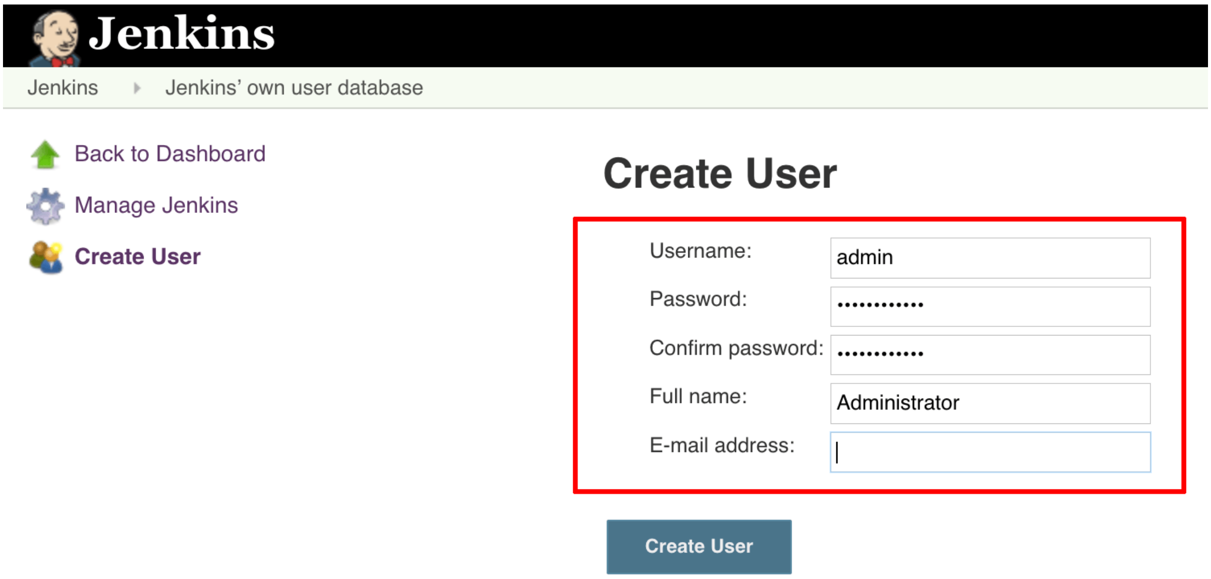
Create new user in Jenkins

- Log in to Jenkins using the existing administrative user account credentials.
- Navigate to the “Manage Jenkins -> Manage Users” page.
- Select the “Create User” option.
- Enter a username, password, name and email address to create a new user. This will become the new administrative user. In the image below, the user is named *admin*.



Username: admin
Password: ....
Confirm password: ....
Full name: Administrator
E-mail address:

- Save the new account.

Manage permissions for a particular user

Follow the steps to do this,

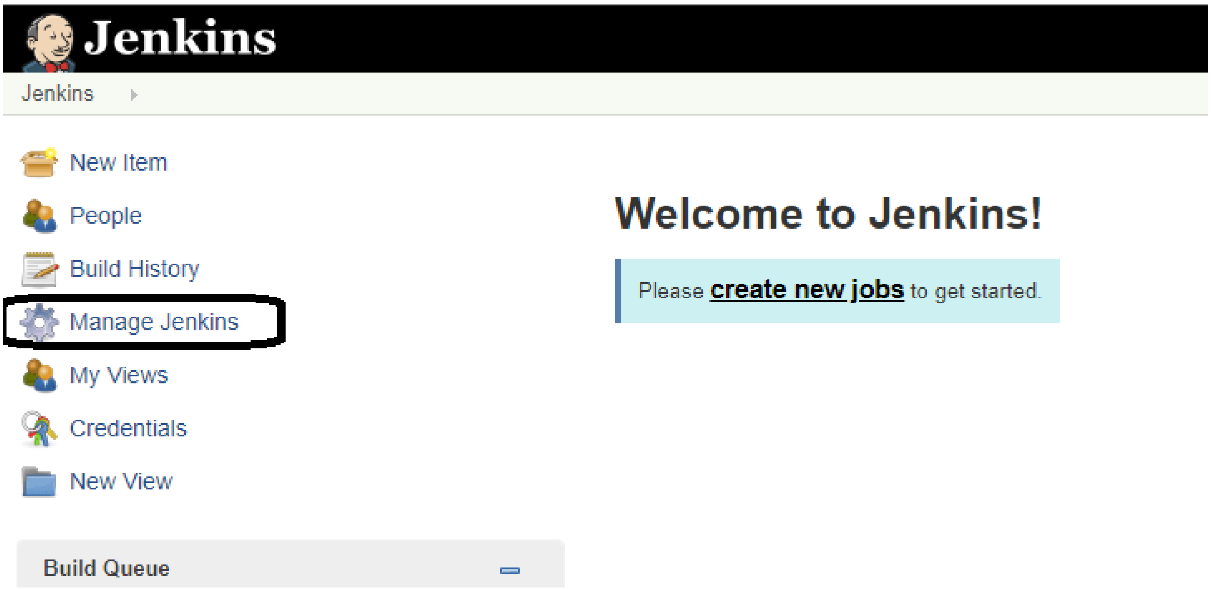
- From the jenkins dashboard,click on Manage Jenkins.
- under Manage jenkins->Configure Global Security->select Enable security.
- Under the Authorization section, select the "Project-based Matrix Authorization Strategy"
- Add the particular user and assign the appropriate permissions.
- And then to assign Job specific permissions :
  - Go to the job (say job1) for which you need to assign permissions.
  - Click Configure->under the general tab->Enable Project-based Security.
  - Add the particular user (say user1) and assign the required permissions.

This is useful when specific jobs shouldn't be accessed by certain users.

Integrating Git under SCM

For this we need to first install git plugin. To install the plugin follow these steps:

Step 1: Open your dashboard.
Click on the **Manage Jenkins** button on your Jenkins dashboard:

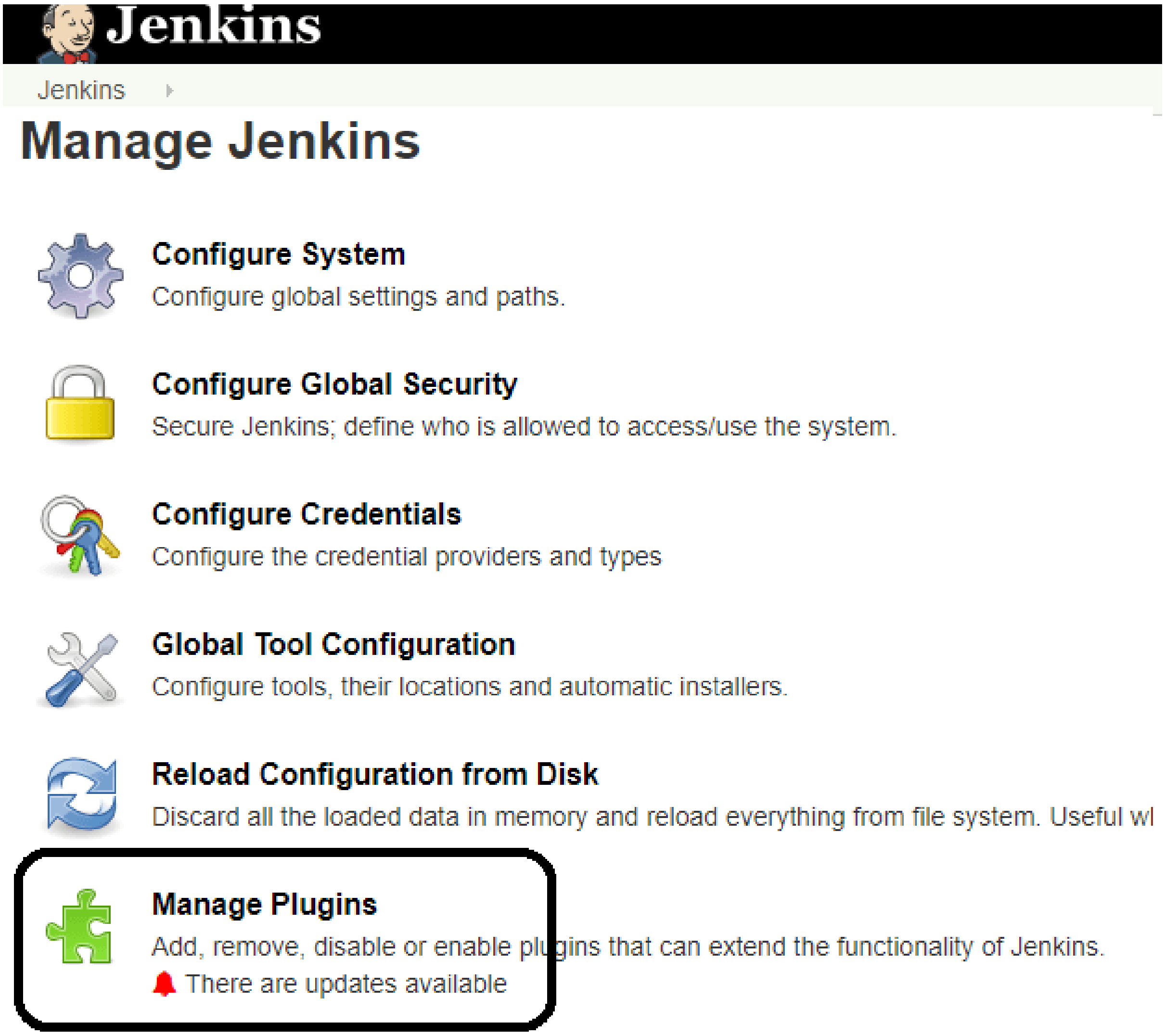


Welcome to Jenkins!

Please **create new jobs** to get started.

Step 2: Find plugins option.

Click on **Manage Plugins**:



Manage Jenkins

Configure System
Configure global settings and paths.

Configure Global Security
Secure Jenkins; define who is allowed to access/use the system.

Configure Credentials
Configure the credential providers and types

Global Tool Configuration
Configure tools, their locations and automatic installers.

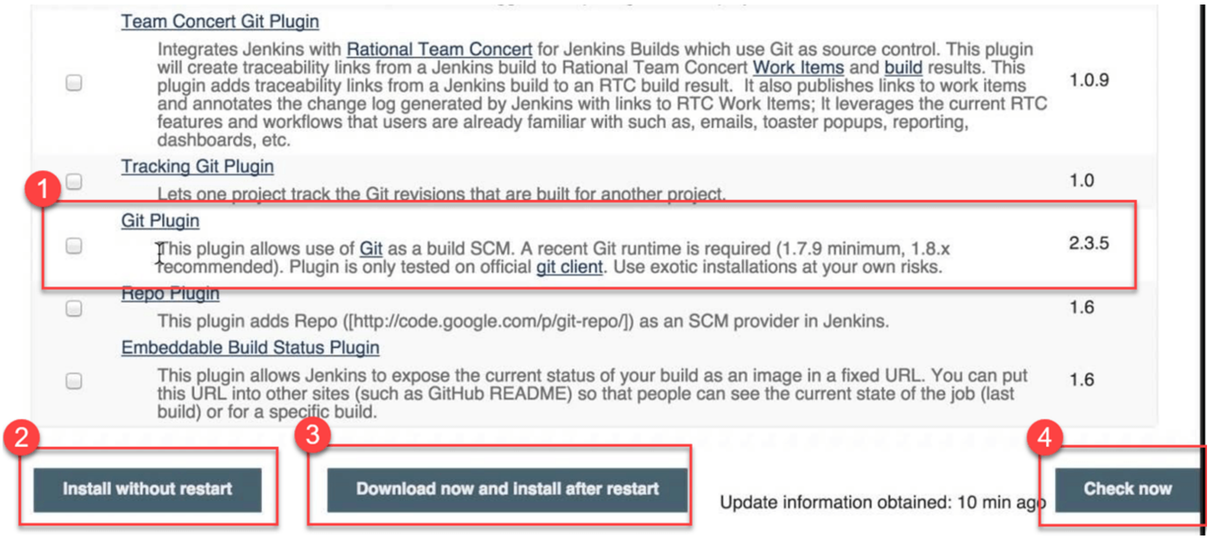
Reload Configuration from Disk
Discard all the loaded data in memory and reload everything from file system. Useful wl

Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
🔔 There are updates available

Step 3: In the Plugins Page,

1. Select the GIT Plugin

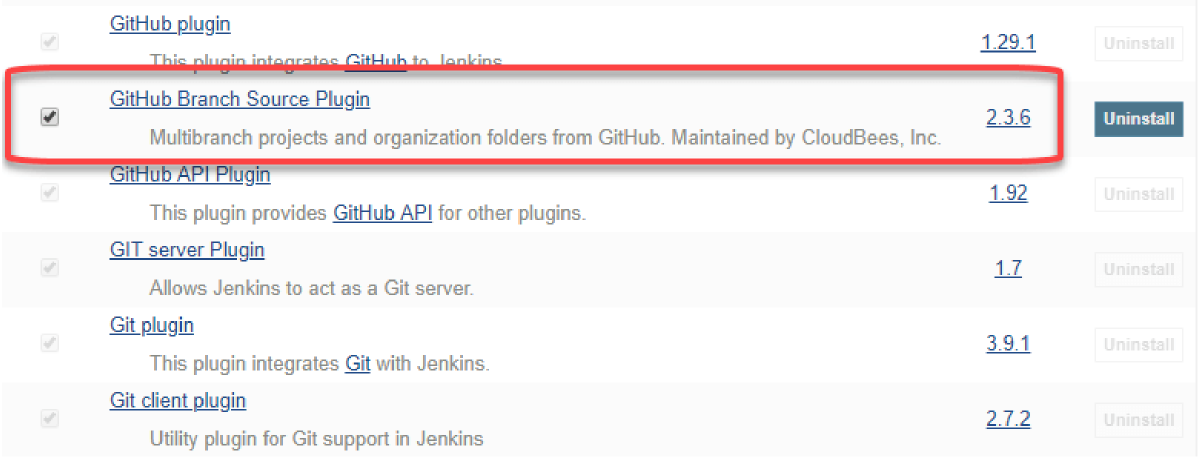
2. Click on **Install without restart.** The plugin will take a few moments to finish downloading depending on your internet connection, and will be installed automatically.
3. You can also select the option **Download now and Install after restart** button. In which plugin is installed after restart
4. You will be shown a "No updates available" message if you already have the Git plugin installed.



□	Team Concert Git Plugin Integrates Jenkins with Rational Team Concert for Jenkins Builds which use Git as source control. This plugin will create traceability links from a Jenkins build to Rational Team Concert Work Items and build results. This plugin adds traceability links from a Jenkins build to an RTC build result. It also publishes links to work items and annotates the change log generated by Jenkins with links to RTC Work Items; It leverages the current RTC features and workflows that users are already familiar with such as, emails, toaster popups, reporting, dashboards, etc.	1.0.9
□	Tracking Git Plugin Lets one project track the Git revisions that are built for another project.	1.0
□	Git Plugin This plugin allows use of Git as a build SCM. A recent Git runtime is required (1.7.9 minimum, 1.8.x recommended). Plugin is only tested on official git client. Use exotic installations at your own risks.	2.3.5
□	Repo Plugin This plugin adds Repo ([http://code.google.com/p/git-repo/]) as an SCM provider in Jenkins.	1.6
□	Embeddable Build Status Plugin This plugin allows Jenkins to expose the current status of your build as an image in a fixed URL. You can put this URL into other sites (such as GitHub README) so that people can see the current state of the job (last build) or for a specific build.	1.6

Update information obtained: 10 min ago

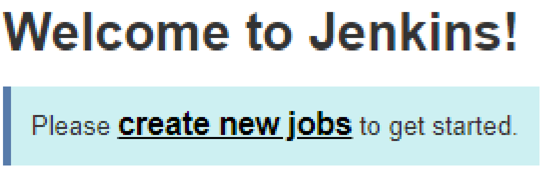
Step 4: Once the plugins have been installed,
Go to **Manage Jenkins** on your Jenkins dashboard. You will see your plugins listed among the rest.



✓	GitHub plugin This plugin integrates GitHub to Jenkins.	1.29.1	Uninstall
☑	GitHub Branch Source Plugin Multibranch projects and organization folders from GitHub. Maintained by CloudBees, Inc.	2.3.6	Uninstall
✓	GitHub API Plugin This plugin provides GitHub API for other plugins.	1.92	Uninstall
✓	GIT server Plugin Allows Jenkins to act as a Git server.	1.7	Uninstall
✓	Git plugin This plugin integrates Git with Jenkins.	3.9.1	Uninstall
✓	Git client plugin Utility plugin for Git support in Jenkins	2.7.2	Uninstall

And then follow these steps.

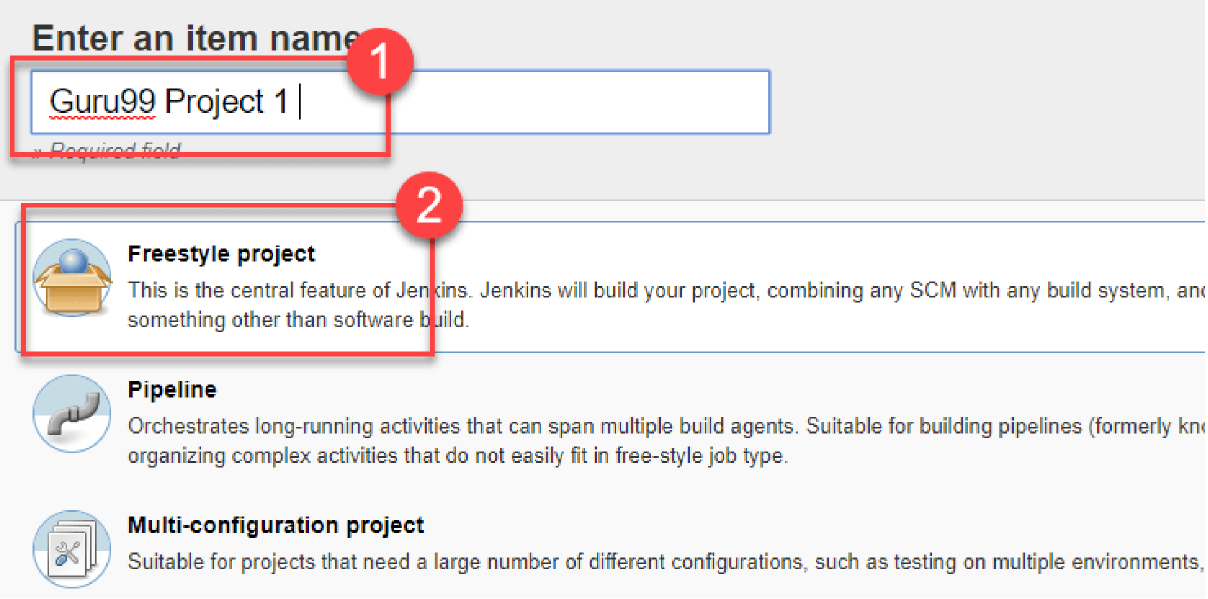
Step 1) Create a new job in Jenkins, open the Jenkins dashboard with your Jenkins URL. For example, http://localhost:8080/
Click on **create new jobs**:



Welcome to Jenkins!

Please **create new jobs** to get started.

Step 2) Enter the item name, select job type and click **OK**. We shall create a Freestyle project as an example.



Enter an item name

Guru99 Project 1

» Required field

Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and something other than software build.

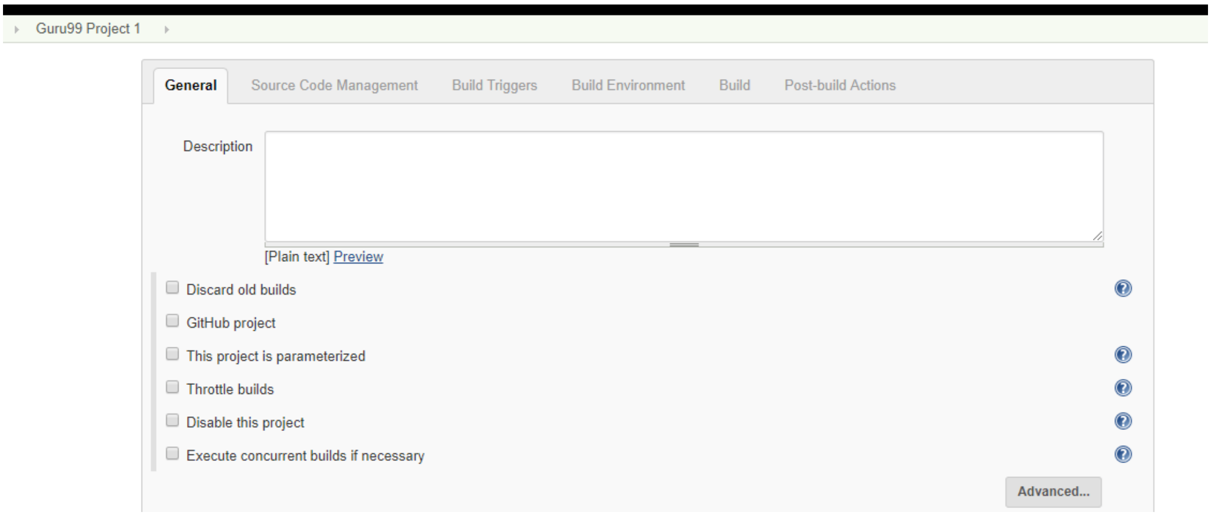
Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly kno organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments,

Step 3) Once you click **OK,** the page will be redirected to its project form. Here you will need to enter the project information:



Description

[Plain text] Preview

□ Discard old builds

□ GitHub project

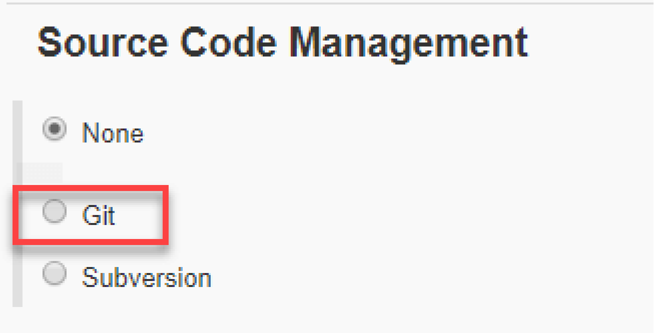
□ This project is parameterized

□ Throttle builds

□ Disable this project

□ Execute concurrent builds if necessary

Step 4) You will see a **Git** option under **Source Code Management** if your Git plugin has been installed in Jenkins:



Source Code Management

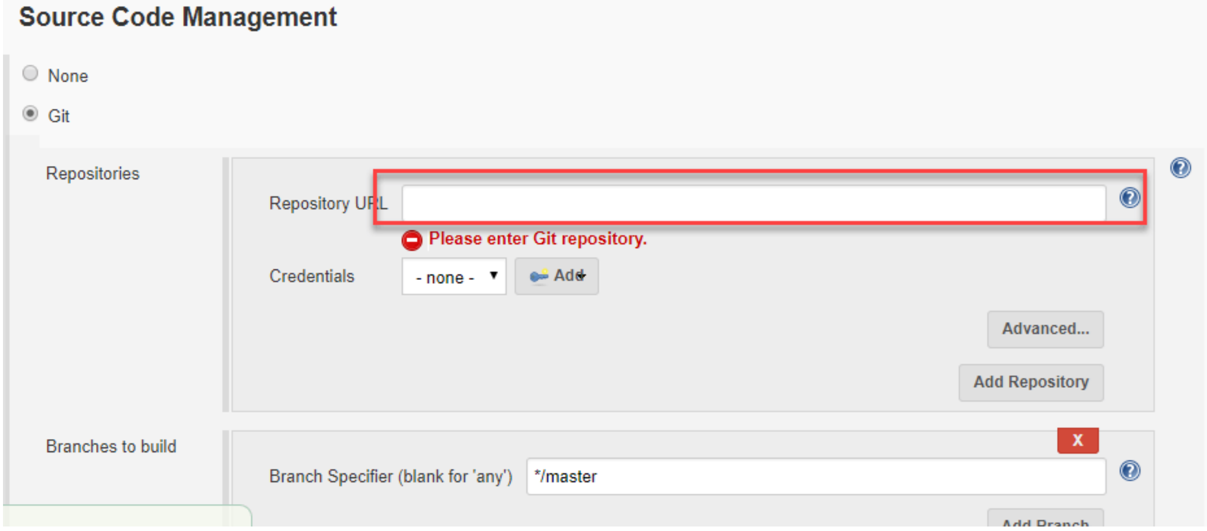
● None

○ Git

○ Subversion

NOTE: If the **Git** option does not appear, try re-installing the plugins, followed by a restart and a re-login into your Jenkins dashboard. You will now be able to see the **Git** option as mentioned above.

Step 5) Enter the Git repository URL to pull the code from GitHub.



Source Code Management

○ None

● Git

Repositories

Repository URL:

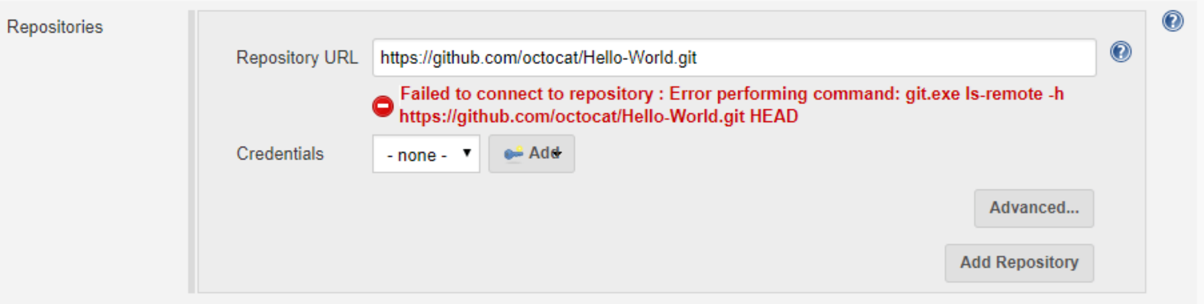
Please enter Git repository.

Credentials: - none -

Branches to build

Branch Specifier (blank for 'any'): */master

Step 6) You might get an error message the first time you enter the repository URL. For example:



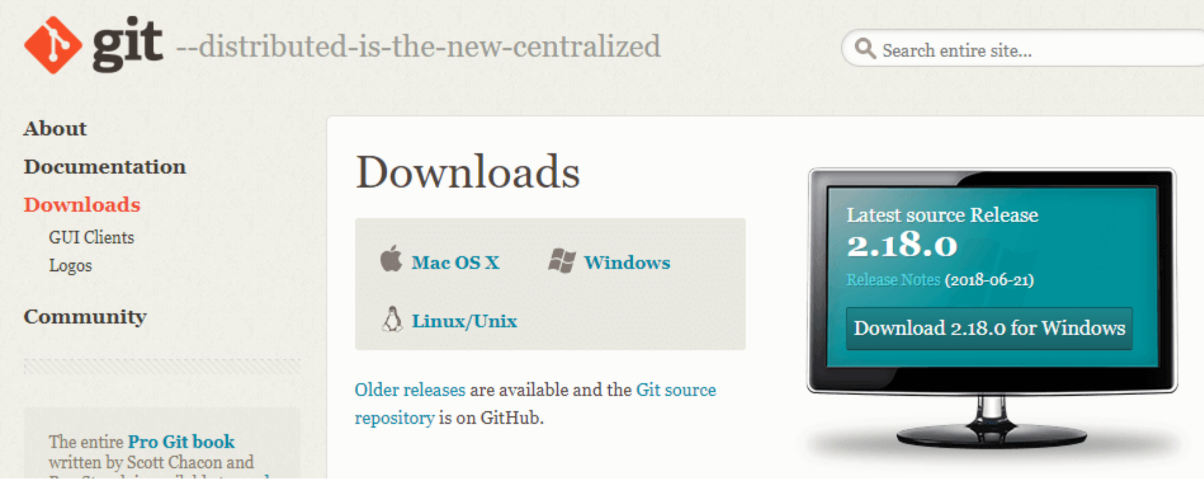
Repositories

Repository URL: https://github.com/octocat/Hello-World.git

Failed to connect to repository : Error performing command: git.exe ls-remote -h https://github.com/octocat/Hello-World.git HEAD

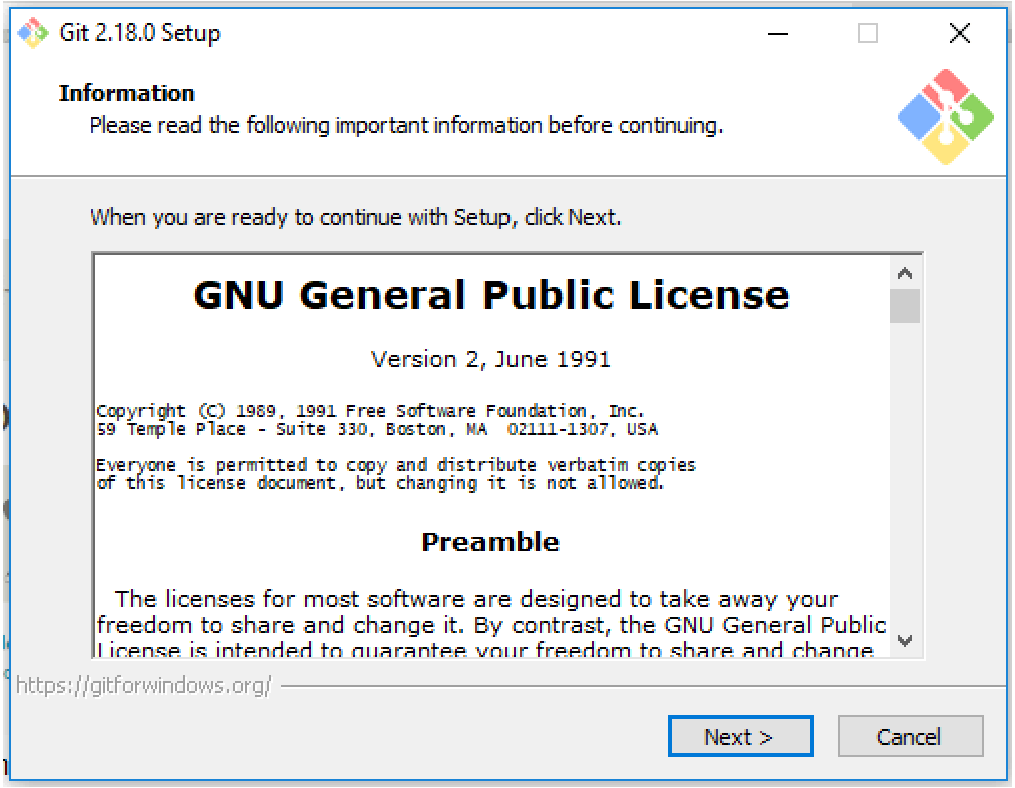
Credentials: - none -

This happens if you do not have Git installed in your local machine. To install Git in your local machine, go to https://git-scm.com/downloads



Older releases are available and the Git source repository is on GitHub.

Download the appropriate Git file for your Operating System, in this case, Windows, and install it onto your local machine running Jenkins. Complete the onscreen instructions to install GIT.



Information

Please read the following important information before continuing.

When you are ready to continue with Setup, click Next.

GNU General Public License

Version 2, June 1991

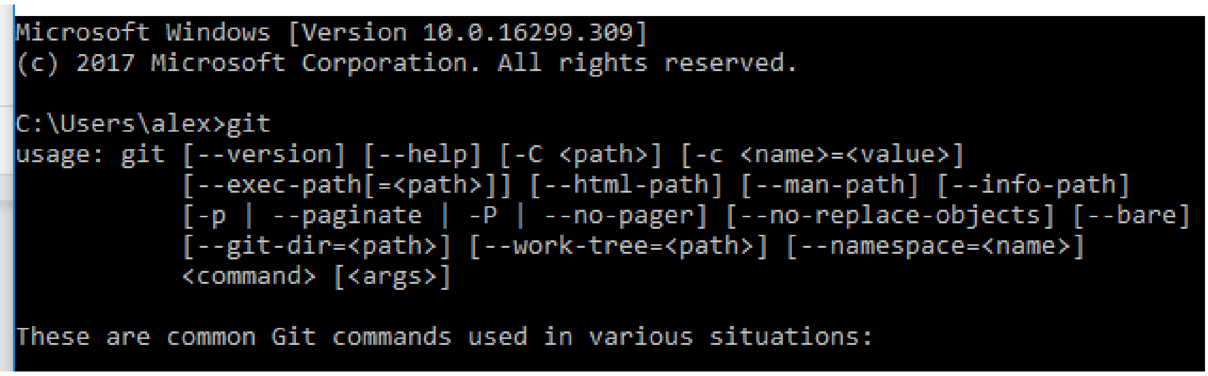
```
Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA  02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
```

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change

https://gitforwindows.org/

Step 7) You can execute Git repositories in your Jenkins once Git has been installed on your machine. To check if it has been successfully installed onto your system, open your **command prompt,** type “Git” and press enter. You should see different options come up for Git:

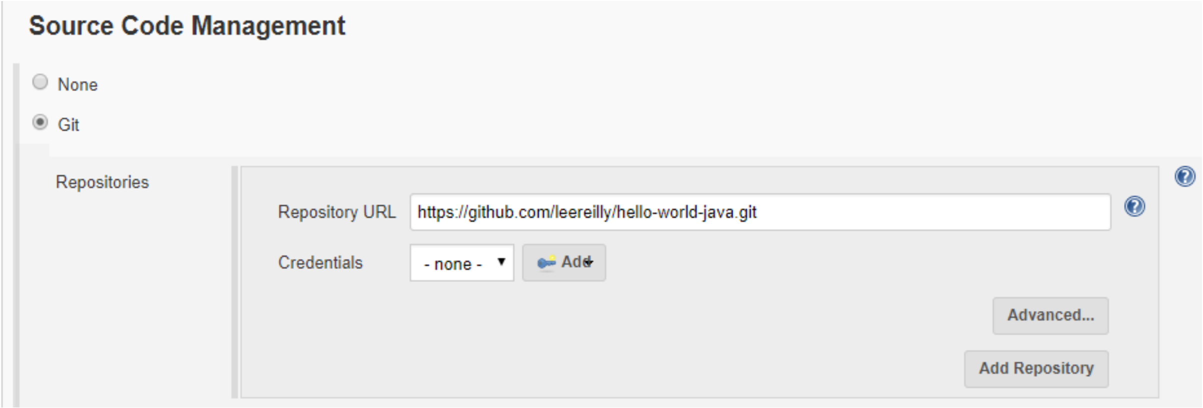


```
Microsoft Windows [Version 10.0.16299.309]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\alex>git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           <command> [<args>]

These are common Git commands used in various situations:
```

This means that Git has been installed in your system.

Note: If you have GIT already installed in your system, just add git.exe path in Global Tool Configuration.

Step 8) Once you have everything in place, try adding the Git URL into Jenkins. You will not see any error messages for Jenkins Git integration:



Source Code Management

○ None

● Git

Repositories

Repository URL: https://github.com/leereilly/hello-world-java.git

Credentials: - none -

Git is now properly configured on your system.

Integrate Email Notification with Jenkins

Introduction

Jenkins is an open-source tool used to perform Continuous Integration (CI) and Continuous Development (CD) by automating build/test/deployments. Using Jenkins you can easily automate the complete process of build to deploy across different environments (Dev/SIT/UAT/Production) by creating different jobs or pipeline.

When you create these jobs there must be some way to get the team and yourself notified about the build/test/deploy status. This is where *Email Notifications* can be used. Jenkins provide plugins to Send Emails.You just need to install and configure the plugin correctly.

This tutorial will help you setting up the email notifications in few simple steps 🙂

You can configure email notifications in your jobs in two ways:

1. Default Email Notifier (Email-Notification)
2. Extended Email Notification (Editable E-mail Notification)

In Extended Email Notifications you can set triggers (e.g. build is unstable or before build), specify email subject, content and recipients.

Prerequisites to Install and Configure Email Notification in Jenkins

This tutorial is based on Jenkins version 2.194 installed on localhost and using http port 8071.

Step 1 - Installing Email Extension Plugin.

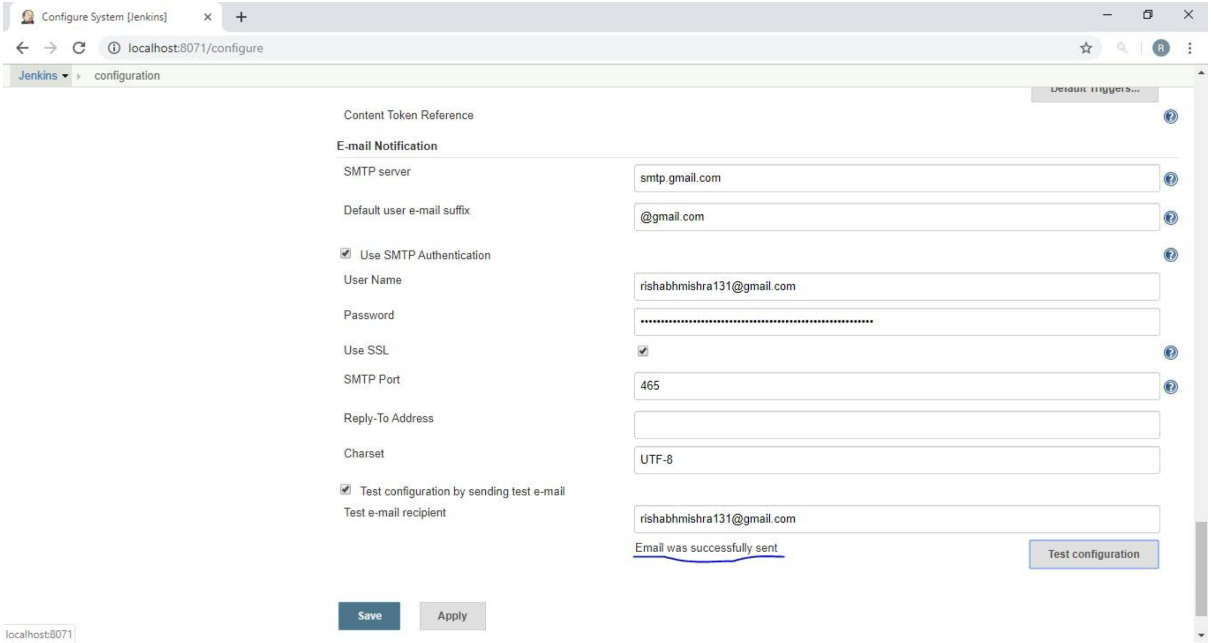
1. Open Jenkins using the following URL: http://localhost:8080/ on any browser (in this tutorial port 8071 is used).
2. Click on *Manage Jenkins*.
3. Click on *Manage Plugins*.
4. Select Email Extension and *Email Extension Template Plugin* and click *Install Without Restart*.

Once The Plugin Is Installed Let's Configure It With SMTP Servers So That Jenkin Emails Can Be Routed Via These SMTP Servers.

Step 2 - Configure Email Notifications

1. Click on Manage Jenkins and then Configure system
2. Scroll below till *E-mail Notification* and click on advanced. Setup up as shown in below screenshot and save it.

You can test configurations by entering recipient email address and clicking on test configuration. If all is good it will show a message - *Email sent successfully*. You may get error while testing configurations, below is possible errors and solution to it.



E-mail Notification

SMTP server: smtp.gmail.com

Default user e-mail suffix: @gmail.com

☑ Use SMTP Authentication

User Name: rishabhmishra131@gmail.com

Password: ••

Use SSL: ☑

SMTP Port: 465

Reply-To Address:

Charset: UTF-8

☑ Test configuration by sending test e-mail

Test e-mail recipient: rishabhmishra131@gmail.com

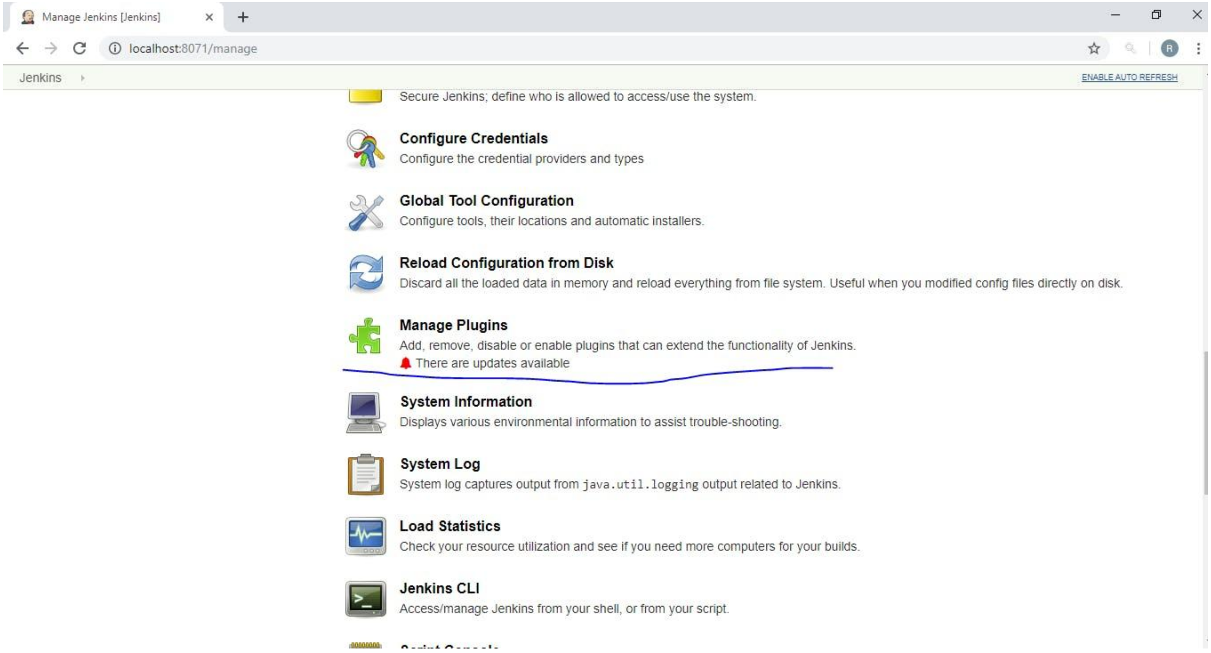
Email was successfully sent

In step 4 you will see the steps to configure email content and set triggers using *Extended Email Notification*.

Step 3 — Configuring Email-Notification in our jobs or pipeline

1. Go to your Jenkins home page and click on the created job (Here its gradlerun)

2) Click on *Manage Plugins*



Secure Jenkins; define who is allowed to access/use the system.

Configure Credentials
Configure the credential providers and types

Global Tool Configuration
Configure tools, their locations and automatic installers.

Reload Configuration from Disk
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.

Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
There are updates available

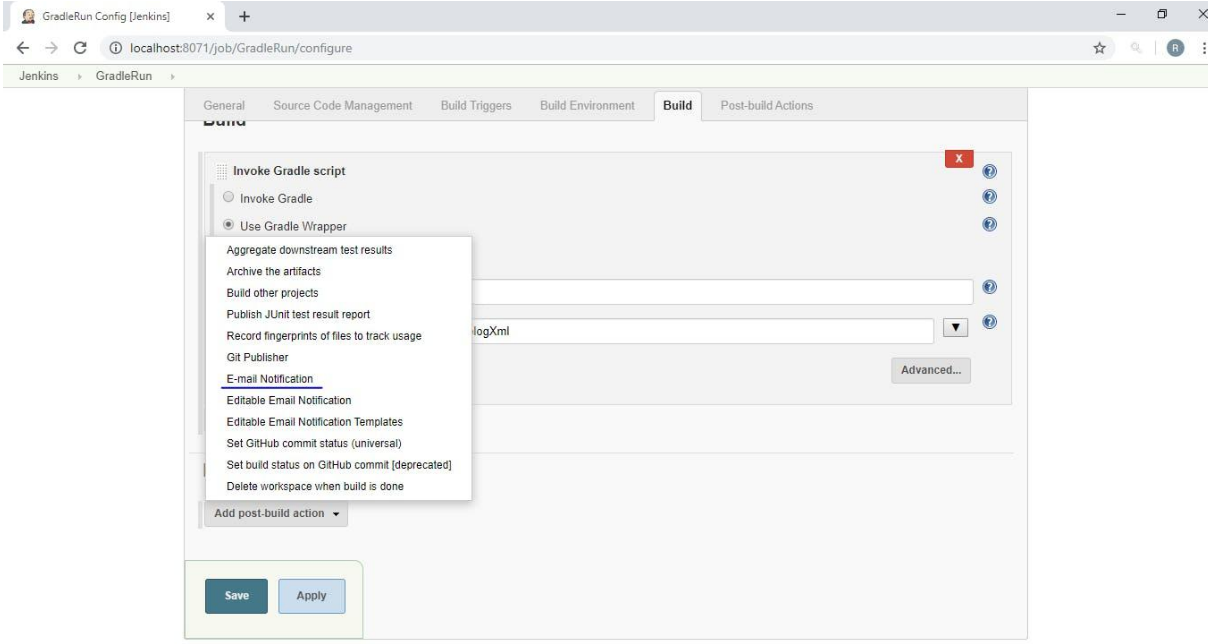
System Information
Displays various environmental information to assist trouble-shooting.

System Log
System log captures output from `java.util.logging` output related to Jenkins.

Load Statistics
Check your resource utilization and see if you need more computers for your builds.

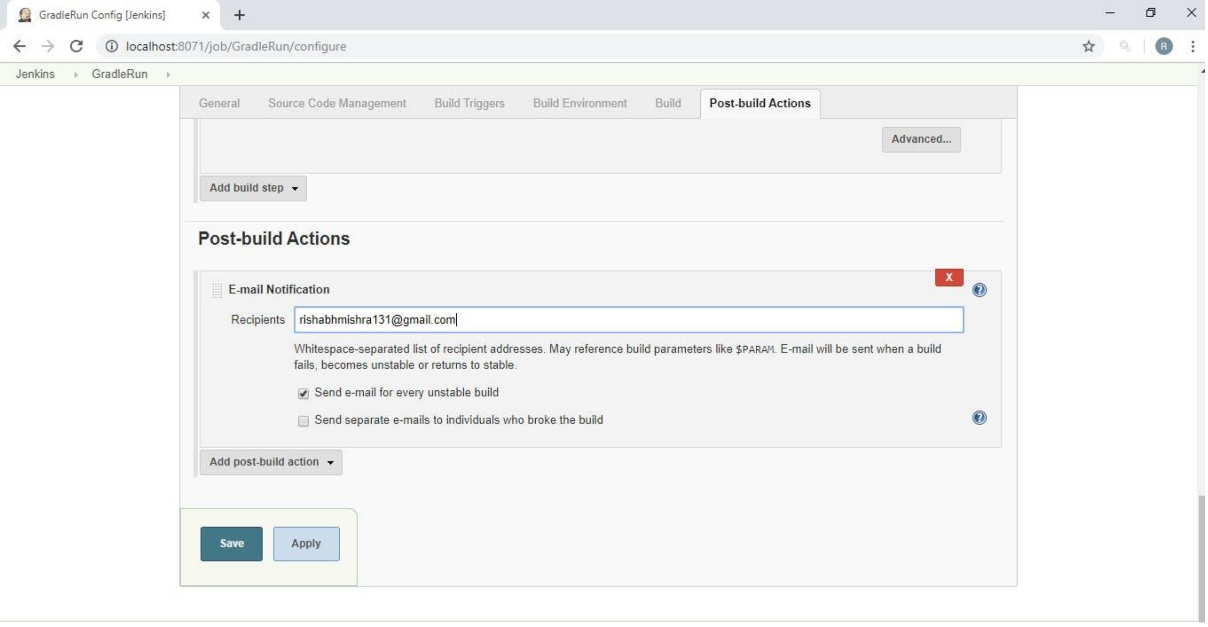
Jenkins CLI
Access/manage Jenkins from your shell, or from your script.

3) Once your job configuration opens, scroll down to add post-build action and select Email Notifications.



GradleRun Config [Jenkins]
localhost:8071/job/GradleRun/configure
Jenkins
GradleRun
General
Source Code Management
Build Triggers
Build Environment
Build
Post-build Actions
Invoke Gradle script
○ Invoke Gradle
● Use Gradle Wrapper
Aggregate downstream test results
Archive the artifacts
Build other projects
Publish JUnit test result report
Record fingerprints of files to track usage
Git Publisher
E-mail Notification
Editable Email Notification
Editable Email Notification Templates
Set GitHub commit status (universal)
Set build status on GitHub commit [deprecated]
Delete workspace when build is done
logXml
Advanced...
Add post-build action
Save
Apply

4) Enter recipients and check Send Email for every unstable build and save it.



Post-build Actions

E-mail Notification

Recipients: rishabhmishra131@gmail.com

Whitespace-separated list of recipient addresses. May reference build parameters like $PARAM. E-mail will be sent when a build fails, becomes unstable or returns to stable.

☑ Send e-mail for every unstable build

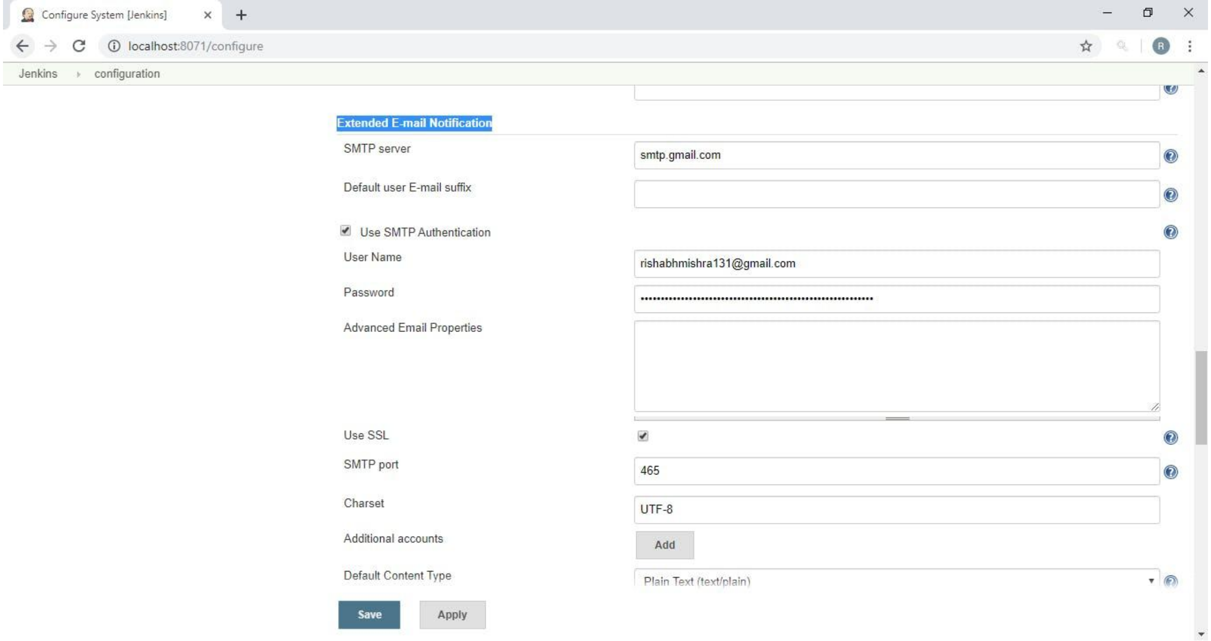
□ Send separate e-mails to individuals who broke the build

Here you go! Run your job and you start receiving emails.

Optional Steps

Step 4 - Configure Extended Email Notification

From step 2, scroll below till Extended E-mail Notification and setup as shown in the screenshot. You can set your own triggers also, every option or setting in self-explanatory, configure it as per your requirement.



Extended E-mail Notification

SMTP server: smtp.gmail.com

Default user E-mail suffix:

☑ Use SMTP Authentication

User Name: rishabhmishra131@gmail.com

Password: ••

Advanced Email Properties:

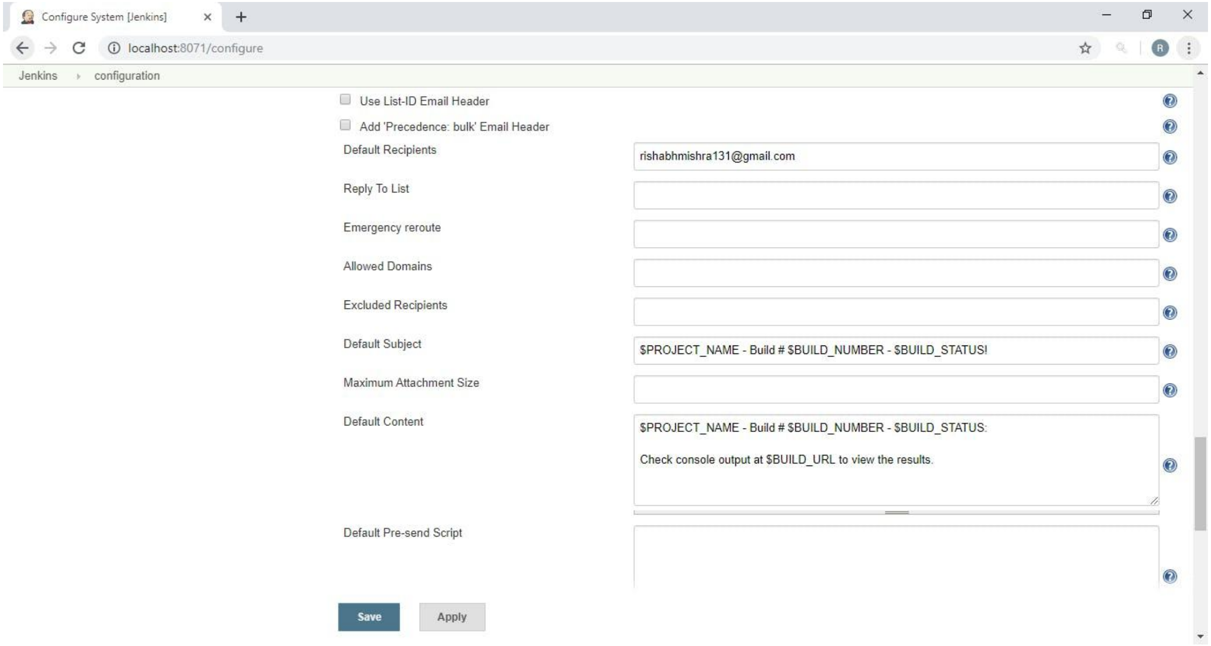
Use SSL: ☑

SMTP port: 465

Charset: UTF-8

Additional accounts

Default Content Type: Plain Text (text/plain)



□ Use List-ID Email Header

□ Add 'Precedence: bulk' Email Header

Default Recipients: rishabhmishra131@gmail.com

Reply To List:

Emergency reroute:

Allowed Domains:

Excluded Recipients:

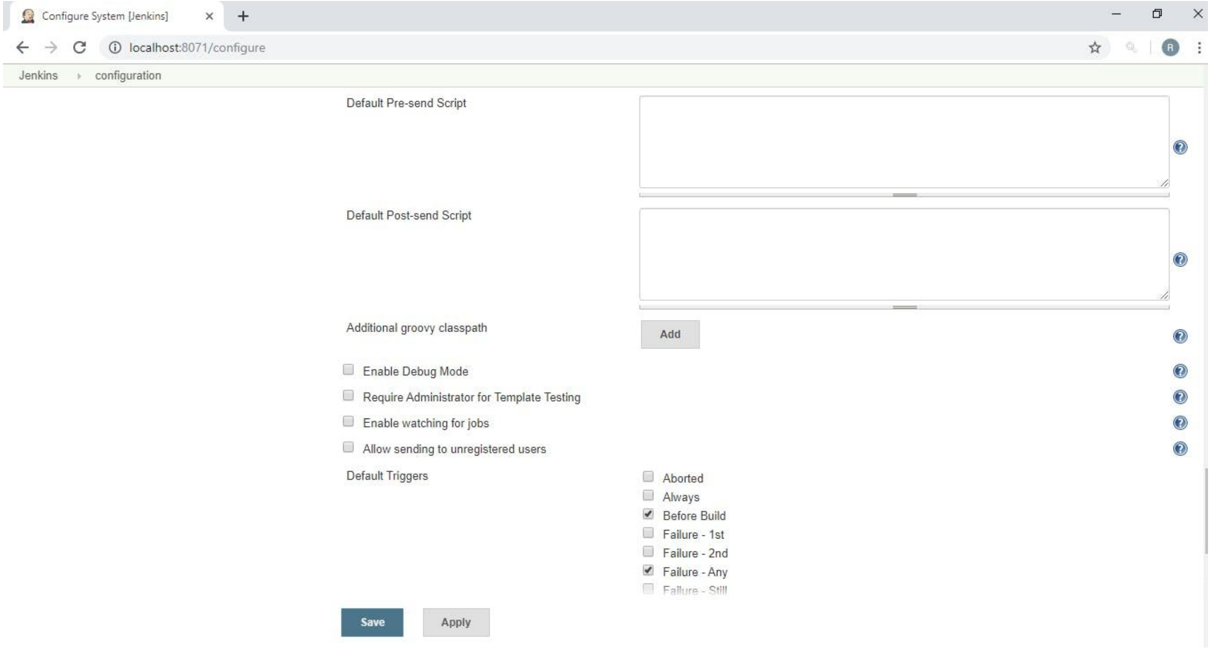
Default Subject: $PROJECT_NAME - Build # $BUILD_NUMBER - $BUILD_STATUS!

Maximum Attachment Size:

Default Content:

```
$PROJECT_NAME - Build # $BUILD_NUMBER - $BUILD_STATUS:

Check console output at $BUILD_URL to view the results.
```

Default Pre-send Script:



Default Pre-send Script:

Default Post-send Script:

Additional groovy classpath: Add

□ Enable Debug Mode

□ Require Administrator for Template Testing

□ Enable watching for jobs

□ Allow sending to unregistered users

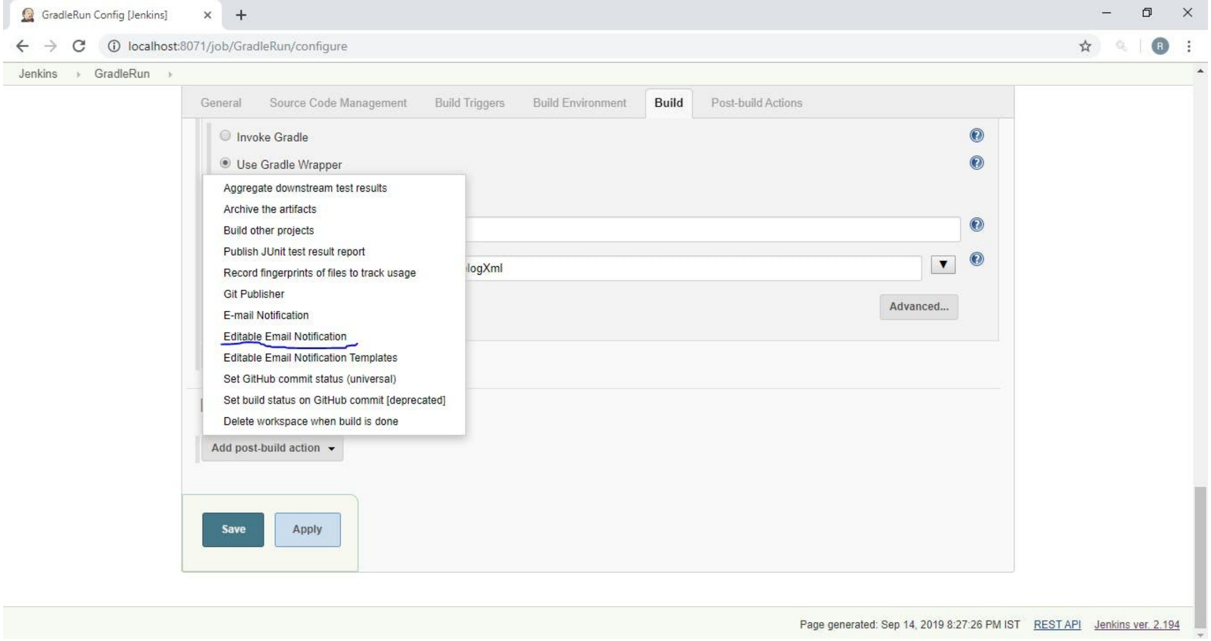
Default Triggers:

- □ Aborted
- □ Always
- ☑ Before Build
- □ Failure - 1st
- □ Failure - 2nd
- ☑ Failure - Any
- □ Failure - Still

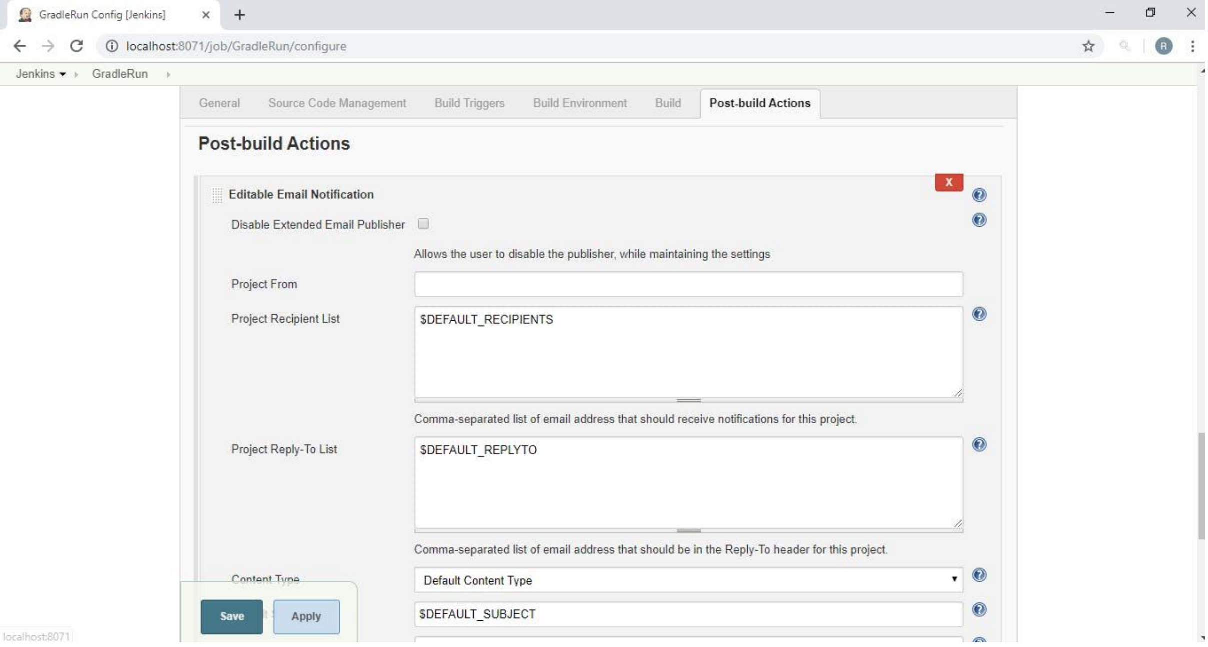
That's it you have successfully configured email settings, Lets set this up in our pipeline or jobs.

Step 5 — Configure Extended Email Notification in Jenkins job

From step 3 in add post-build action and select Editable Email Notifications.


GradleRun Config [Jenkins]
localhost:8071/job/GradleRun/configure
Jenkins › GradleRun ›
General
Source Code Management
Build Triggers
Build Environment
Build
Post-build Actions
○ Invoke Gradle
● Use Gradle Wrapper
Aggregate downstream test results
Archive the artifacts
Build other projects
Publish JUnit test result report
Record fingerprints of files to track usage
Git Publisher
E-mail Notification
Editable Email Notification
Editable Email Notification Templates
Set GitHub commit status (universal)
Set build status on GitHub commit [deprecated]
Delete workspace when build is done
logXml
Advanced...
Add post-build action
Save
Apply
Page generated: Sep 14, 2019 8:27:26 PM IST
REST API
Jenkins ver. 2.194

Once it is added in build action save it.



Post-build Actions

Editable Email Notification

Disable Extended Email Publisher □

Allows the user to disable the publisher, while maintaining the settings

Project From

Project Recipient List: $DEFAULT_RECIPIENTS

Comma-separated list of email address that should receive notifications for this project.

Project Reply-To List: $DEFAULT_REPLYTO

Comma-separated list of email address that should be in the Reply-To header for this project.

Content Type: Default Content Type

$DEFAULT_SUBJECT

Now you will receive custom email for each of your job actions.

Problems that you might face while sending test email from Jenkins

Error 1 - Unable To Connect To SMTP Server

```
java.net.ConnectException: Connection timed out: connect
  at java.net.DualStackPlainSocketImpl.waitForConnect(Native Method)
    Caused: javax.mail.MessagingException: Could not connect to SMTP host: smtp.pepipost.com, port: 25;
```

Solution: Try telnetting SMTP server on different port 587, 2525, 25 etc. if that works change SMTP port in your configuration.

Error 2 - SSL Connection Is Blocked

If using port 587, SSL must be turned on.

```
com.sun.mail.smtp.SMTPSendFailedException: 530 5.7.0 Must issue a STARTTLS command first.
x8sm31507336pfn.106 - gsmtp
```

Solution: Turn on SSL from configurations.

Error 3

If your Jenkins is running on HTTP port you may get below error.

```
javax.net.ssl.SSLException: Unrecognized SSL message, plaintext connection?
    at sun.security.ssl.InputRecord.handleUnknownRecord(Unknown Source)
```

Solution: Either host Jenkins or HTTPS/change port to 465,

Error 4

If you are using Gmail as SMTP server you may below error:

```
javax.mail.AuthenticationFailedException: 535-5.7.8 Username and Password not accepted. Learn more at
535 5.7.8  https://support.google.com/mail/?p=BadCredentials r2sm51378882pfq.60 - gsmtp
```

Solution: If you have entered the wrong password or username correct it. And in Gmail allow less secure apps to log in or install Jenkins to use HTTPS.