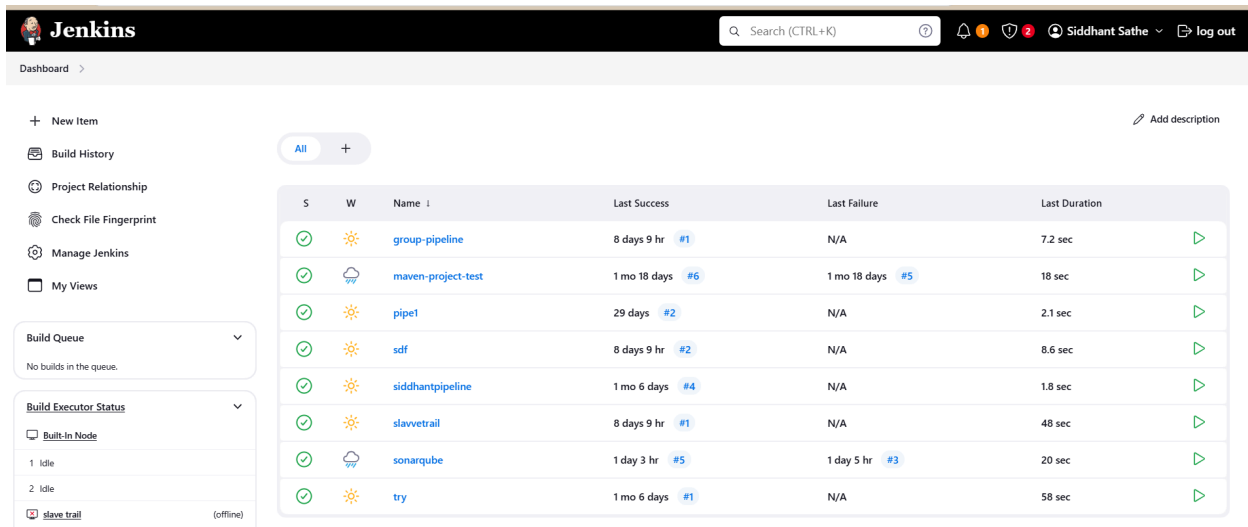


Experiment 7

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

Theory:

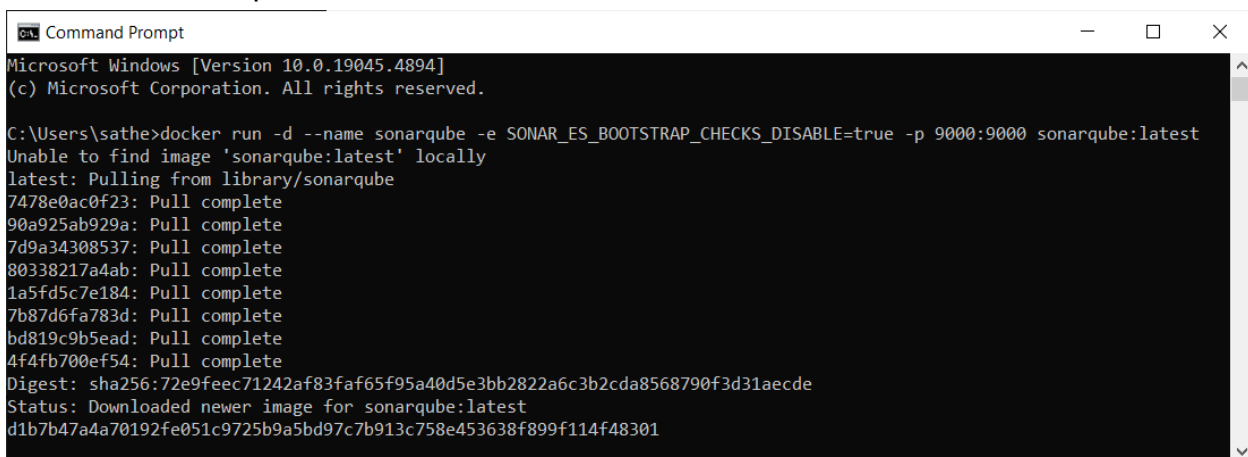
Step-1: Open up Jenkins Dashboard on localhost, port 8080 or whichever port it is at for you.



The screenshot shows the Jenkins Dashboard interface. On the left, there is a sidebar with navigation links: New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, and My Views. Below these are sections for Build Queue (showing 'No builds in the queue') and Build Executor Status (showing '1 idle', '2 idle', and 'slave trail (offline)'). The main area displays a table of builds with columns: S (Status), W (Icon), Name, Last Success, Last Failure, Last Duration, and a play button icon. The table lists several builds, including 'group-pipeline', 'maven-project-test', 'pipet', 'sdf', 'siddhantpipeline', 'slavvetrail', 'sonarqube', and 'try'.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	group-pipeline	8 days 9 hr #1	N/A	7.2 sec
✓	☁	maven-project-test	1 mo 18 days #6	1 mo 18 days #5	18 sec
✓	☀	pipet	29 days #2	N/A	2.1 sec
✓	☀	sdf	8 days 9 hr #2	N/A	8.6 sec
✓	☀	siddhantpipeline	1 mo 6 days #4	N/A	1.8 sec
✓	☀	slavvetrail	8 days 9 hr #1	N/A	48 sec
✓	☁	sonarqube	1 day 3 hr #5	1 day 5 hr #3	20 sec
✓	☀	try	1 mo 6 days #1	N/A	58 sec

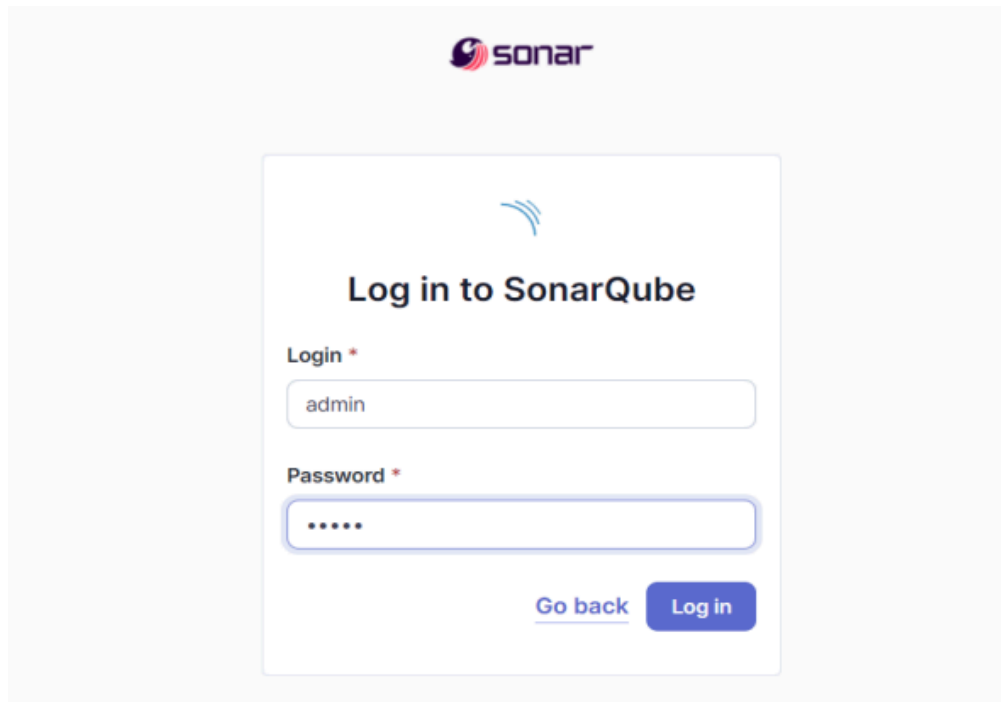
Step-2: Run SonarQube in a Docker container using this command :- `docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest`



```

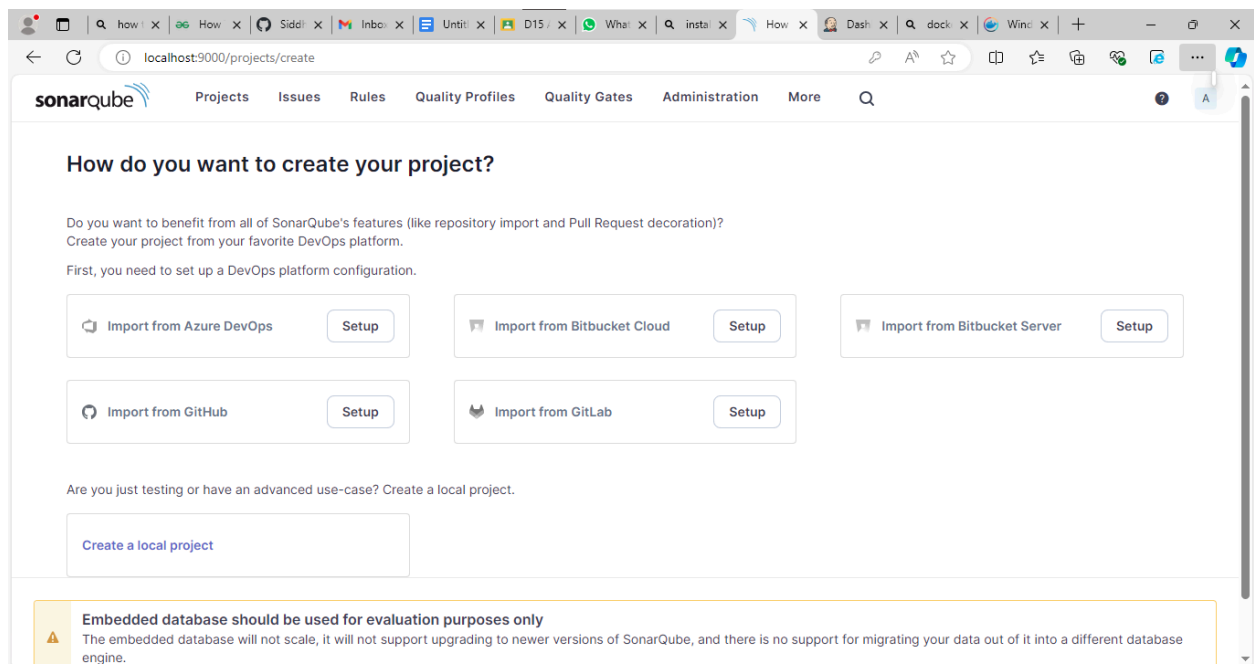
C:\Users\sathe>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
Unable to find image 'sonarqube:latest' locally
latest: Pulling from library/sonarqube
7478e0ac0f23: Pull complete
90a925ab929a: Pull complete
7d9a34308537: Pull complete
80338217a4ab: Pull complete
1a5fd5c7e184: Pull complete
7b87d6fa783d: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Downloaded newer image for sonarqube:latest
d1b7b47a4a70192fe051c9725b9a5bd97c7b913c758e453638f899f114f48301
  
```

Step-3: Once the container is up and running, you can check the status of SonarQube at localhost port 9000. The login id is “admin” and the password is also “admin”.



The image shows the SonarQube login interface. At the top is the Sonar logo. Below it is a large white box with a blue Sonar icon and the text "Log in to SonarQube". Inside this box, there are two input fields: "Login *" with the text "admin" and "Password *" with masked characters "*****". Below the password field are two buttons: "Go back" (a blue link) and "Log in" (a blue button).

Step-4: Create a local project in SonarQube with the name sonarqube



The image shows the SonarQube project creation page in a web browser. The browser's address bar shows "localhost:9000/projects/create". The page has a navigation bar with links: "sonarqube", "Projects", "Issues", "Rules", "Quality Profiles", "Quality Gates", "Administration", and "More". The main heading is "How do you want to create your project?". Below this, it asks if the user wants to benefit from all features and suggests creating a project from a DevOps platform. It lists several options with "Setup" buttons: "Import from Azure DevOps", "Import from Bitbucket Cloud", "Import from Bitbucket Server", "Import from GitHub", and "Import from GitLab". At the bottom, it asks if the user is just testing or has an advanced use-case, and provides a "Create a local project" button. A yellow warning box at the bottom states: "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."

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](#) 

Cancel

Next

1 2 3

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 to your project, enabling you to 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 become part of the overall code.

Recommended for projects following continuous delivery.

☐ Reference branch

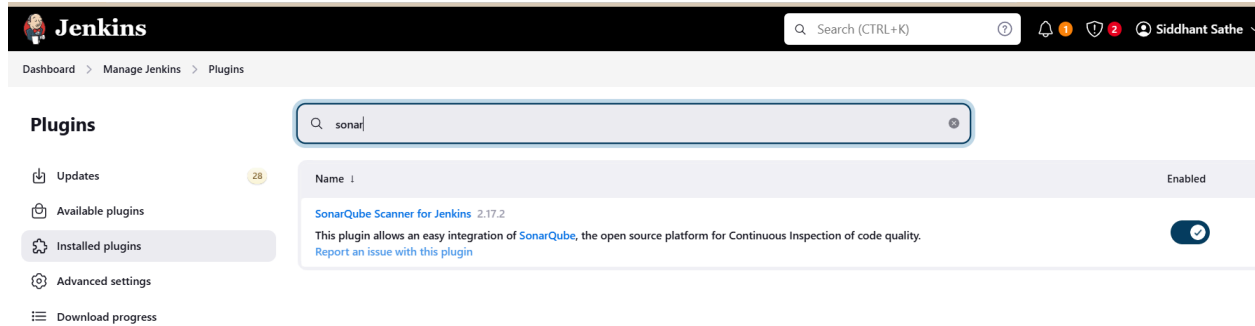
Choose a branch as the baseline for the new code.

Recommended for projects using feature branches.

Back

Create project

Step-5: Setup the project and come back to Jenkins Dashboard. Go to Manage Jenkins → Plugins and search for SonarQube Scanner in Available Plugins and install it.



The screenshot shows the Jenkins Dashboard with the breadcrumb 'Dashboard > Manage Jenkins > Plugins'. The 'Plugins' section is active, displaying a search bar with 'sonar' entered. Below the search bar, a table lists the installed plugins. The 'SonarQube Scanner for Jenkins' plugin, version 2.17.2, is shown as installed and enabled. The description states: 'This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality. Report an issue with this plugin'.

Name	Enabled
SonarQube Scanner for Jenkins 2.17.2 This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality. Report an issue with this plugin	<input checked="" type="checkbox"/>

Step-6: Under 'Manage Jenkins → System', look for SonarQube Servers and enter these details. Name : sonarqube, Server URL : http://localhost:9000

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

sonarqube

Server URL

Default is http://localhost:9000

http://localhost:9000

Server authentication token

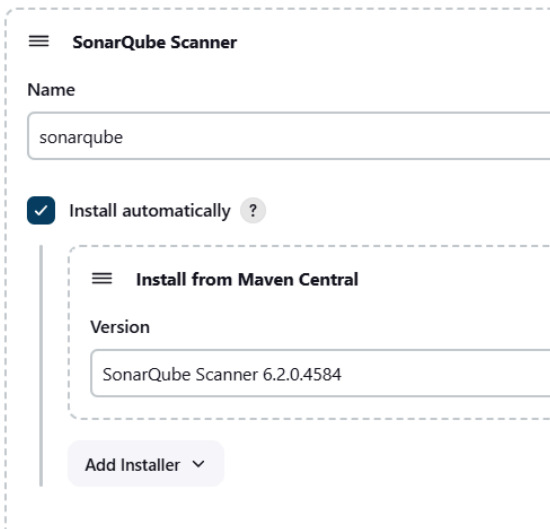
SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

+ Add

Advanced

Step-7: Search for SonarQube Scanner under Global Tool Configuration. Choose the latest configuration and choose Install automatically. Manage Jenkins → Tools → SonarQube Scanner Installation



☰ SonarQube Scanner

Name

sonarqube

☒ Install automatically ?

☰ Install from Maven Central

Version

SonarQube Scanner 6.2.0.4584

Add Installer ▾

Step-8: After the configuration, create a New Item in Jenkins, choose a freestyle project named sonarqube.

New Item

Enter an item name

sonarqube

Select an item type



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.

Step-9: Choose this GitHub repository in Source Code Management.

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 ?

Credentials ?

- none -

+ Add +

Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

Save

Apply

Step-10: Under Build-> Execute SonarQube Scanner, enter these Analysis Properties. Mention the SonarQube Project Key, Login, Password, Source path and Host URL.

Execute SonarQube Scanner

JDK ?

JDK to be used for this SonarQube analysis

Path to project properties ?

Analysis properties ?

```
sonar.projectKey=sonarqube
sonar.login=sqp_9b85237263881f919ee5cb245bab294c2f3ce329
sonar.sources=HelloWorldCore
sonar.host.url=http://localhost:9000
```

Additional arguments ?

JVM Options ?

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	<input type="text" value="Search for users or groups..."/>	
	Administer System ?	Administer ?	Execute Analysis ?	Create ?
A Administrator admin	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input type="checkbox"/> Projects

jenkins

Dashboard > sonarqube >

Status

Changes

Workspace

Build Now

Configure

Delete Project

SonarQube

Rename

Build History

trend

Filter...

#5

Sep 26, 2024, 7:40 PM

#4

Sep 26, 2024, 5:58 PM

#3

Sep 26, 2024, 5:47 PM

#2

Sep 26, 2024, 5:34 PM

sonarqube

SonarQube

Permalinks

- Last build (#5), 1 day 4 hr ago
- Last stable build (#5), 1 day 4 hr ago
- Last successful build (#5), 1 day 4 hr ago
- Last failed build (#3), 1 day 5 hr ago
- Last unsuccessful build (#3), 1 day 5 hr ago
- Last completed build (#5), 1 day 4 hr ago

Add description

Disable Project

sonarqube

Build History

Filter...

#5

Sep 26, 2024, 7:40 PM

#4

Sep 26, 2024, 5:58 PM

#3

Sep 26, 2024, 5:47 PM

#2

Sep 26, 2024, 5:34 PM


Snip & Sketch

Snip saved to clipboard

Select here to mark up and share the image



Console Output



```
Started by user Siddhant Sathe
Running as SYSTEM

Building on the built-in node in workspace C:\ProgramData\Jenkins\jenkins\workspace\sonarqube
[sonarqube] $ C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube\bin\sonar-scanner.bat -Dsonar.host.url=http://localhost:9000 -Dsonar.projectKey=sonarqube -Dsonar.login=sqp_9b85237263881f919ee5cb245bab294c2f3ce329 -Dsonar.host.url=http://localhost:9000 -Dsonar.sources=HelloWorldCore -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\jenkins\workspace\sonarqube
17:58:01.518 WARN Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
17:58:01.527 INFO Scanner configuration file: C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube\bin\..\conf\sonar-scanner.properties
17:58:01.528 INFO Project root configuration file: NONE
17:58:01.551 INFO SonarScanner CLI 6.2.0.4584
17:58:01.553 INFO Java 21.0.4 Eclipse Adoptium (64-bit)
17:58:01.558 INFO Windows 10 10.0 amd64
17:58:01.586 INFO User cache: C:\Windows\system32\config\systemprofile\.sonar\cache
17:58:02.288 INFO JRE provisioning: os[windows], arch[amd64]
17:58:02.600 INFO Communicating with SonarQube Server 10.6.0.92116
17:58:03.043 INFO Starting SonarScanner Engine...
17:58:03.044 INFO Java 17.0.11 Eclipse Adoptium (64-bit)
17:58:04.214 INFO Load global settings
17:58:04.535 INFO Load global settings (done) | time=318ms
17:58:04.540 INFO Server id: 147B411E-AZ1k70wCd_37MaFi8bMy
```

```
17:58:37.850 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarqube
17:58:37.851 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
17:58:37.851 INFO More about the report processing at http://localhost:9000/api/ce/task?id=30158bd7-ca2f-47d2-94c3-09b6c41bf32b
17:58:37.865 INFO Analysis total time: 33.094 s
17:58:37.867 INFO SonarScanner Engine completed successfully
17:58:37.917 INFO EXECUTION SUCCESS
17:58:37.919 INFO Total time: 36.394s
Finished: SUCCESS
```

Step-13: Once the build is complete, check the project in SonarQube.

☆ sonarqube PUBLIC ✓ Passed

Last analysis: 1 minute ago

The main branch of this project is empty.

1 of 1 shown

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More

☆ sonarqube / main ✓

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

main Version not provided [Set as homepage](#)

Quality Gate ✓ **Passed** Last analysis 17 minutes ago

⚠ The last analysis has warnings. [See details](#)

New Code Overall Code

Security 0 Open issues 0 H 0 M 0 L A	Reliability 0 Open issues 0 H 0 M 0 L A	Maintainability 0 Open issues 0 H 0 M 0 L A
Accepted issues 0 Valid issues that were not fixed	Coverage On 0 lines to cover.	Duplications 0.0% On 86 lines.
Security Hotspots 0 A		

1 of 2

Create a local project

Project display name *

Adv_exp7



Project key *

Adv_exp7



Main branch name *

main

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

Cancel

Next

localhost:9000/tutorials?id=Adv_exp7

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More


Adv_exp7 / main


Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information


Analysis Method


Use this page to manage and set-up the way your analyses are performed.


How do you want to analyze your repository?

 With Jenkins

 With GitHub Actions

 With Bitbucket Pipelines

 With GitLab CI

 With Azure Pipelines

Other CI
SonarQube integrates with your workflow no matter which CI tool you're using.

Locally
Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI.

Error:

Build failed due to missing sonar source while configurations

Overcame by adding correct path to sonar.source

Console Output

```
Started by user Siddhant Sathe
Running as SYSTEM
Building on the built-in node in workspace C:\ProgramData\Jenkins\jenkins\workspace\sonarqube
[sonarqube] $ C:\ProgramData\Jenkins\jenkins\tools\udson.plugins.sonar.SonarRunnerInstallation\sonarqube\bin\sonar-scanner.bat -Dsonar.host.url=http://localhost:9000 -Dsonar.projectKey=sonarqube -Dsonar.login=sqp_9b85237263881f919ee5cb245bab294c2f3ce329 -Dsonar.host.url=http://localhost:9000 -Dsonar.sources=sonarqube -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\jenkins\workspace\sonarqube
17:47:09.958 WARN Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
17:47:09.970 INFO Scanner configuration file: C:\ProgramData\Jenkins\jenkins\tools\udson.plugins.sonar.SonarRunnerInstallation\sonarqube\bin\..\conf\sonar-scanner.properties
17:47:09.972 INFO Project root configuration file: NONE
17:47:09.996 INFO SonarScanner CLI 6.2.0.4584
17:47:09.998 INFO Java 21.0.4 Eclipse Adoptium (64-bit)
17:47:10.003 INFO Windows 10 10.0 amd64
17:47:10.039 INFO User cache: C:\Windows\system32\config\systemprofile\.sonar\cache
17:47:10.718 INFO JRE provisioning: os[windows], arch[amd64]
17:47:11.098 INFO Communicating with SonarQube Server 10.6.0.92116
17:47:11.709 INFO Starting SonarScanner Engine...
17:47:11.710 INFO Java 17.0.11 Eclipse Adoptium (64-bit)
17:47:13.465 INFO Load global settings
17:47:13.665 INFO Load global settings (done) | time=199ms
17:47:13.670 INFO Server id: 147B411E-AZIk70wCd_37MaE18bMy
17:47:13.688 INFO Loading required plugins
17:47:13.688 INFO Load plugins index
17:47:13.757 INFO Load plugins index (done) | time=69ms
17:47:13.758 INFO Load/download plugins
17:47:13.855 INFO Load/download plugins (done) | time=96ms
17:47:14.525 INFO Process project properties
17:47:14.542 ERROR Invalid value of sonar.sources for sonarqube
17:47:14.559 ERROR The folder 'sonarqube' does not exist for 'sonarqube' (base directory = C:\ProgramData\Jenkins\jenkins\workspace\sonarqube)
17:47:14.597 INFO EXECUTION FAILURE
17:47:14.599 INFO Total time: 4.632s
WARN: Unable to locate 'report-task.txt' in the workspace. Did the SonarScanner succeed?
ERROR: SonarQube scanner exited with non-zero code: 1
Finished: FAILURE
```