost:8080/login?from=%2F'. The page features a large, colorful, abstract background image on the left side, depicting a stylized figure in a tuxedo. On the right side, there is a 'Sign in to Jenkins' section with a form. The form includes a 'Username' field with the text 'admin', a 'Password' field with masked characters '.....', and a checkbox labeled 'Keep me signed in'. A blue 'Sign in' button is positioned below the form fields.

localhost:8080/login?from=%2F

Sign in to Jenkins

Username

Password

☐ Keep me signed in

Sign in

Plugins

🔍 nodejs

Name ↓

NodeJS Plugin 1.6.1

NodeJS Plugin executes **NodeJS** script as a build step.

[Report an issue with this plugin](#)

DevOps-Notes-Oct'23 - Google

Dashboard

Tools [Jenkins]


+

localhost:8080/manage/configureTools/

Dashboard > Manage Jenkins > Tools

Add Ant


Maven installations

Maven installations ▾  Edited

JFrog CLI installations

Add JFrog CLI

NodeJS installations

NodeJS installations ▾  Edited

Save

Apply

NodeJS installations ✎ Edited

Add NodeJS

NodeJS

Name

Installation directory

☐ Install automatically ?

Add NodeJS

Phase3-Lesson2-APITesting New Import GET LocalValue CollectionRUN-newman dev

Collections

- ApiTesting-Lesson2
 - CollectionRUN-newman

Environments

- dev

History

- GET Ge
- HEAD HE
- OPT Oc
- PATCH PA

Variables

- Generate tests BETA
- Edit
- Add request
- Add folder
- Monitor collection
- Mock collection
- Create a fork Ctrl+Alt+F
- Create pull request

CollectionRUN-newman

Overview Authorization Pre-request Script Tests Variables Runs

CollectionRUN-newman

Make things easier for your teammates with a complete collection description.

[View complete documentation →](#)

Created by You

Created on 03 Nov 2023, 12:40 PM

Phase3-Lesson2-APITesting New Import GET LocalValue CollectionRUN-newman Runner dev

Collections

- ApiTesting-Lesson2
 - CollectionRUN-newman

Environments

- dev

History

- GET GetMethod-Demo
- HEAD HEADMethod-Demo
- OPT OptionsMethod-Demo
- PATCH PATCHMethod

Variables

- Variables-Demo

Run order

Deselect All Select All Reset

- ☒ GET GetMethod-Demo
- ☒ HEAD HEADMethod-Demo
- ☒ OPT OptionsMethod-Demo
- ☒ PATCH PATCHMethod

Functional Performance

Choose how to run your collection

- ☐ Run manually
Run this collection in the Collection Runner.
- ☐ Schedule runs
Periodically run collection at a specified time on the Postman Cloud.
- ☒ Automate runs via CLI
Configure CLI command to run on your build pipeline.

Run on Postman CLI

Automate using Postman's command-line tool. [Download Postman CLI](#)

Copy this command and run it in your local terminal.

```
postman login --with-api-key Add API Key  
postman collection run 17738345-fd5882de-7fba-4b2d-aeb3-678d24b42892 -e 17738345-4899fcd6-7cae-46c2-9546-c84a54668499
```

① You can view all your runs for this collection under the [runs](#) tab.

Run on CI/CD

[Configure command](#) to run collection on CI/CD pipeline.

Click on configure command

Phase3-Lesson2-APITesting New Import GET LocalValue CollectionRUN-newman Runner + dev

Collections

- ApiTesting-Lesson2
 - CollectionRUN-newman
 - GET GetMethod-Demo
 - HEAD HEADMethod-Demo
 - OPT OptionsMethod-Demo
 - PATCH PATCHMethod
 - Variables-Demo

Run order

Deselect All Select All Reset

- ☒ GET GetMethod-Demo
- ☒ HEAD HEADMethod-Demo
- ☒ OPT OptionsMethod-Demo
- ☒ PATCH PATCHMethod

Functional Performance

Choose how to run your collection

- ☐ Run manually
Run this collection in the Collection Runner.
- ☐ Schedule runs
Periodically run collection at a specified time on the Postman Cloud.
- ☒ Automate runs via CLI
Configure CLI command to run on your build pipeline.

Run on Postman CLI

Automate using Postman's command-line tool. [Download Postman CLI](#)

Copy this command and run it in your local terminal.

```
postman login --with-api-key Add API Key
postman collection run 17730345-fd5082de-7fba-4b2d-aeb3-678d24b42892 -e 17730345-4899fcd6-7cae-46c2-9546-ca4a54668499
```

You can view all your runs for this collection under the runs tab.

Run on CI/CD

Configure command to run collection on [CI/CD pipeline](#).

Phase3-Lesson2-APITesting New Import GET LocalValue CollectionRUN-newman Runner Postman CLI Configuration + dev

Collections

- ApiTesting-Lesson2
 - CollectionRUN-newman
 - GET GetMethod-Demo
 - HEAD HEADMethod-Demo
 - OPT OptionsMethod-Demo
 - PATCH PATCHMethod
 - Variables-Demo

Generate Postman CLI Configuration

Run your tests and validations on CI/CD using Postman CLI configuration. [Download Postman CLI](#)

Choose the collections you want to run

Collection: Variables-Demo Environment: dev

CI/CD configuration

CI/CD Provider: Jenkins Operating system for CI/CD: Windows

Postman CLI command preview

Jenkins

Search (CTRL+K)

Dashboard > All >

Enter an item name

postmanpipeline1

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

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.

Now copy the entire script form postman

Postman CLI command preview

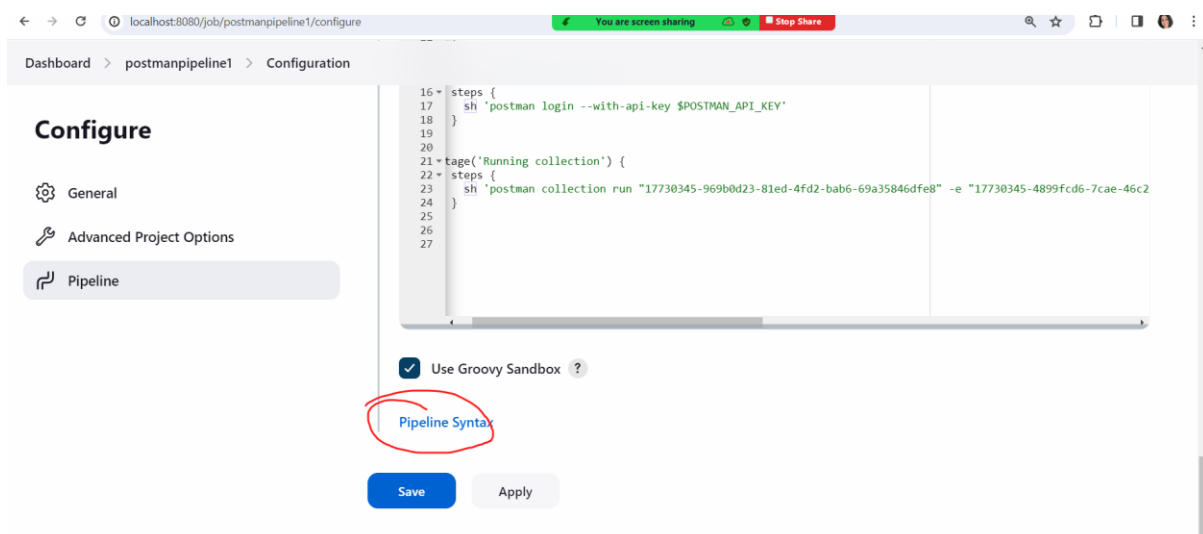
```
Copy this command and paste it to your build configuration file

2 # install the server on a windows machine by following https://www.jenkins.io/doc/book/installing/windows/
3 # once it has been setup add the below step to your pipeline file to run automated tests using Postman CLI.
4
5 pipeline {
6   agent any
7
8   tools {nodejs "your_nodejs_configured_tool_name"}
9
10  stages {
11    stage('Install Postman CLI') {
12      steps {
13        sh 'powershell.exe -NoProfile -InputFormat None -ExecutionPolicy AllSigned -Command "[System.Net.
14          ServicePointManager]::SecurityProtocol = 3072; iex ((New-Object System.Net.WebClient).DownloadString
15            ('https://dl-cli.pstmn.io/install/win64.ps1'))'"
16      }
17    }
18  }
19
20  stage('Postman CLI Login') {
21    steps {
22      sh 'postman login --with-api-key $POSTMAN_API_KEY'
23    }
24  }
25
26  stage('Running collection') {
27    steps {
28      sh 'postman collection run "17730345-969b0d23-81ed-4fd2-bab6-69a35846dfe8" -e "17730345-4899fcd6-7cae-46c2
29    }
30  }
31}
```

Note: Postman API key is needed to log in to Postman CLI

Copy this pipeline in Jenkins Job

Click on Pipeline syntax



Here select windows batch script

↑ Back

⚙ Snippet Generator

⚙ Declarative Directive Generator

🔍 Declarative Online Documentation

🔍 Steps Reference

🔍 Global Variables Reference

🔍 Online Documentation

🔍 Examples Reference

Overview

This **Snippet Generator** will help you learn the Pipeline Script code which can be used to define various steps. Pick a step you are interested in from the list, configure it, click **Generate Pipeline Script**, and you will see a Pipeline Script statement that would call the step with that configuration. You may copy and paste the whole statement into your script, or pick up just the options you care about. (Most parameters are optional and can be omitted in your script, leaving them at default values.)

Steps

Sample Step

bat: Windows Batch Script

> Pipeline Syntax

Sample Step

bat: Windows Batch Script

bat

Batch Script ?

```
powershell.exe -NoProfile -InputFormat None -ExecutionPolicy AllSigned -Command "[System.Net.ServicePointManager]::SecurityProtocol = 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://dl-cli.pstmn.io/install/win64.ps1'))"
```

Advanced ▾

Generate Pipeline Script

```
bat 'powershell.exe -NoProfile -InputFormat None -ExecutionPolicy AllSigned -Command "[System.Net.ServicePointManager]::SecurityProtocol = 3072; iex ((New-Object System.Net.WebClient).DownloadString('\https://dl-cli.pstmn.io/install/win64.ps1\'))'
```

Now go to jenkins job.. remove the sh command in first stage and add the bta command

← → ↺

🔍 ☆ 📄 📱 🧑

Dashboard > postmanpipeline1 > Configuration

```

2
3 pipeline {
4   agent any
5
6   tools {nodejs 'node'}
7
8   stages {
9     stage('Install Postman CLI') {
10      steps {
11        bat 'powershell.exe -NoProfile -InputFormat None -ExecutionPolicy AllSigned -Command "[System.Net.ServicePointManager]::SecurityProtocol = 3072; iex ((New-Object System.Net.WebClient).DownloadString('\https://dl-cli.pstmn.io/install/win64.ps1\'))'
12      }
13    }
14  }

```

try sample pipeline...

Get the api key from postman

The screenshot displays the 'API keys' management page in the Sonal application. The left sidebar contains navigation links: Profile, Account, Notifications, Active sessions, Teams, and API keys (which is highlighted). The main content area is titled 'API keys' and includes a 'Generate API Key' button. Below this is a description: 'Fetch live Postman data to integrate with your build system and tool chain, auto-export to your repository and even create converters.' The 'API key settings' section shows two options: 'Never expire API keys' (unselected) and 'API keys expire after 60 days of inactivity' (selected). A 'Save API Key Settings' button is present. The 'API keys' table lists three keys: 'testKey', 'keyrest', and 'newkey'. The 'newkey' row is highlighted, and a context menu is open over it, showing 'Regenerate', 'Rename', and 'Delete' options. A 'Regenerate API key' dialog box is overlaid on the screen, containing the text: 'This will regenerate your API key and provides you with a new one. Your existing key will be automatically deleted in 6 hours.' There is a checked checkbox for 'Delete the existing key now' and two buttons: 'Cancel' and 'Regenerate API Key'.

Sonal
View your profile

Profile
Account
Notifications
Active sessions
Teams
API keys

API keys

Fetch live Postman data to integrate with your build system and tool chain, auto-export to your repository and even create converters.

API key settings

☐ Never expire API keys ☒ API keys expire after 60 days of inactivity

Save API Key Settings

Name	API key	Created on	Last used	Status
testKey	PMak-61a45a21d78bd6003bdb8b21-XXXX	29 Nov, 2021	6 Dec, 2021	⚠️ ...
keyrest	PMak-61ade3ac0647ea0035bd0a63-XXXX	6 Dec, 2021	7 Dec, 2021	⚠️ ...
newkey	PMak-6543b7fb6d0f58002a8ebe36-XXXX	2 Nov, 2023	5 Nov, 2023	🔴 ...

Collection access keys

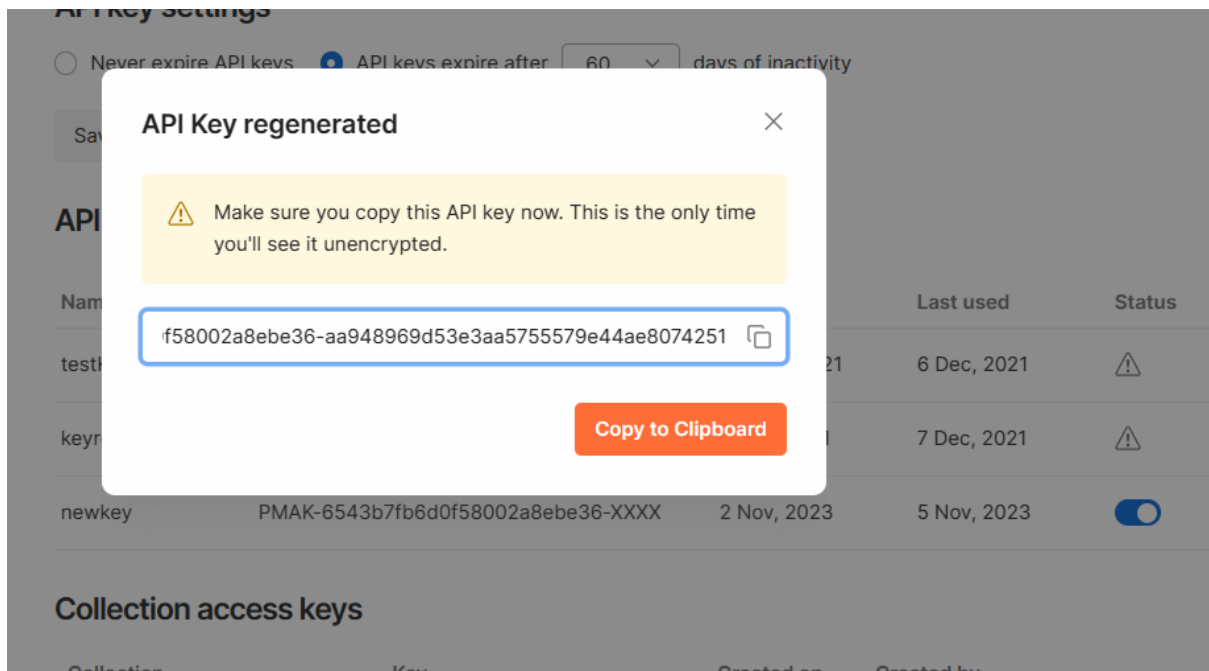
Collection	Key	Created on	Created by
------------	-----	------------	------------

Regenerate API key

This will regenerate your API key and provides you with a new one. Your existing key will be automatically deleted in 6 hours.

☒ Delete the existing key now

Cancel Regenerate API Key



Go to Jenkins pipeline code

Stage 2 -> change sh to bat

And replace \$POSTMAN_API_KEY to the actual postman apikey value



```
12     }
13   }
14
15   stage('Postman CLI Login') {
16     steps {
17       bat 'postman login --with-api-key PMAK-6543b7fb6d0f58002a8ebe36-aa948969d53e3aa5755579e44ae8074251'
18     }
19   }
20
21   stage('Running collection') {
22     steps {
23       bat 'postman collection run "17730345-969b0d23-81ed-4fd2-bab6-69a35846dfe8" -e "17730345-4899fcd6-7cae-46'
24     }
25   }
26 }
27 }
```