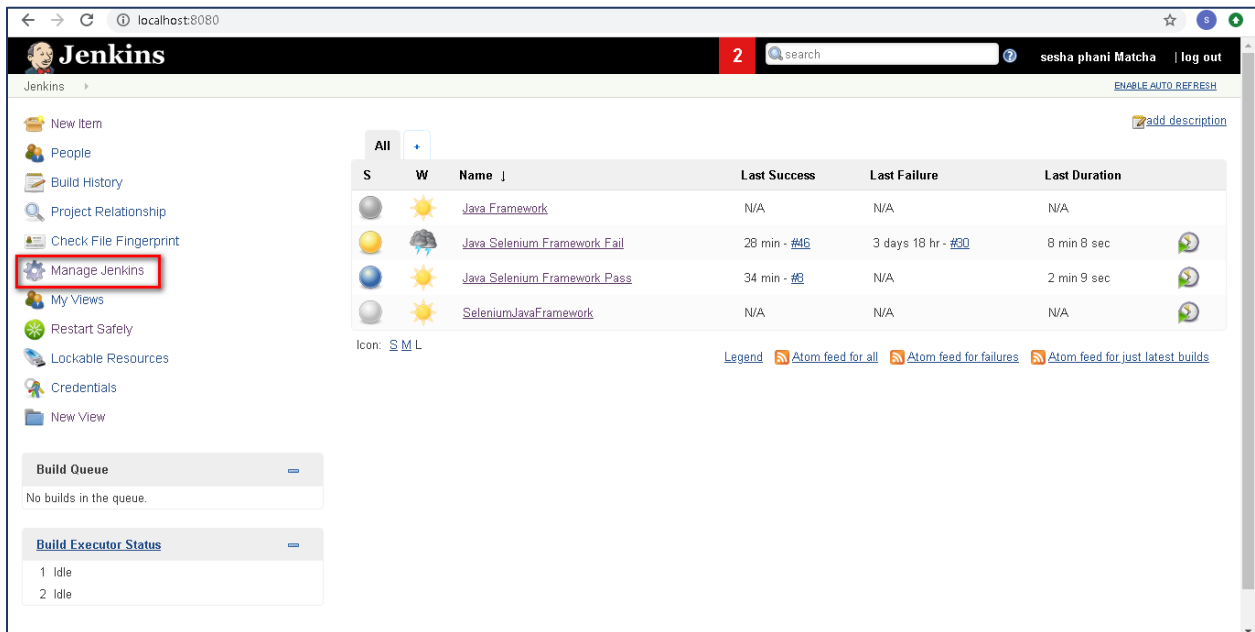


## Jenkins Integration with MS Teams and Slack

1. Download and install Jenkins. Use the following URL to familiarize about the installation process (Only if you are doing this on your local machine. Otherwise, start from step 2, consider the Jenkins hosted over cloud):

<https://www.blazemeter.com/blog/how-to-install-jenkins-on-windows>

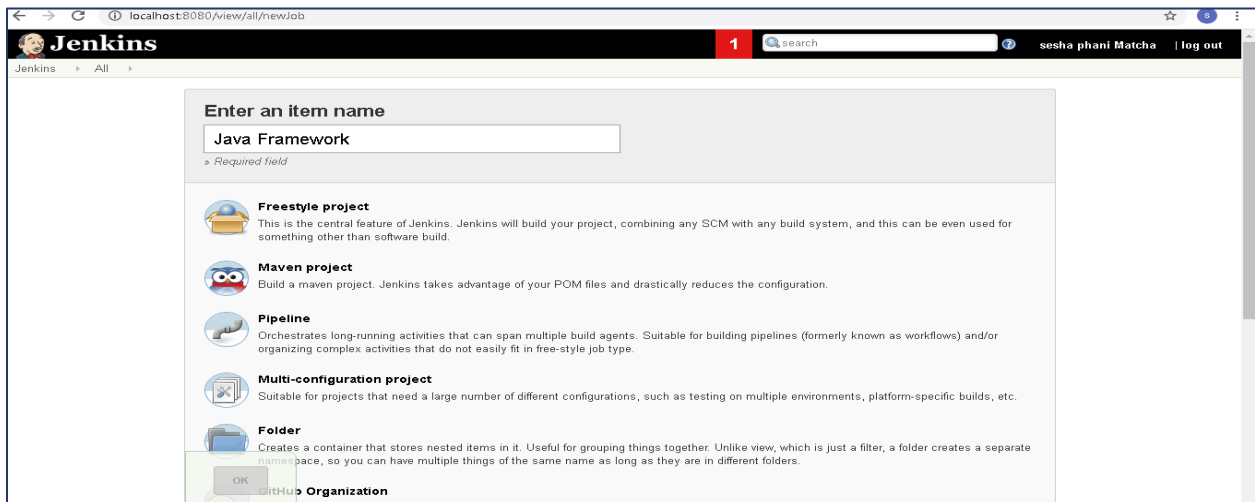
2. In Jenkins, to install any plugin we should go to “Manage Jenkins” tab available left side:



The screenshot shows the Jenkins dashboard at localhost:8080. The left sidebar contains various navigation options, with 'Manage Jenkins' highlighted by a red rectangle. The main content area displays a table of build jobs. The table has columns for 'S' (Status), 'W' (Weather icon), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. The jobs listed are 'Java Framework', 'Java Selenium Framework Fail', 'Java Selenium Framework Pass', and 'SeleniumJavaFramework'. Below the table, there are links for 'Icon: S M L' and 'Legend', and Atom feed links for all, failures, and latest builds.

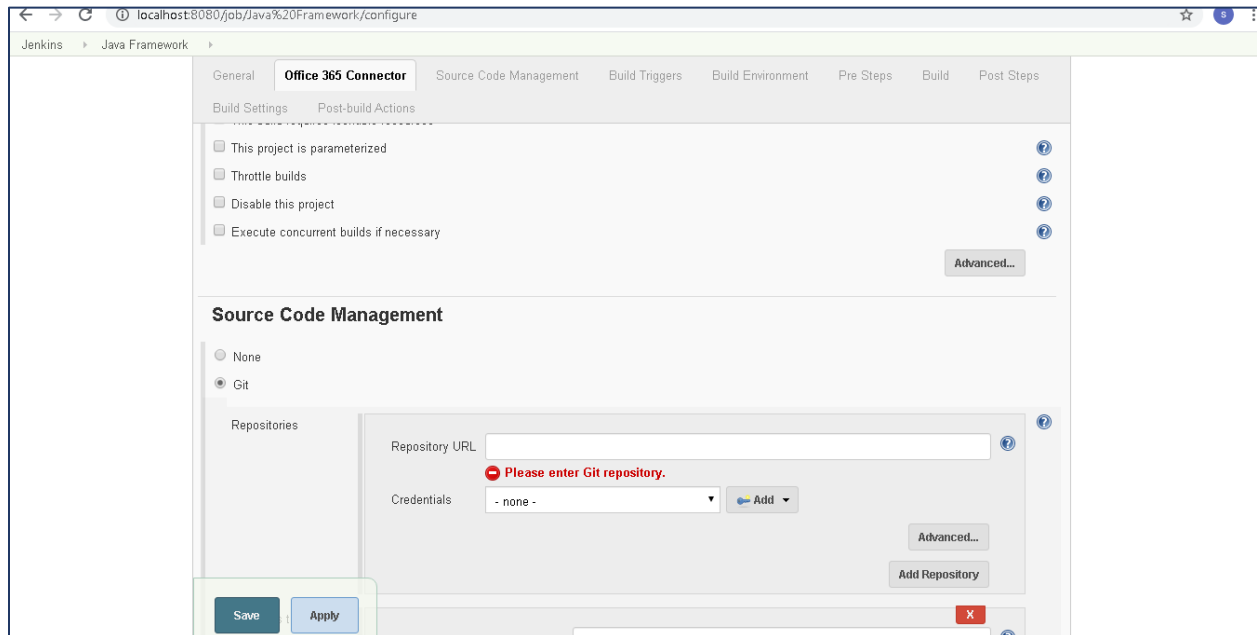
S	W	Name	Last Success	Last Failure	Last Duration
🟢	☀️	<a href="#">Java Framework</a>	N/A	N/A	N/A
🟡	☁️	<a href="#">Java Selenium Framework Fail</a>	28 min - <a href="#">#46</a>	3 days 18 hr - <a href="#">#30</a>	8 min 8 sec
🟢	☀️	<a href="#">Java Selenium Framework Pass</a>	34 min - <a href="#">#6</a>	N/A	2 min 9 sec
🟢	☀️	<a href="#">SeleniumJavaFramework</a>	N/A	N/A	N/A

3. Click on Create New Job
4. Enter the Item Name and Select Maven Project and click on “OK”

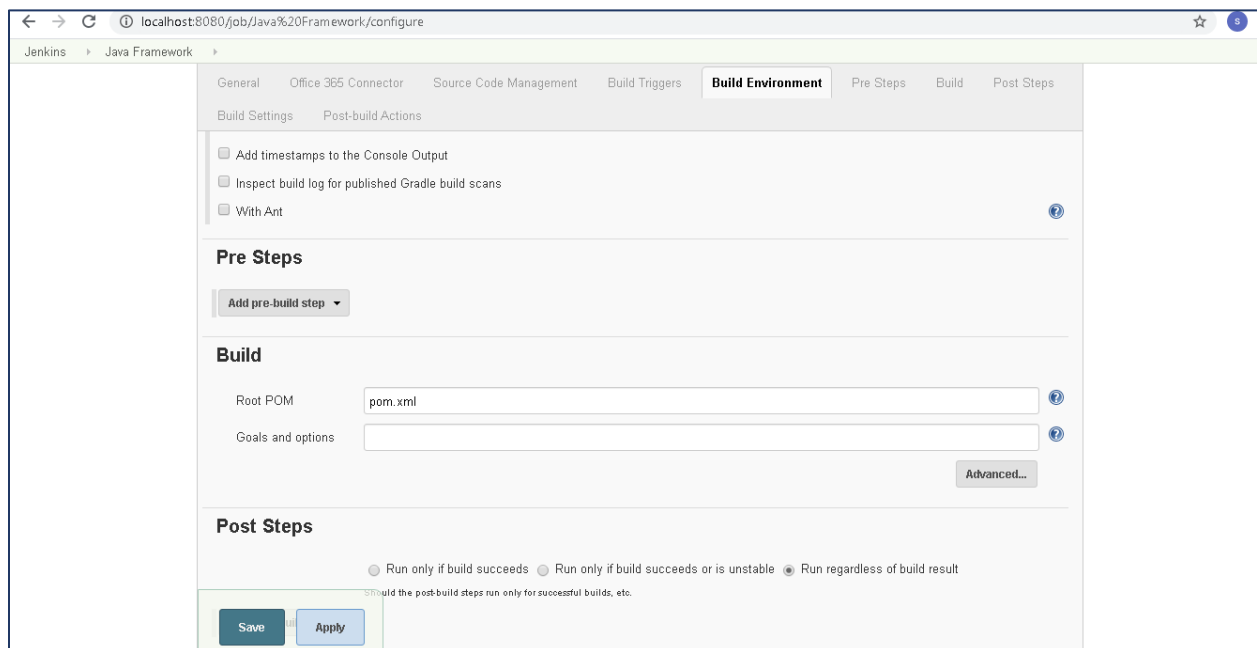


The screenshot shows the Jenkins 'Create New Job' page at localhost:8080/view/all/newJob. The 'Enter an item name' field is filled with 'Java Framework'. Below the field, there are several project type options: 'Freestyle project', 'Maven project', 'Pipeline', 'Multi-configuration project', 'Folder', and 'Organization'. The 'Maven project' option is selected, indicated by a blue circle and a checkmark. The 'Maven project' description states: 'Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.'

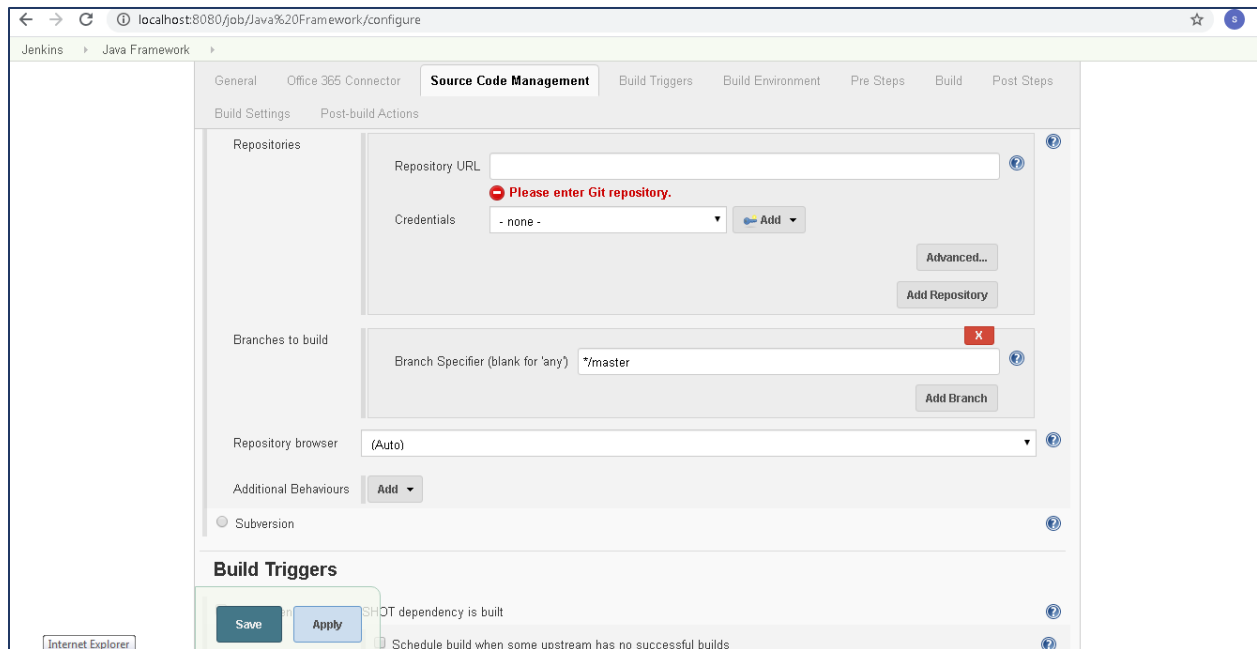
5. Select Git under Source Code Management and enter the Git Repository URL and add Git credentials by clicking on Add and select the added credentials



6. Add the path of "Pom.xml" file in RootPOM field and maven command("clean install") in Goals and options Field under Build Environment

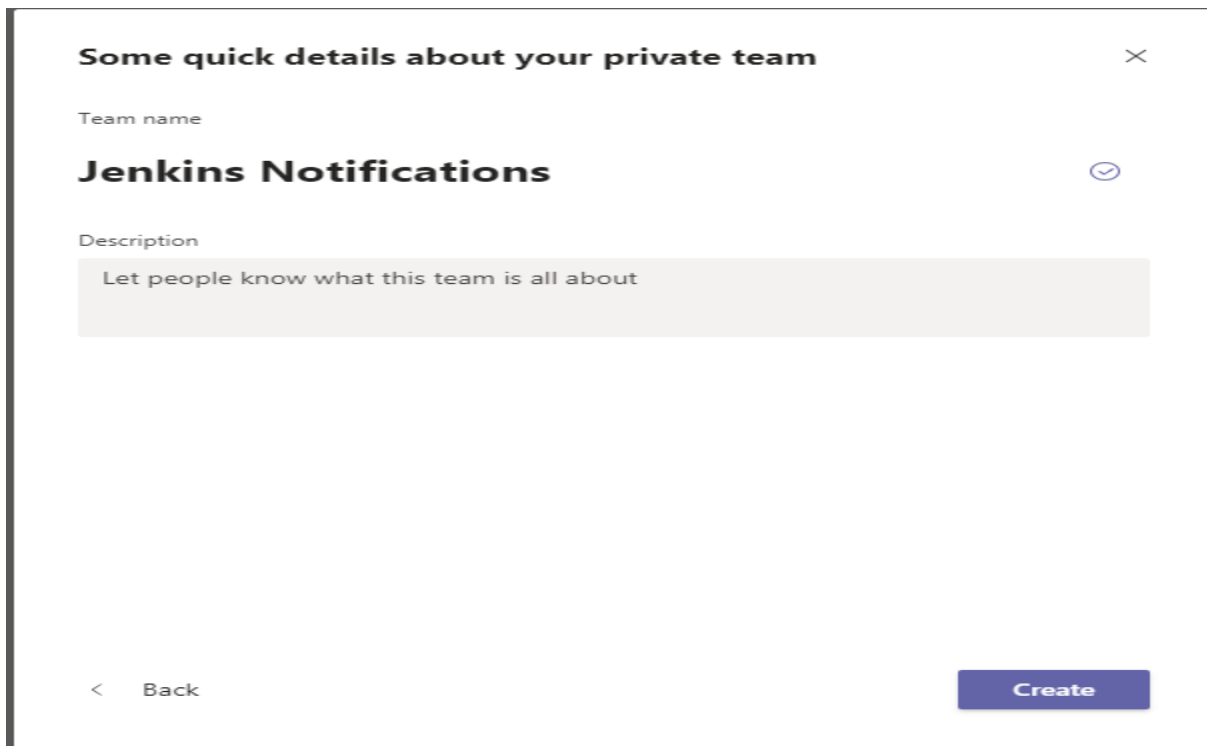


7. Enter the branch name in "Branches to build" field under Source Code Management



## Integration of Jenkins with Teams:

1. Create a new team/group in Microsoft Teams:
2. Name the team/group and click on Create:



3. Add the members to this group to whom Jenkins Notifications should be sent:

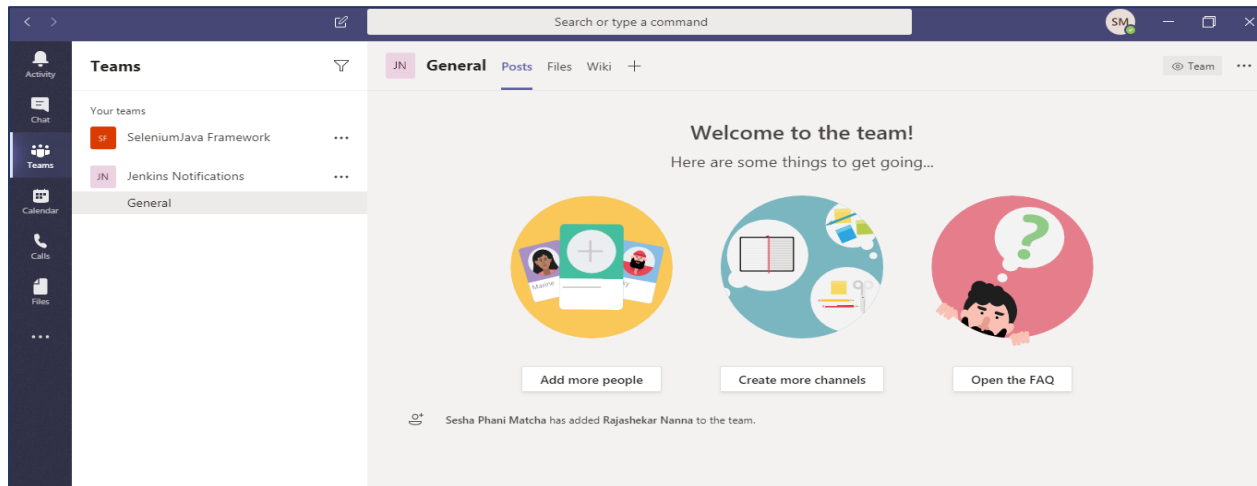
### Add members to Jenkins Notifications

Start typing a name, distribution list, or security group to add to your team. You can also add people outside your organization as guests by typing their email addresses.

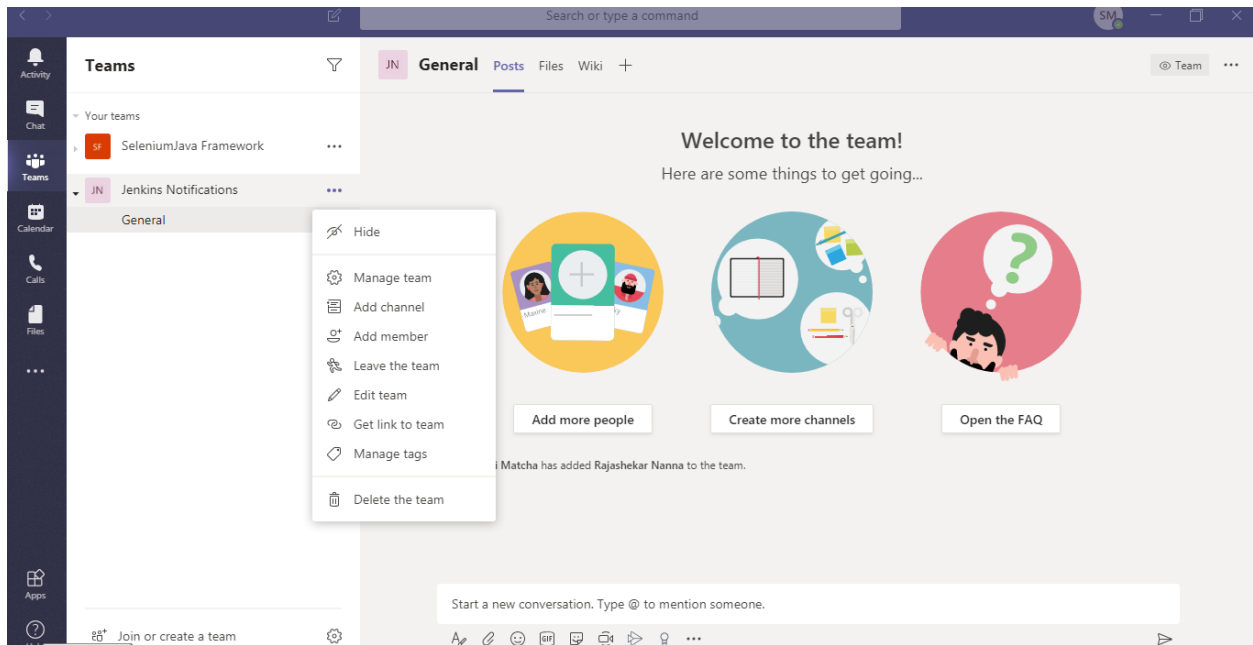
Add

Skip

4. After creating Group with required members, we will see the below page:



5. Select more options(...) beside the created group in teams and click on “Add Channel”:



6. Add Channel Name and click on Add:

### Create a channel for "Jenkins Notifications" team

Channel name

Notifications From Jenkins

Description (optional)

Help others find the right channel by providing a description

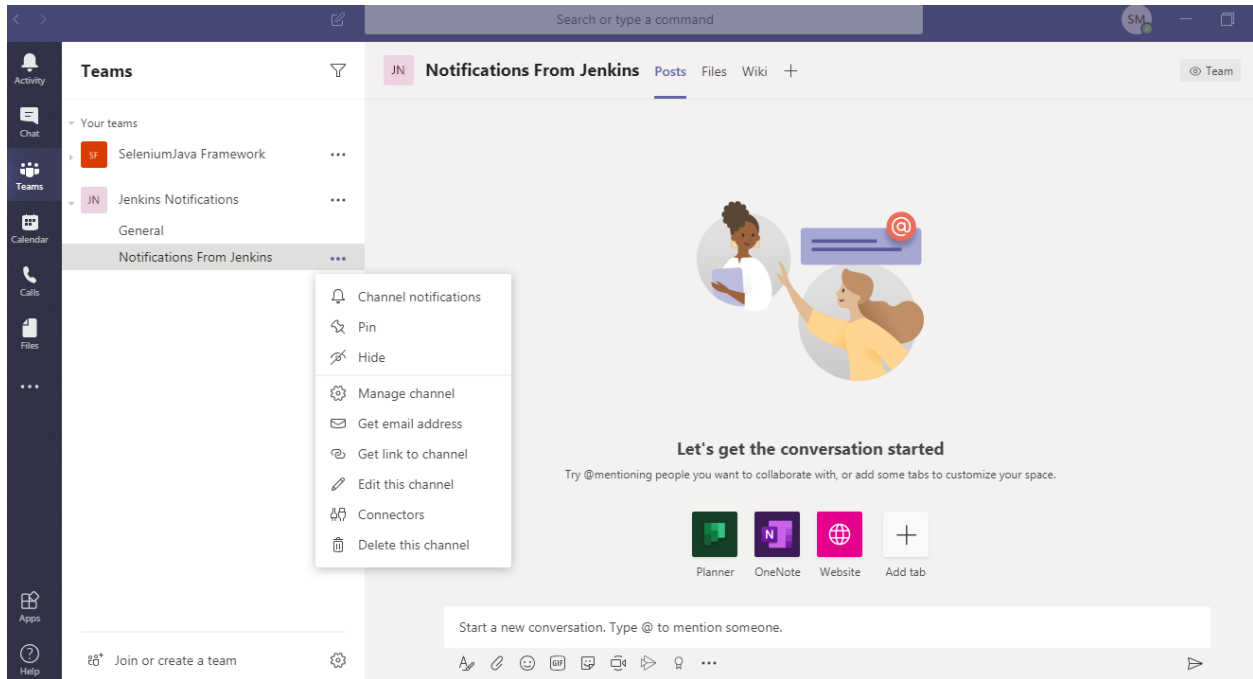
Privacy

Standard - Accessible to everyone on the team

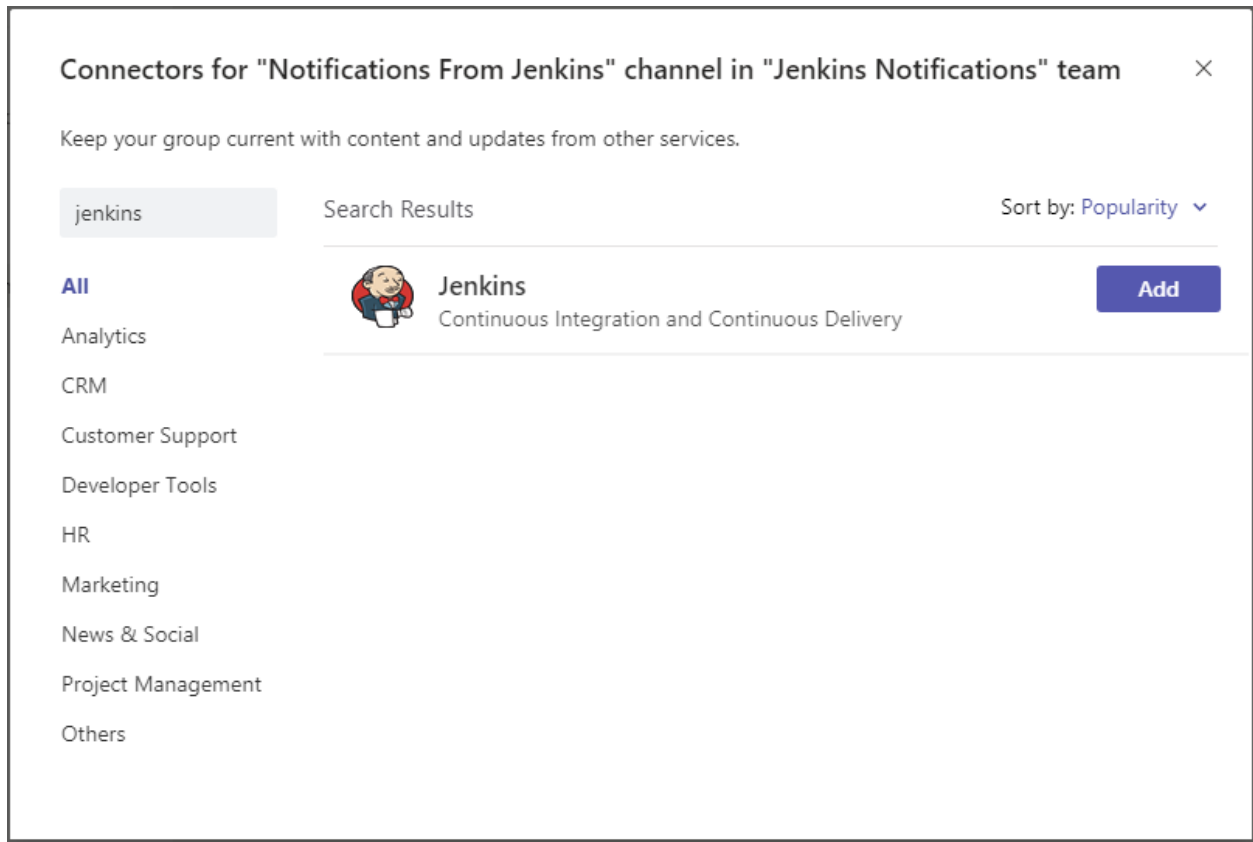
☐ Automatically show this channel in everyone's channel list

CancelAdd


7. By Clicking on more options(...) beside the created channel, click on "Connectors"



## 8. Search for Jenkins and Click on “Add”



9. Click on "Add" as below:

**Jenkins**  
Developer tools

Add


**About**  
Privacy and Permissions

**Continuous Integration and Continuous Delivery**  
The Jenkins connector sends notifications about build-related activities.  
**Notifications**  
Get notifications from the app in a channel  
Created by: [Jenkins](#)  
Version 1.0  
**Privacy and permissions**  
By using Jenkins, you agree to the [privacy policy](#) and [terms of use](#).  
This app will have permission to:

- Send messages and notifications in a channel.

10. Enter the connection Name and click on create:

Connectors for "Notifications From Jenkins" channel in "Jenkins Notifications" team

**Jenkins**[Send feedback](#)

The Jenkins connector sends notifications about build-related activities. To use this connector, you'll need to install Office 365 Connector plugin from Jenkins update center and configure for your project by following a few easy steps. If you don't already have Jenkins installed, you can download it at [Jenkins](#) website.

Fields marked with \* are mandatory

**Name \***  
Enter a name for your Jenkins connection.


CreateCancel

11. Copy the WebHook URL:

### Connectors for "Notifications From Jenkins" channel in "Jenkins Notifications" team

**Name \***  
Enter a name for your Jenkins connection.

**Webhook URL**  
Copy the following URL to save it to the Clipboard. You'll need this URL when you go to the Jenkins website.



Notifications will be sent about the following events in Jenkins:

- Whenever activity occurs in Jenkins.

Instructions

Hide details ^

12. Scroll Down and Click on Done

## Configurations to be done in Jenkins (settings for integrating MS Teams):

1. Jenkins->Manage Jenkins->Manage Plugins:

← → ↻ 🌐 localhost:8080/pluginManager/available

Jenkins

1 🔍 search

🔔 sesha phani Matcha | log out

Jenkins > Plugin Manager

🏠 Back to Dashboard

⚙️ Manage Jenkins

Filter: 🔍

Updates Available Installed Advanced

Install ↓

Name	Version
.NET Development	
<input type="checkbox"/> <a href="#">CCM</a> This plug-in collects the <a href="#">CCM</a> analysis results of the project modules and visualizes the found warnings.	3.2
<input type="checkbox"/> <a href="#">change-assembly-version-plugin</a>	1.10
<input type="checkbox"/> <a href="#">DotNet as Script</a> This plugin ables you to execute your c# code as a jenkins build step.	1.0.2
<input type="checkbox"/> <a href="#">FxCop Runner</a> FxCopCmd.exe support plugin.	1.1
<input type="checkbox"/> <a href="#">MSBuild</a> This plugin makes it possible to build a Visual Studio project (.proj) and solution files (.sln).	1.29
<input type="checkbox"/> <a href="#">MSTest</a> This plugin converts <a href="#">MSTest</a> TRX test reports into JUnit XML reports so it can be integrated with Hudson's JUnit features.	1.0.0
<input type="checkbox"/> <a href="#">MSTestRunner</a> This plugin run MSTest command line tool to execute unit tests for .NET projects.	1.3.0
<input type="checkbox"/> <a href="#">NAnt</a> This plugin is a NAnt Builder for building .NET projects.	1.4.3

Install without restart

Download now and install after restart

Update information obtained: 4 hr 57 min ago

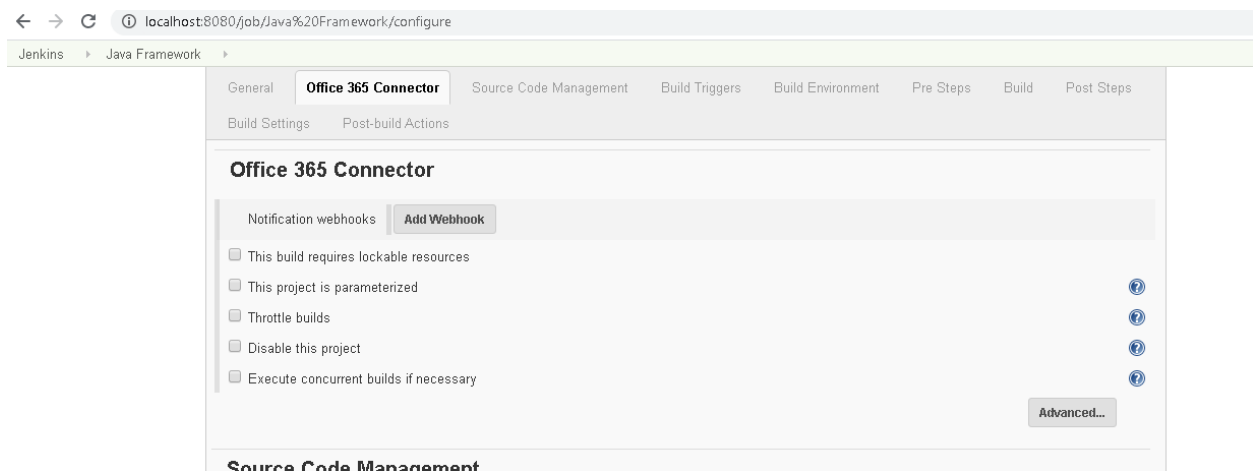
Check now

- Go to Available tab -> Search for "Office 365 Connector" and select that and click on "Install without Restart"
- After installation completed, go to the job created:





4. Click on Configure and select “Office 365 Connector” tab



5. Click on “AddWebHook” and paste the copied webhook (Refer to Step 11 in “Integration of Jenkins with Teams”) in “URL” field. Click on “Advanced” button

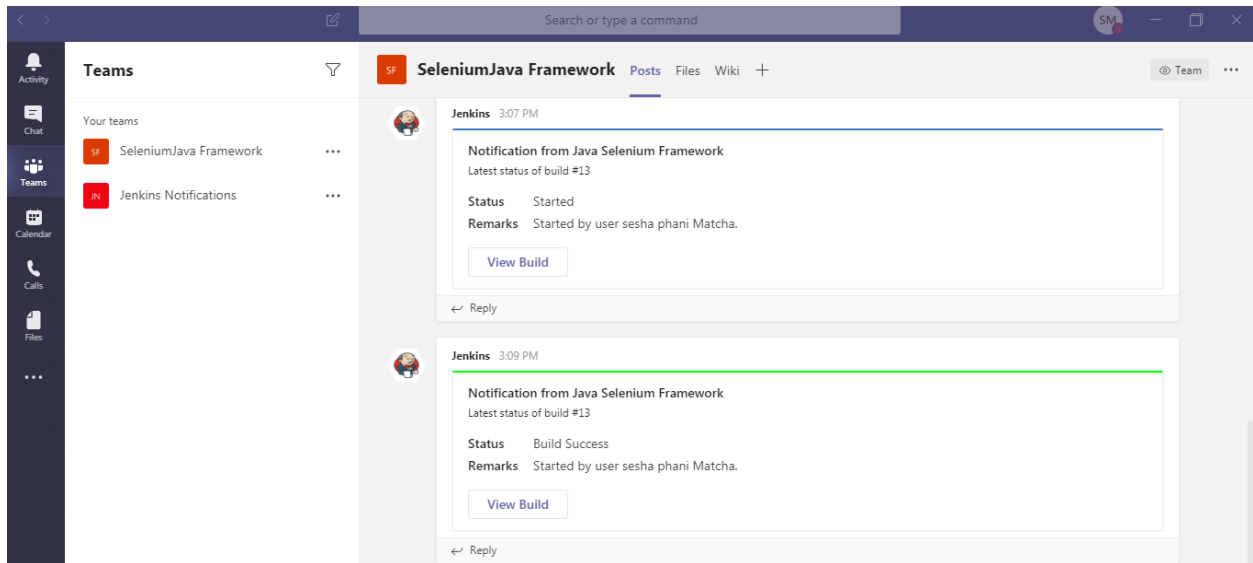
The screenshot shows the Jenkins configuration page for the 'Office 365 Connector'. The 'General' tab is selected. Under 'Notification webhooks', there is a form with two fields: 'URL' and 'Name'. The 'URL' field has a red error message: 'Valid URL or variable reference must be provided'. Below the fields are buttons for 'Add Webhook' and 'Advanced...'. At the bottom, there are several checkboxes for build settings: 'This build requires lockable resources', 'This project is parameterized', 'Throttle builds', 'Disable this project', and 'Execute concurrent builds if necessary'. There are also help icons for each checkbox.

6. Select the required Notifications and click on “Apply” and “Save”:

The screenshot shows the same Jenkins configuration page, but now the 'Build status' section is visible. This section defines for which build statuses the notification is sent. The checkboxes are: 'Notify Build Start' (unchecked), 'Notify Aborted' (unchecked), 'Notify Failure' (checked), 'Notify Not Built' (unchecked), 'Notify Success' (checked), 'Notify Unstable' (checked), 'Notify Back To Normal' (checked), and 'Notify Repeated Failure' (unchecked). Below this is the 'Advanced configuration' section with 'Macros' and 'Fact Definitions' buttons. At the bottom left, there are 'Save' and 'Apply' buttons. The browser's address bar shows 'localhost:8080/job/Java%20Framework/configure'.

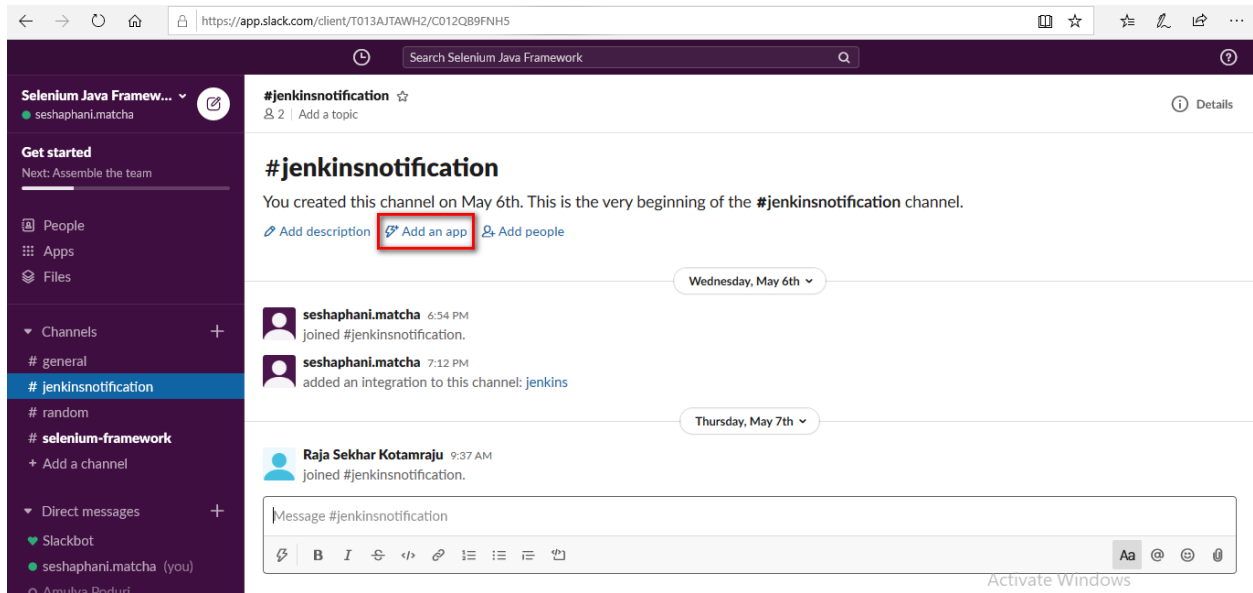
7. Trigger the Build now.

8. We will get Notifications in Teams as below:



## Jenkins Integration with Slack:

1. Create a workspace in Slack
2. Create a channel in slack workspace with required members to send Jenkins Notifications
3. In that channel, click on “Add an App”:



4. Search with “Jenkins” and click on “Install”

Manage apps...

✕  
ESC

## Add apps to #jenkinsnotification

View App Directory

In your workspace

**Jenkins CI**  
An open source continuous integration server. [View](#)

From the App Directory

**Healthchecks.io**  
Cron Monitoring. Get Notified When Your Cron Jobs Fail. [Install](#)

**Slash Webtask**  
Extend Slack with Node.js [Install](#)

**YellowAnt** [Install](#)

Activate Windows  
Go to Settings to activate Windows.

5. In next page Click on “Add To Slack” button

slack App Directory Search App Directory Browse Manage Build Selenium Java ...

[Browse Apps](#)

## Jenkins CI

App Info **Settings** Security & Compliance

**This app was made by Slack.**  
This app was made by a member of the Slack team to help connect Slack with a third-party service; these apps may not be tested, documented, or supported by Slack in the way we support our core offerings, like Slack Enterprise Grid and Slack for Teams. You may provide feedback about these apps at [feedback@slack.com](mailto:feedback@slack.com).

It only uses data Slack already has access to (view our [Privacy Policy](#) to learn more). By enabling and/or using this app, you may be connecting with a service that is not part of Slack.

**Add to Slack**

App help

Terms

Categories:  
Developer Tools

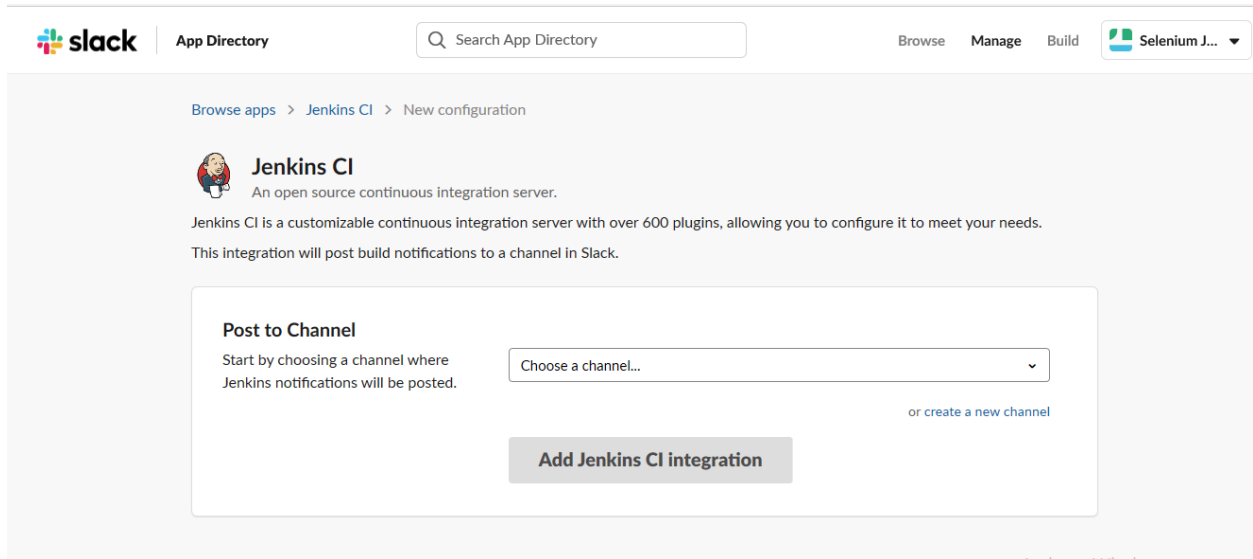
Don't use this app to Slack for

### Configurations

Posts to #selenium-framework as jenkins

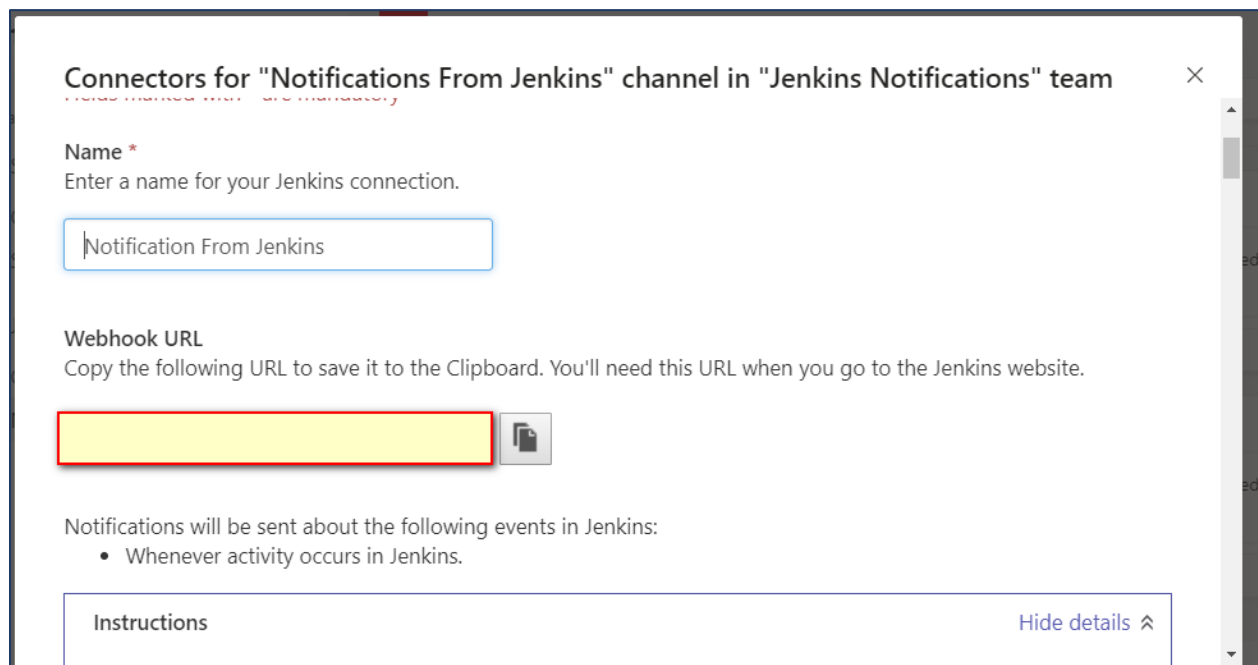
Activate Windows  
Go to Settings to activate Windows.

6. Select the Channel to which Jenkins should be integrated and click on “Add Jenkins CI integration” button:



The screenshot shows the Slack App Directory interface. At the top, there's a search bar and navigation links for 'Browse', 'Manage', and 'Build'. Below the search bar, the breadcrumb trail reads 'Browse apps > Jenkins CI > New configuration'. The main section features the Jenkins CI logo and a description: 'An open source continuous integration server. Jenkins CI is a customizable continuous integration server with over 600 plugins, allowing you to configure it to meet your needs. This integration will post build notifications to a channel in Slack.' A configuration box titled 'Post to Channel' contains a dropdown menu labeled 'Choose a channel...' and a link 'or create a new channel'. At the bottom of this box is a button labeled 'Add Jenkins CI integration'.

7. Scroll down and copy the Token under Integration Settings and click on “Save Settings” button at the end of page:



The screenshot shows a dialog box titled 'Connectors for "Notifications From Jenkins" channel in "Jenkins Notifications" team'. It has a close button (X) in the top right corner. The dialog contains the following fields and text:

- Name \***: Enter a name for your Jenkins connection. The input field contains 'Notification From Jenkins'.
- Webhook URL**: Copy the following URL to save it to the Clipboard. You'll need this URL when you go to the Jenkins website. Below this text is a yellow highlighted box containing a URL, followed by a clipboard icon.
- Notifications will be sent about the following events in Jenkins:**
  - Whenever activity occurs in Jenkins.
- At the bottom, there is an 'Instructions' tab and a 'Hide details ^' link.

## Jenkins Configuration to Integrate with Slack:

1. Jenkins->Manage Jenkins->Manage Plugins:

localhost:8080/pluginManager/available

# Jenkins

1 search sesha phani Matcha log out

Jenkins > Plugin Manager

Back to Dashboard Manage Jenkins

Filter:

Updates Available Installed Advanced

Install	Name	Version
.NET Development		
<input type="checkbox"/>	<a href="#">CCM</a>	3.2
	This plug-in collects the <a href="#">CCM</a> analysis results of the project modules and visualizes the found warnings.	
<input type="checkbox"/>	<a href="#">change-assembly-version-plugin</a>	1.10
<input type="checkbox"/>	<a href="#">DotNet as Script</a>	1.0.2
	This plugin ables you to execute your c# code as a jenkins build step.	
<input type="checkbox"/>	<a href="#">FxCop Runner</a>	1.1
	FxCopCmd.exe support plugin.	
<input type="checkbox"/>	<a href="#">MSBuild</a>	1.29
	This plugin makes it possible to build a Visual Studio project (.proj) and solution files (.sln).	
<input type="checkbox"/>	<a href="#">MSTest</a>	1.0.0
	This plugin converts <a href="#">MSTest</a> TRX test reports into JUnit XML reports so it can be integrated with Hudson's JUnit features.	
<input type="checkbox"/>	<a href="#">MSTestRunner</a>	1.3.0
	This plugin run MSTest command line tool to execute unit tests for .NET projects.	
<input type="checkbox"/>	<a href="#">NAnt</a>	1.4.3
	This plugin is a NAnt Builder for building .NET projects.	

Install without restart Download now and install after restart Update information obtained: 4 hr 57 min ago Check now

- Go to Available tab -> Search for "Slack Notification" and select that and click on "Install without Restart"
- After installation completed, go to the job created:

localhost:8080/job/Java%20Framework/

# Jenkins

1 search

Jenkins > Java Framework

Back to Dashboard

Status

Changes

Workspace

Build Now

Delete Maven project

Configure

Modules

Rename

## Maven project Java Framework

Workspace

Recent Changes

### Permalinks

Build History trend

find

Atom feed for all Atom feed for failures

- Click on Configure and Select the "Post-Build Actions":

Jenkins > Java Framework >

General Office 365 Connector Source Code Management Build Triggers Build Environment Pre Steps Build Post Steps

Build Settings **Post-build Actions**

**Post Steps**

☐ Run only if build succeeds
 ☐ Run only if build succeeds or is unstable
 ☒ Run regardless of build result

Should the post-build steps run only for successful builds, etc.

Add post-build step ▾

**Build Settings**

☐ E-mail Notification

**Post-build Actions**

Add post-build action ▾

Save Apply

5. Select “Slack Notifications” under “Add post-build action” dropdown

Jenkins > Java Framework >

General Office 365 Connector Source Code Management Build Triggers Build Environment Pre Steps Build Post Steps

Build Settings **Post-build Actions**

**Slack Notifications**

☐ Notify Build Start  
☐ Notify Success  
☐ Notify Aborted  
☐ Notify Not Built  
☐ Notify Unstable  
☐ Notify Regression  
☐ Notify Every Failure  
☐ Notify Back To Normal

Advanced...

Add post-build action ▾

Save Apply

6. Select required Notifications checkboxes and click on Advanced and fill the Slack workspace name in “Workspace” field and click on “Add” dropdown and select “Jenkins”:
7. See that “Notification Message Includes” dropdown should contain “nothing about commits” option selected - With this configuration “No Changes.” Message will not appear in slack notification

The screenshot shows the Jenkins configuration page for a job named "Java Selenium Framework Fail". The "Build Settings" tab is active, displaying various notification and workspace options. The "Post-build Actions" tab is also visible. The "Build Settings" section includes checkboxes for "Notify Regression", "Notify Every Failure", "Notify Back To Normal", "Notify First Failure Only", "Notify Repeated Failure Only", "Include Test Summary", and "Include Failed Tests". The "Include Custom Message" checkbox is unchecked. The "Notification message includes" dropdown is set to "nothing about commits". The "What commit information to include into notification message" section is empty. The "Workspace" field is set to "seleniumjavaframework". The "Override url" field is empty. The "Credential" field is set to "Secret text" with an "Add" button. The "Save" and "Apply" buttons are at the bottom left.

localhost:8080/job/Java%20Selenium%20Framework%20Fail/configure

Jenkins > Java Selenium Framework Fail >

General Office 365 Connector Source Code Management Build Triggers Build Environment Pre Steps Build Post Steps

**Build Settings** Post-build Actions

Notify Regression ☒

Notify Every Failure ☒

Notify Back To Normal ☒

Notify First Failure Only ☐

Notify Repeated Failure Only ☐

Include Test Summary ☒

Include Failed Tests ☒

☐ Include Custom Message

Notification message includes nothing about commits

What commit information to include into notification message

Workspace seleniumjavaframework

Override url

Credential Secret text Add

Save Apply

8. Next page is as below:

The screenshot shows the "Jenkins Credentials Provider: Jenkins" page. The "Add Credentials" section is active, displaying a form to add new credentials. The "Domain" dropdown is set to "Global credentials (unrestricted)". The "Kind" dropdown is set to "Username with password". The "Scope" dropdown is set to "Global (Jenkins, nodes, items, all child items, etc)". The "Username", "Password", "ID", and "Description" fields are empty. The "Add" button is at the bottom right.

Jenkins Credentials Provider: Jenkins

**Add Credentials**

Domain Global credentials (unrestricted)

Kind Username with password

Scope Global (Jenkins, nodes, items, all child items, etc)

Username

Password

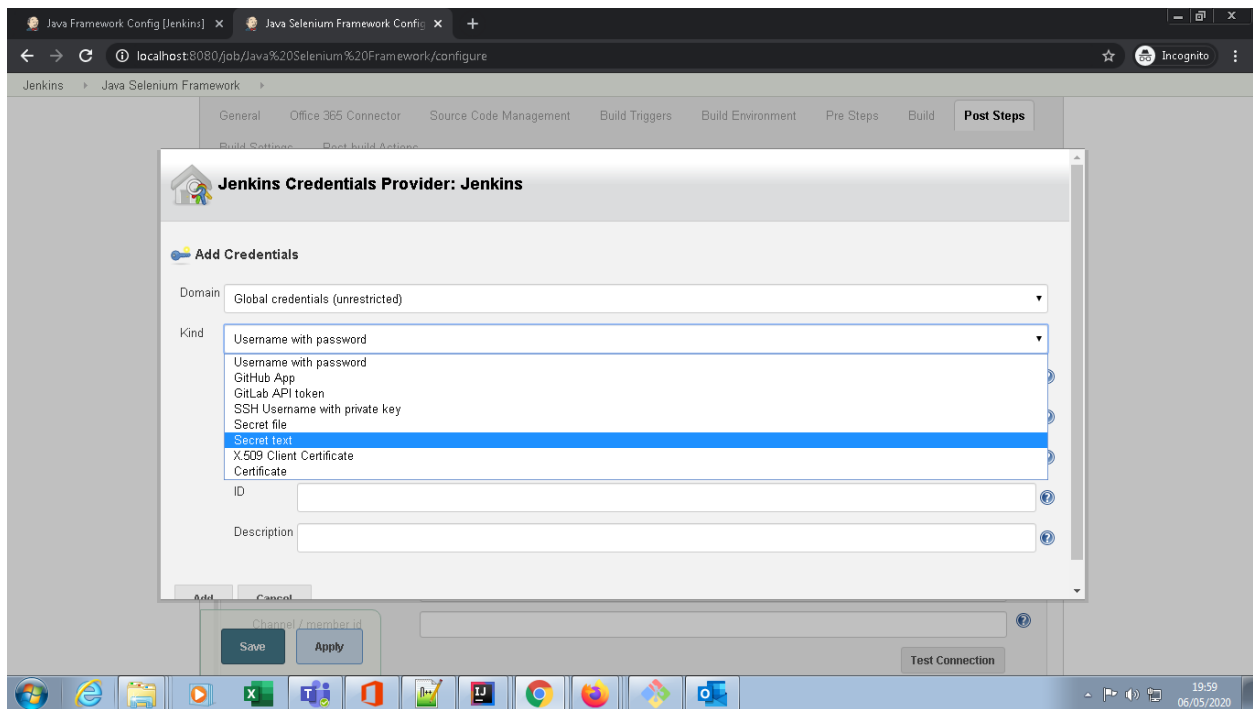
ID

Description

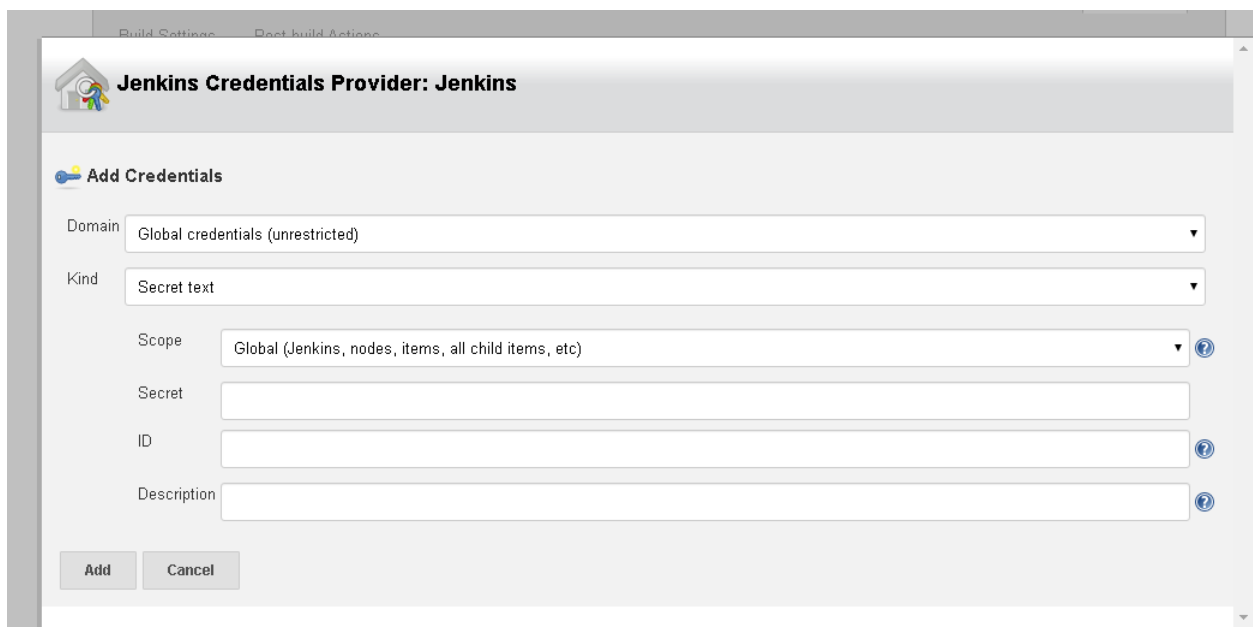
Add

9. From "Kind" Dropdown, select "Secret text" option:





10. Paste the copied Token (Refer Step 7 in “Jenkins Integration with Slack”) from Slack in “Secret” field and click on “OK”:



11. Test the connection by clicking on “Test Connection” and click on “Apply” and “Save”:

Jenkins > Java Selenium Framework >

General Office 365 Connector Source Code Management Build Triggers Build Environment Pre Steps Build **Post Steps**

Build Settings Post-build Actions

Notification message includes **commit list with authors only**

What commit information to include into notification message

Workspace **seleniumjavaframework**

Override url

Credential **Secret text** **Add**

Slack Token (deprecated)

Custom slack app bot user ☐

Icon Emoji

Username

Channel / member id

**Test Connection**

**Add post-build action**

**Save** **Apply**

12. Trigger the build and we can see the build details in Slack:

Slack interface showing build details in the # selenium-framework channel.

**jenkins** **APP** 3:06 PM

Slack/Jenkins plugin: you're all set on <http://localhost:8080/>

Java Selenium Framework - #13 Started by user sesha phani Matcha ([Open](#))

Java Selenium Framework - #13 Success after 2 min 45 sec ([Open](#))

Test Status:  
Passed: 1, Failed: 0, Skipped: 0  
No Changes.

Send your first message to let your teammates know you're here.

**To modify the Jenkins Workspace from One folder to another (Only if you are doing this on your local machine):**

1. Move the Jenkins folder from "C" to "D" folder.
2. Set the Environment Variable for Jenkins:

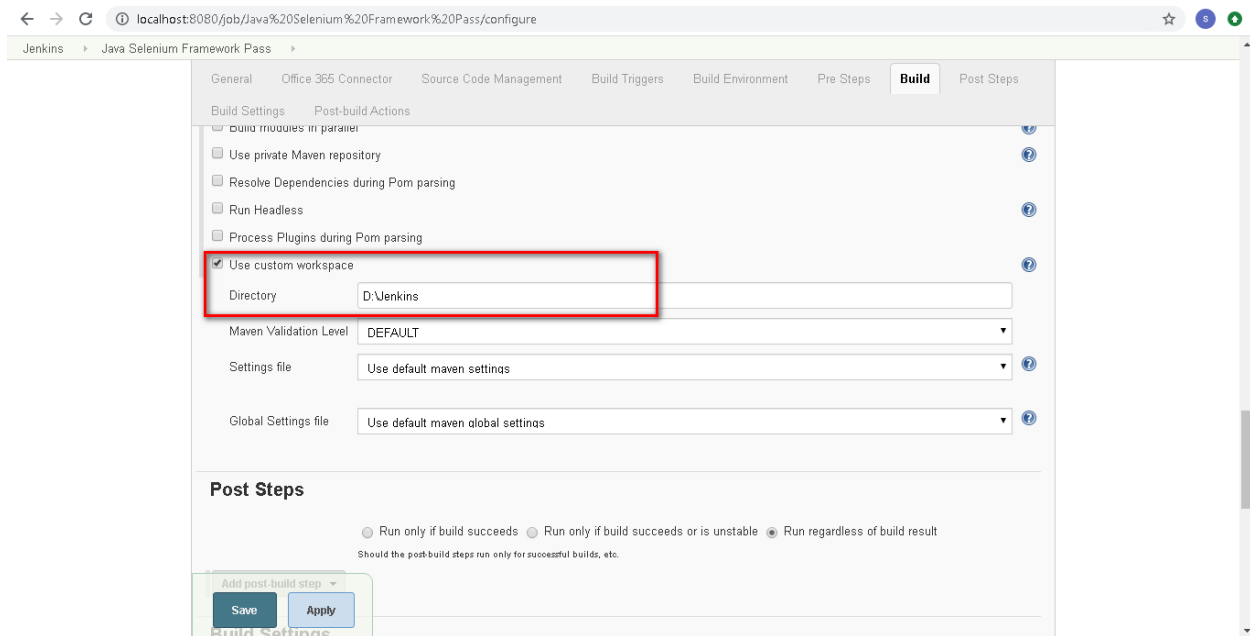
System variables

Variable	Value
JAVA_HOME	C:\Program Files\Java\jdk-9.0.4
JAVA_TOOL_OP...	-agentlib:jvmti
JENKINS_HOME	D:\Jenkins
LEANFT_HOME	C:\Program Files (x86)\Micro Focus\Unif...

**New...** **Edit...** **Delete**

**OK** **Cancel**

3. Go to Jenkins -> Select the Job ->Configure ->Click on Build Tab ->Select Advanced :



4. Check for “Use custom workspace” and fill the “Directory” field with the Jenkins path
5. Trigger the build and we can find the reports in the specified Jenkins Folder