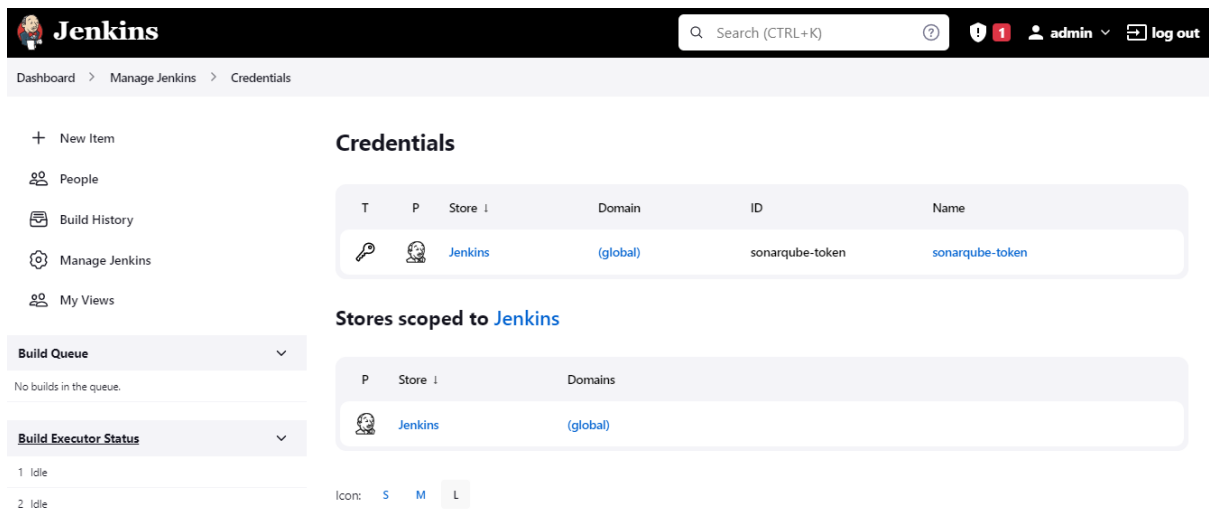


## Procedure:

1: Go to Jenkins dashboard and go to manage Jenkins and Credentials :



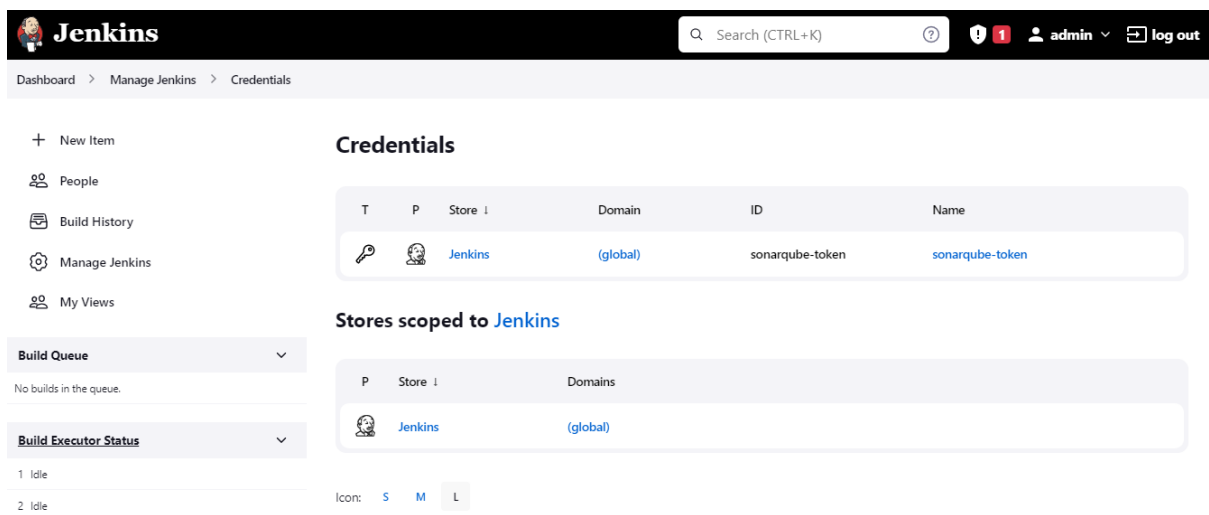
The screenshot shows the Jenkins dashboard with the 'Credentials' page selected. The left sidebar contains navigation links: 'New Item', 'People', 'Build History', 'Manage Jenkins', and 'My Views'. Below these are two expandable sections: 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing two 'Idle' executors). The main content area is titled 'Credentials' and displays a table of credentials. The table has columns: 'T' (Type), 'P' (Provider), 'Store' (Storage), 'Domain', 'ID', and 'Name'. One credential is listed: a 'SecretText' type, 'Jenkins' provider, 'global' domain, with ID 'sonarqube-token' and Name 'sonarqube-token'. Below the table, there is a section 'Stores scoped to Jenkins' showing a table with columns 'P' (Provider) and 'Domains'. It lists 'Jenkins' provider with 'global' domain. At the bottom, there is a 'Icon' section with buttons for 'S', 'M', and 'L'.

T	P	Store	Domain	ID	Name
SecretText	Jenkins	global	sonarqube-token	sonarqube-token	

P	Store	Domains
Jenkins	global	

Icon: S M L

2: After that go to the system :



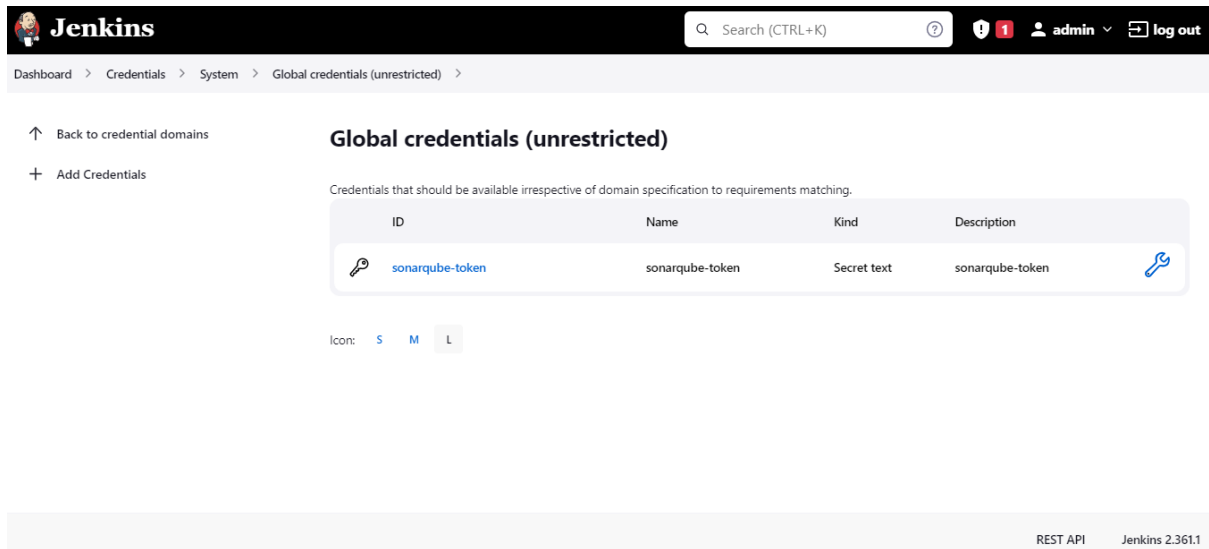
This screenshot is identical to the one above, showing the Jenkins dashboard with the 'Credentials' page. It displays the same navigation sidebar, 'Build Queue' and 'Build Executor Status' sections, and the 'Credentials' table with one entry. The 'Stores scoped to Jenkins' section also shows the same data. The 'Icon' section at the bottom has buttons for 'S', 'M', and 'L'.

T	P	Store	Domain	ID	Name
SecretText	Jenkins	global	sonarqube-token	sonarqube-token	



P	Store	Domains
Jenkins	global	

Icon: S M L

3: After that go to the global credentials:

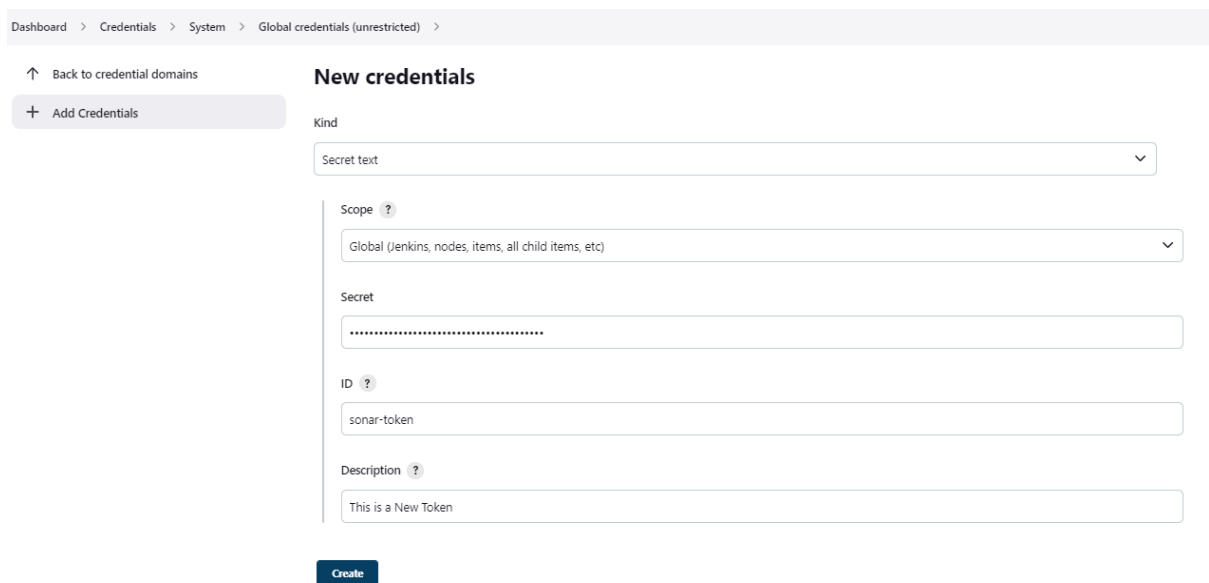


The screenshot shows the Jenkins web interface. At the top is the Jenkins logo and a search bar. Below the header is a breadcrumb trail: Dashboard > Credentials > System > Global credentials (unrestricted) >. On the left sidebar, there are two links: 'Back to credential domains' and 'Add Credentials'. The main content area is titled 'Global credentials (unrestricted)' and contains a sub-header 'Credentials that should be available irrespective of domain specification to requirements matching.' Below this is a table with the following data:

ID	Name	Kind	Description
 sonarqube-token	sonarqube-token	Secret text	sonarqube-token 

Below the table, there is a section for 'Icon' with three tabs: S, M, and L. At the bottom right of the page, there is a footer with 'REST API' and 'Jenkins 2.361.1'.

4: After that go to add credentials and fill information :



The screenshot shows the 'New credentials' form in Jenkins. The breadcrumb trail is: Dashboard > Credentials > System > Global credentials (unrestricted) >. The left sidebar has 'Back to credential domains' and 'Add Credentials'. The form is titled 'New credentials' and contains the following fields:

- Kind:** A dropdown menu with 'Secret text' selected.
- Scope:** A dropdown menu with 'Global (Jenkins, nodes, items, all child items, etc)' selected.
- Secret:** A text input field with a masked value '.....'.
- ID:** A text input field with the value 'sonar-token'.
- Description:** A text input field with the value 'This is a New Token'.

At the bottom of the form is a blue 'Create' button.

5: After that go to manage Jenkins go to Configure system:

**Jenkins** Search (CTRL+K) admin log out

Dashboard > Manage Jenkins > Configure System

**Configure System**

Home directory ?  
By default, Jenkins stores all of its data in this directory on the file system  
C:\ProgramData\Jenkins\jenkins

System Message  
This message will be displayed at the top of the Jenkins main page. This can be useful for posting notifications to your users

[Plain text] Preview

# of executors  
2

Save Apply

6: Scroll down at SonarQube server and add server url of sonarqube:

Dashboard > Manage Jenkins > Configure System

☐ Tool Locations

**SonarQube servers**

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☒ Environment variables Enable injection of SonarQube server configuration as build environment variables

**SonarQube installations**

List of SonarQube installations

Name  
sonarqube-8.9.2

Server URL  
Default is http://localhost:9000  
http://localhost:9000/dashboard

Server authentication token  
SonarQube authentication token. Mandatory when anonymous access is disabled.

Save Apply

7: After that go to sonarqube and copy that token:

The screenshot shows the SonarQube dashboard for a project named 'demo-project'. The browser address bar shows 'localhost:9000/dashboard?id=demo-project'. The dashboard has a top navigation bar with links for Projects, Issues, Rules, Quality Profiles, Quality Gates, and Administration. Below the navigation bar, there's a search bar and a user profile icon. The main content area is titled 'Analyze your project' and includes a message: 'We initialized your project on SonarQube, now it's up to you to launch analyses!'. There are two steps in a sequence: 1. 'Provide a token' and 2. 'Run analysis on your project'. Step 1 shows a token for 'demo-project': '10a90a9a4be9a9f71729450f5591f9fe204d2408'. A note explains that the token is used for identification and can be revoked. A 'Continue' button is present. At the bottom, there's a yellow warning box stating: 'Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.' Below this, it mentions 'SonarQube™ technology is powered by SonarSource SA' and provides links for Community Edition, Version 8.9.9, LGPL v3, Community, Documentation, Plugins, Web API, and About.

8: Paste that token in to this is new token:

The screenshot shows the Jenkins 'Configure System' page for SonarQube. The breadcrumb trail at the top is 'Dashboard > Manage Jenkins > Configure System >'. The main form has a dashed border and contains the following fields: 'Name' with the value 'sonarqube 8.9.9', 'Server URL' with the value 'http://localhost:9000/dashboard?id=demo-project', and 'Server authentication token' with a dropdown menu showing 'This is a New Token'. There are buttons for '+ Add' and 'Advanced...'. At the bottom of the form, there are 'Save' and 'Apply' buttons.

9: Then go to global tool configuration and scroll down Sonarqube scanner installation add version:

Dashboard > Manage Jenkins > Global Tool Configuration

### SonarQube Scanner installations

List of SonarQube Scanner installations on this system

[Add SonarQube Scanner](#)

SonarQube Scanner

Name

☒ Install automatically ?

Install from Maven Central

Version

[Add Installer](#)

[Save](#) [Apply](#)

10: After that go to Jenkins dashboard and create pipeline:

**Jenkins** Search (CTRL+K) admin log out

Dashboard > All >

Enter an item name

\* 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.

**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.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

**Folder**  
container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a namespace, so you can have multiple things of the same name as long as they are in different folders.

[OK](#)

11: go to advance project option:

Dashboard > sonar-pipeline >

### Configuration

- General
- Advanced Project Options**
- Pipeline

#### Pipeline

Definition

Pipeline script

Script ?

1

try sample Pipeline...

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save Apply

12: After that copy the script from here:

← → ↻ github.com/ravdy/DevOps

jleetutorial/maven-... Google

master 1 branch 0 tags

Go to file Add file Code

ravdy Update README.md a#56d82 27 days ago 53 commits	
Artifactory	Update Integrate_Artifactory_with_Jenkins.md 10 months ago
docker	added docker module 11 months ago
jenkins	Update Jenkins master and slave setup 7 months ago
maven	Create maven_installation.md 2 years ago
sonarqube	Update Sonarqube_with_database.md 13 months ago
README.md	Update README.md 27 days ago
Untitled Diagram.drawio	VPC peering 5 months ago

≡ README.md

## DevOps

This repository contains the documents related to DevOps Complete Course.

No description, website, or

Readme

88 stars

4 watching

1.8k forks

### Releases

No releases published

### Packages

No packages published

### Contributors 2

- ravdy
- yankils

13: paste that script over here:

The screenshot shows the Jenkins 'Configuration' page for a pipeline named 'sonar-pipeline'. The left sidebar has 'Pipeline' selected. The main area is titled 'Pipeline' and shows the 'Definition' as 'Pipeline script'. A 'Script' editor contains a Groovy script for a SonarQube analysis stage. Below the editor, the 'Use Groovy Sandbox' checkbox is checked. At the bottom are 'Save' and 'Apply' buttons.

```
13  steps{
14      sh 'mvn clean package'
15  }
16  }
17  stage('SonarQube analysis') {
18      def scannerHome = tool 'SonarScanner 4.0';
19      //
20      steps{
21          withSonarQubeEnv('sonarqube-8.9') {
22              // If you have configured more than one global server connection, you can specify its name
23              sh "s${scannerHome}/bin/sonar-scanner"
24          }
25      }
26  }
27  }
28  }
```

14: go to status :

The screenshot shows the Jenkins 'Status' page for the 'sonar-pipeline'. The left sidebar has 'Status' selected. The main area shows the pipeline name 'Pipeline sonar-pipeline' and a description 'This is Sonar-pipeline hehehehee'. There are buttons for 'Edit description' and 'Disable Project'. Below this is a 'Stage View' section with a message: 'No data available. This Pipeline has not yet run.' At the bottom is a 'Build History' section showing 'No builds'.

15 : After this we can see status of our project is there any bugs or not on the dashboard of sonarqube but due to some script issue we are not able to proceed further.