

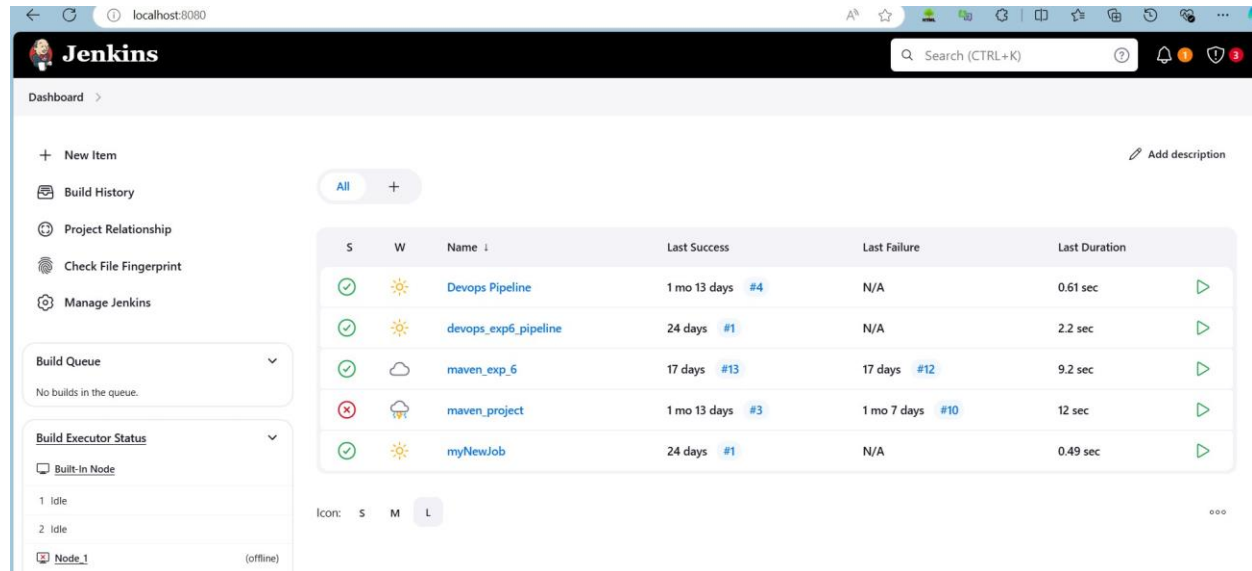
**Name : Aditya Kirtane , Div:D15C , Roll No.26**

## **Adv DevOps Practical 7**

**Aim:** To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

### **Steps to integrate Jenkins with SonarQube**

1. Open up Jenkins Dashboard on localhost, port 8080 or whichever port it is at for you.



2. Run SonarQube in a Docker container using this command -

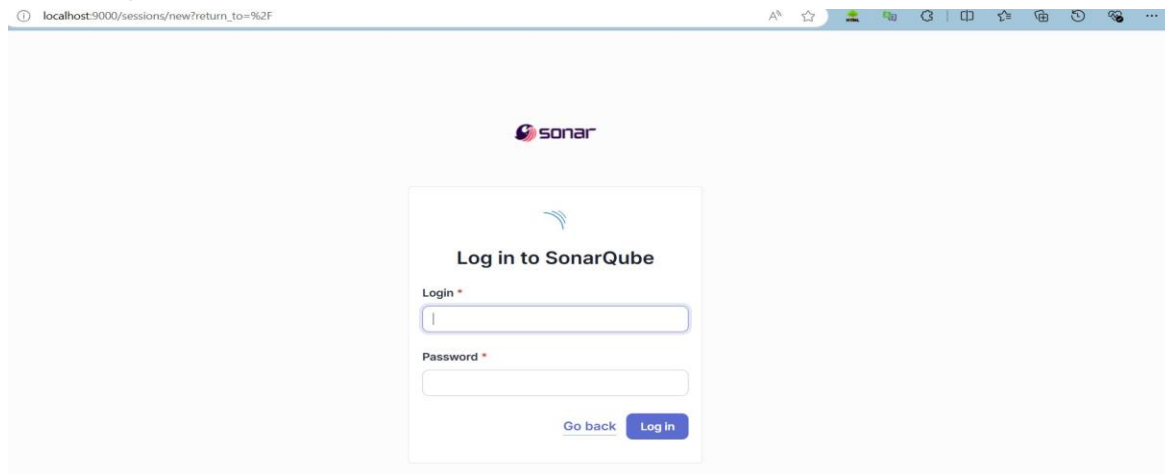
***docker run -d --name sonarqube -e SONAR\_ES\_BOOTSTRAP\_CHECKS\_DISABLE=true -p 9000:9000 sonarqube:latest***

**-----Warning: run below command only once**

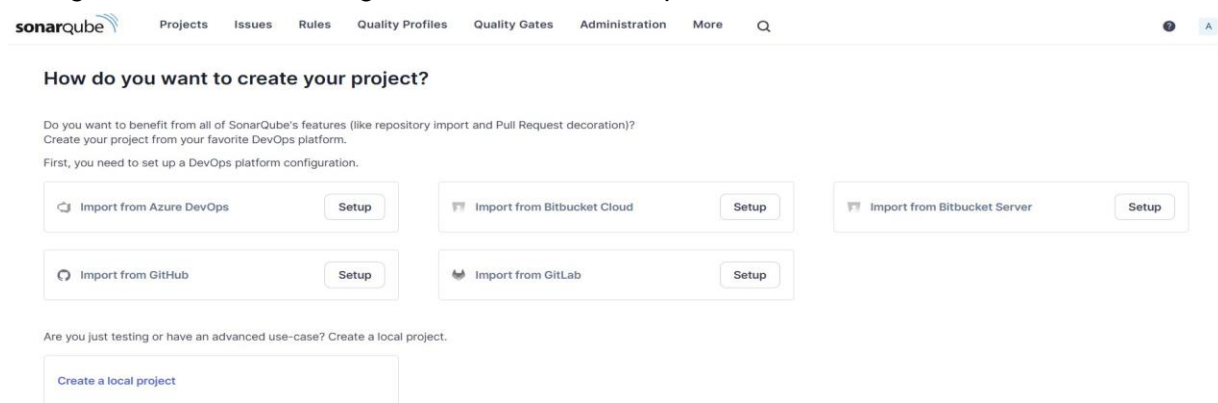
```
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Ayush Maurya> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
Unable to find image 'sonarqube:latest' locally
latest: Pulling from library/sonarqube
762bedf4b1b7: Pull complete
95f9bd9906fa: Pull complete
a32d681e6b99: Pull complete
aabdd0a18314: Pull complete
5161e45ecd8d: Pull complete
aeb0020dfa06: Pull complete
01548d361aea: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:bb44c58c1e04d8a147a3bb12af941c57e0100a5b21d10e599384d59bed36c86
Status: Downloaded newer image for sonarqube:latest
4af48468290f95b22362652ee37b96c935b0bed754945c62cf3b0d5d51a2ac0c
PS C:\Users\Ayush Maurya>
```

3. Once the container is up and running, you can check the status of SonarQube at localhost port 9000.



4. Login to SonarQube using username admin and password admin.



5. Create a manual project in SonarQube with the name sonarqube

1 of 2

## Create a local project

**Project display name \***

**Project key \***

**Main branch name \***

The name of your project's default branch [Learn More](#)

### Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code. This helps you focus attention on the most recent changes that follow the Clean as You Code methodology. [Learn more: Defining New Code](#)

Choose the baseline for new code for this project

☒ Use the global setting

Previous version

Any code that has changed since the previous version is considered new code.  
Recommended for projects following regular versions or releases.

☐ Define a specific setting for this project

☐ Previous version

Any code that has changed since the previous version is considered new code.  
Recommended for projects following regular versions or releases.

☐ Number of days

Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will be closed.  
Recommended for projects following continuous delivery.

Go to Manage Jenkins and search for SonarQube Scanner for Jenkins and install it.

The screenshot shows the Jenkins 'Manage Jenkins' > 'Plugins' page. A search bar at the top contains 'sonarq'. Below the search bar, a table lists the 'SonarQube Scanner' plugin (version 2.17.2), which was released 6 months and 29 days ago. The 'Download progress' section shows the installation status: 'Preparation' (Checking internet connectivity, Checking update center connectivity, Success), 'SonarQube Scanner' (Success), and 'Loading plugin extensions' (Success). A link to 'Go back to the top page' is provided, along with a checkbox to 'Restart Jenkins when installation is complete and no jobs are running'.

**Plugins**

Search: sonarq

Install Name Released

☐ SonarQube Scanner 2.17.2

External Site/Tool Integrations Build Reports

This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.

6 mo 29 days ago

**Download progress**

Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

SonarQube Scanner ☒ Success

Loading plugin extensions ☒ Success

→ [Go back to the top page](#)  
(you can start using the installed plugins right away)

→ ☐ Restart Jenkins when installation is complete and no jobs are running

6. Under Jenkins 'Manage Jenkins' then go to 'system', scroll and look for **SonarQube Servers** and enter the details.

In SonarQube installations: Under **Name** add <project name of sonarqube> for me

**adv\_devops\_7\_sonarqube**

In **Server URL** Default is **http://localhost:9000**

The screenshot shows the 'SonarQube servers' configuration page in Jenkins. It includes a checkbox for 'Environment variables' and a section for 'SonarQube installations'. The 'List of SonarQube installations' section contains a form with fields for 'Name' (adv\_devops\_7\_sonarqube), 'Server URL' (https://localhost:9000), and 'Server authentication token' (none). There is also an 'Add' button and an 'Advanced' dropdown menu.

**SonarQube servers**

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

☐ Environment variables

**SonarQube installations**

List of SonarQube installations

Name

adv\_devops\_7\_sonarqube

Server URL

Default is http://localhost:9000

https://localhost:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

+ Add

Advanced

7. Search for SonarQube Scanner under Global Tool Configuration.

**Dashboard > Manage Jenkins > Tools**

Dashboard > Manage Jenkins > Tools

Add Git

Gradle installations

Add Gradle

SonarScanner for MSBuild installations

Add SonarScanner for MSBuild

SonarQube Scanner installations

Add SonarQube Scanner

Ant installations

Check the “Install automatically” option. → Under name any name as identifier → Check the “Install automatically” option.

SonarQube Scanner installations

Add SonarQube Scanner

**SonarQube Scanner**

Name

sonarqube\_exp7

☒ Install automatically ?

**Install from Maven Central**

Version

SonarQube Scanner 6.1.0.4477

Add Installer

8. After the configuration, create a New Item in Jenkins, choose a freestyle project.

adv\_devops\_exp7

» Required field

**Freestyle project**  
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

**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.

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

**OK** **Branch Pipeline**

9. Choose this GitHub repository in Source Code Management.

[https://github.com/shazforiot/MSBuild\\_firstproject.git](https://github.com/shazforiot/MSBuild_firstproject.git)

It is a sample hello-world project with no vulnerabilities and issues, just to test the integration.

#### Source Code Management

☐ None

☒ Git ?

##### Repositories ?

###### Repository URL ?

[https://github.com/shazforiot/MSBuild\\_firstproject.git](https://github.com/shazforiot/MSBuild_firstproject.git)

###### Credentials ?

- none -

+ Add

Advanced

Add Repository

Execute SonarQube Scanner

JDK ?

JDK to be used for this SonarQube analysis

(Inherit From Job)

Path to project properties ?

Analysis properties ?

```
sonar.projectKey=adv_devops_7_sonarqube
sonar.host.url=http://localhost:9000
sonar.login=admin
sonar.sources=.
```

Additional arguments ?

JVM Options ?

Then save

Status

Changes

Workspace

Build Now

Configure

Delete Project

SonarQube

Rename

adv\_devops\_exp7

SonarQube

Permalinks

- Last build (#2), 1 day 20 hr ago
- Last stable build (#2), 1 day 20 hr ago
- Last successful build (#2), 1 day 20 hr ago
- Last completed build (#2), 1 day 20 hr ago

Add description

Disable Project

10. Go to [http://localhost:9000/<user\\_name>/permissions](http://localhost:9000/<user_name>/permissions) and allow Execute Permissions to the Admin user.

sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More

Administration

Configuration Security Projects System Marketplace

Global Permissions

Grant and revoke permissions to make changes at the global level. These permissions include editing Quality Profiles, executing analysis, and performing global system administration.

All Users Groups

Search for users or groups...

	Administer System ?	Administer ?	Execute Analysis ?	Create ?
<div>sonar-administrators</div> <div>System administrators</div>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Quality Gates <input checked="" type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input checked="" type="checkbox"/> Projects
<div>sonar-users</div> <div>Every authenticated user automatically belongs to this group</div>	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Projects
<div>Anyone <b>DEPRECATED</b></div> <div>Anybody who browses the application belongs to this group. If authentication is not enforced, assigned permissions also apply to non-authenticated users.</div>	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input type="checkbox"/> Projects
<div>Administrator admin</div>	<input checked="" type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input type="checkbox"/> Projects

4 of 4 shown

## IF CONSOLE OUTPUT FAILED:

### Step 1: Generate a New Authentication Token in SonarQube

#### 1. Login to SonarQube:

- Open your browser and go to `http://localhost:9000`.
- Log in with your admin credentials (default username is `admin`, and the password is either `admin` or your custom password if it was changed).

#### 2. Generate a New Token:

- Click on your **username** in the top-right corner of the SonarQube dashboard.
- Select **My Account** from the dropdown menu.
- Go to the **Security** tab.
- Under **Generate Tokens**, type a name for the token (e.g., "Jenkins-SonarQube").
- Click **Generate**.
- Copy the token and save it securely. You will need it in Jenkins.

### Step 2: Update the Token in Jenkins

#### 1. Go to Jenkins Dashboard:

#### 2. Configure the Jenkins Job:

#### 3. Update the SonarQube Token:

≡

Execute SonarQube Scanner

✕

JDK ?

JDK to be used for this SonarQube analysis

(Inherit From Job) ▾

Path to project properties ?

Analysis properties ?

sonar.projectKey=adv\_devops\_7\_sonarqube  
sonar.host.url=http://localhost:9000  
-Dsonar.login=sqp\_568834b7b5e77a92843e4b3072e044643ce921c1  
sonar.sources=.

Additional arguments ?

JVM Options ?

11. Run the Jenkins build.

The screenshot shows the Jenkins web interface for a project named 'adv\_devops\_exp7'. On the left, there is a sidebar with navigation options: Status, Changes, Workspace, Build Now, Configure, Delete Project, SonarQube, and Rename. The main area displays the SonarQube logo and a 'Permalinks' section with a list of build links: 'Last build (#10), 19 sec ago', 'Last stable build (#10), 19 sec ago', 'Last successful build (#10), 19 sec ago', 'Last failed build (#8), 22 min ago', 'Last unsuccessful build (#8), 22 min ago', and 'Last completed build (#10), 19 sec ago'. Below this is a 'Build History' table showing build #10 as successful on Sep 18, 2024, at 2:36 PM. On the right, there are buttons for 'Add description' and 'Disable Project'.

12. Once the build is complete, check project on SonarQube

The screenshot shows the SonarQube project overview page for 'adv\_devops\_7\_sonarqube'. The page has a top navigation bar with links to Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, and More. Below the navigation bar, the project name is shown with a dropdown menu set to 'main'. The 'Overview' tab is selected, showing a 'main' branch with a 'Version not provided' status and a 'Set as homepage' button. A prominent message box says 'Don't let issues accumulate. Discover 'Clean as You Code!'' with 'Take the Tour' and 'Not now' buttons. Below this, a 'Quality Gate' section shows a green checkmark and the word 'Passed', with a note 'Last analysis 5 minutes ago'. A warning box indicates 'The last analysis has warnings. See details'. At the bottom, there are tabs for 'New Code' and 'Overall Code', and a section for 'Security', 'Reliability', and 'Maintainability'.

In this way, we have integrated Jenkins with SonarQube for SAST.



**Conclusion:**

In this project, we integrated Jenkins with SonarQube for automated static application security testing (SAST). We set up SonarQube using Docker, configured Jenkins with the necessary plugins and authentication, and linked it to a GitHub repository. The SonarQube scanner was added as a build step, enabling continuous code analysis for vulnerabilities, code smells, and quality issues, ensuring automated reporting and continuous code quality improvement.