

Experiment 7

Aim :- To perform static analysis on python programs using sonarqube SAST process

Theory :-

SonarQube is a universal tool for static code analysis that has become more or less the industry standard. Keeping code clean, simple, and easy to read is also a lot easier with SonarQube.

What is SonarQube?

SonarQube is an open-source platform developed by SonarSource for continuous inspection of code quality. Sonar does static code analysis, which provides a detailed report of bugs, code smells, vulnerabilities, code duplications. It supports 25+ major programming languages through built-in rulesets and can also be extended with various plugins.

Benefits of SonarQube

Sustainability - Reduces complexity, possible vulnerabilities, and code duplications, optimising the life of applications. **Increase productivity** - Reduces the scale, cost of maintenance, and risk of the application; as such, it removes the need to spend more time changing the code

Quality code - Code quality control is an inseparable part of the process of software development.

Detect Errors - Detects errors in the code and alerts developers to fix them automatically before submitting them for output.

Increase consistency - Determines where the code criteria are breached and enhances the quality

Business scaling - No restriction on the number of projects to be evaluated

Enhance developer skills - Regular feedback on quality problems helps developers to improve their coding skills

Why SonarQube?

Developers working with hard deadlines to deliver the required functionality to the customer. It is so important for developers that many times they compromise with the code quality, potential bugs, code duplications, and bad distribution of complexity. Additionally, they tend to leave unused variables, methods, etc. In this scenario, the code would work in the desired way.

To avoid these issues in code, developers should always follow the good coding practice, but sometimes it is not possible to follow the rules and maintain the good quality as there may be many reasons.

In order to achieve continuous code integration and deployment, developers need a tool that not only works once to check and tell them the problems in the code but also to track and control the code to check continuous code quality. To satisfy all these requirements, here comes SonarQube in the picture.

Steps :-

1)Download sonarqube

Download | SonarQube

SonarScanner | SonarQube Docs

sonarqube.org/downloads/

NEW SonarQube 9.1 is here! Project PDFs, JS AWS Lambda taint analysis, Kotlin coroutine rules and more! →

sonarqube

Product ▾ What's New Documentation Community Download

Download SonarQube

The leading product for Code Quality and Security

HELPING DEVS SINCE 2008

Version: 9.1 | Release: September 2021 | Getting Started | Release Notes | Upgrade Notes | Available From DockerHub

Community EDITION

Used and loved by 200,000+ companies

FREE & OPEN SOURCE

Download for free

All the following features:

- ✓ Static code analysis for 15 languages
Java, JavaScript, C#, TypeScript, Kotlin, Ruby, Go, Scala, Flex, Python, PHP, LISP, REXX, COBOL, C, C++

Developer EDITION

Built for developers by developers

Download

Community Edition plus:

- ✓ C, C++, Obj-C, Swift, ABAP, T-SQL, PL/SQL support
- ✓ Detection of Injection Flaws

Enterprise EDITION

Designed to meet Enterprise Requirements

Download

Developer Edition plus:

- ✓ Portfolio Management & PDF Executive Reports
- ✓ Project PDF reports

Data Center EDITION

Designed for High Availability

Download

Enterprise Edition plus:

- ✓ Component redundancy
- ✓ Data resiliency
- ✓ Horizontal Scalability

2) After downloading, set Environment Variables. Add “sonarqube-9.1.0.47736\bin” to Path.

Thank you for downloading SonarQube

SonarScanner | SonarQube Docs

docs.sonarqube.org/latest/analysis/scan/sonarscanner/

sonarqube Docs 9.1 ▾

Search...

Try Out SonarQube

Requirements ▾

Setup and Upgrade ▾

Analyzing Source Code ▴

Overview

Scanners ▴

- SonarScanner for Gradle
- SonarScanner for .NET
- SonarScanner for Maven
- SonarScanner for Azure DevOps
- SonarScanner for Jenkins
- SonarScanner for Ant
- SonarScanner**

Analysis Parameters

Languages ▾

Test Coverage & Execution

Importing External Issues ▾

Background Tasks

SonarQube Community

SonarScanner

By SonarSource | GNU LGPL 3 | Issue Tracker

4.6.2

2021-05-07

Update dependencies, bug fix

Linux 64-bit | Windows 64-bit | Mac OS X 64-bit | Docker

Any (Requires a pre-installed JVM) | Release notes

Show more versions

The SonarScanner is the scanner to use when there is no specific scanner for your build system.

Configuring your project

Create a configuration file in your project's root directory called sonar-project.properties

```
# must be unique in a given SonarQube instance
sonar.projectKey=my:project

# --- optional properties ---

# defaults to project key
#sonar.projectName=My project
# defaults to 'not provided'
#sonar.projectVersion=1.0

# Path is relative to the sonar-project.properties file. Defaults to .
#sonar.sources=.
```

On this page

- Configuring your project
- Running SonarScanner from the zip file
- Running SonarScanner from the Docker image
- Scanning C, C++, or ObjectiveC Projects
- Sample Projects
- Alternatives to sonar-project.properties
- Alternate Analysis Directory
- Advanced Docker Configuration
- Troubleshooting

3) Open command prompt. Run commands:

- **cd “sonarqube-9.1.0.47736\bin\windows-x86-64”**
- **StartSonar.bat**

```

Microsoft Windows [Version 10.0.19043.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Priyansid> "C:\Users\Priyansid\Downloads\sonarqube-9.1.0.47736\bin\windows-x86-64"

C:\Users\Priyansid\Downloads\sonarqube-9.1.0.47736\bin\windows-x86-64>StartSonar.bat
Wrapper --> Wrapper Started as Console
Wrapper Launching a JVM...
JVM 1 Wrapper (Version 3.2.3) http://wrapper.tanukisoftware.org
JVM 1 Copyright 1999-2006 Tanuki Software, Inc. All Rights Reserved.
JVM 1
JVM 1 2021.09.29 13:50:37 INFO app[[0.s.s.a.Filesystem] Cleaning or creating temp directory C:\Users\Priyansid\Downloads\sonarqube-9.1.0.47736\temp
JVM 1 2021.09.29 13:50:37 INFO app[[0.s.s.es.EsSettings] Elasticsearch listening on [HTTP: 127.0.0.1:9001, TCP: 127.0.0.1:53055]
JVM 1 2021.09.29 13:50:37 INFO app[[0.s.s.a.ProcessLauncherImpl] Launch process[key=es*, ipcIndex=1, logfileNamePrefix=es]] from C:\Users\Priyansid\Downloads\sonarqube-9.1.0.47736\elasticsearch: C:\Program Files\Java\jdk-11.0.12\bin\java -XX:+UseG1GC -Djava.io.tmpdir=C:\Users\Priyansid\Downloads\sonarqube-9.1.0.47736\temp -XX:ErrorFile=.../logs/es_err_pid%u.log -Des.networkaddress.cache.ttl=60 -Des.networkaddress.cache.negative.ttl=10 -Xss1m -Djava.net.handler.io.ssl=DnsNioSocketFactory -Djava.encoding=UTF-8 -Djna.nosys=true -XX:-OmitStackTraceInFastThrow -Dio.netty.noUnsafe=true -Dio.netty.noKeySetOptimization=true -Dio.netty.recycler.maxCapacityPerThread=0 -Dio.netty.allocator.numDirectArenas=0 -Dlog4j2.shutdownHookEnabled=false -Dlog4j2.disable.jmx=true -Djava.locale.providers=COMPAT -Xms512m -Xmx512m -XX:MaxDirectMemorySize=256m -XX:HeapDumpOnOutOfMemoryError -Delasticsearch.es-path.home=C:\Users\Priyansid\Downloads\sonarqube-9.1.0.47736\elasticsearch -Des.path.conf=C:\Users\Priyansid\Downloads\sonarqube-9.1.0.47736\temp\conf\es -cp lib\* org.elasticsearch.bootstrap.Elasticsearch
JVM 1 2021.09.29 13:50:37 INFO app[[0.s.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running
JVM 1 2021.09.29 13:50:39 ERROR org.elasticsearch.common.ProcessManager$EsManagedProcess Failed to check status
JVM 1 org.elasticsearch.ElasticsearchException: java.util.concurrent.ExecutionException: java.net.ConnectException: Timeout connecting to [/127.0.0.1:9001]
JVM 1 at org.elasticsearch.client.RestHighLevelClient.performClientRequest(RestHighLevelClient.java:2078)
JVM 1 at org.elasticsearch.client.RestHighLevelClient.internalPerformRequest(RestHighLevelClient.java:1732)
JVM 1 at org.elasticsearch.client.RestHighLevelClient.performRequest(RestHighLevelClient.java:1702)
JVM 1 at org.elasticsearch.client.RestHighLevelClient.performRequestAndParseEntity(RestHighLevelClient.java:1672)
JVM 1 at org.elasticsearch.client.ClusterClient.health(ClusterClient.java:119)
JVM 1 at org.sonar.application.es.EsConnectorImpl.checkClusterHealthStatus(EsConnectorImpl.java:64)
JVM 1 at org.sonar.application.process.EsManagedProcess.checkStatus(EsManagedProcess.java:90)
JVM 1 at org.sonar.application.process.EsManagedProcess.checkOperational(EsManagedProcess.java:75)
JVM 1 at org.sonar.application.process.EsManagedProcess.isOperational(EsManagedProcess.java:60)
JVM 1 at org.sonar.application.process.ManagerProcessHandler.refreshState(ManagerProcessHandler.java:220)
JVM 1 at org.sonar.application.process.ManagerProcessHandler$EventWatcher.run(ManagerProcessHandler.java:285)
JVM 1 Caused by: java.util.concurrent.ExecutionException: java.net.ConnectException: Timeout connecting to [/127.0.0.1:9001]
JVM 1 at org.elasticsearch.common.util.concurrent.BaseFuture$Sync.getValue(BaseFuture.java:262)
JVM 1 at org.elasticsearch.common.util.concurrent.BaseFuture$Sync.get(BaseFuture.java:249)
JVM 1 at org.elasticsearch.common.util.concurrent.BaseFuture.get(BaseFuture.java:76)
JVM 1 at org.elasticsearch.client.RestHighLevelClient.performClientRequest(RestHighLevelClient.java:2075)
JVM 1 ... 10 common frames omitted
JVM 1 Caused by: java.net.ConnectException: Timeout connecting to [/127.0.0.1:9001]
JVM 1 at org.apache.http.nio.pool.RouteSpecificPool.timeout(RouteSpecificPool.java:169)
JVM 1 at org.apache.http.nio.pool.AbstractNioConnPool.requestTimeout(AbstractNioConnPool.java:628)
JVM 1 at org.apache.http.nio.pool.AbstractNioConnPool$InternalSessionRequestCallback.timeout(AbstractNioConnPool.java:894)
JVM 1 at org.apache.http.impl.nio.reactor.SessionRequestImpl.timeout(SessionRequestImpl.java:184)
JVM 1 at org.apache.http.impl.nio.reactor.DefaultConnectingIOReactor.processTimeouts(DefaultConnectingIOReactor.java:214)
JVM 1 at org.apache.http.impl.nio.reactor.DefaultConnectingIOReactor.processEvents(DefaultConnectingIOReactor.java:158)
JVM 1 at org.apache.http.impl.nio.reactor.AbstractMultiworkerIOReactor.execute(AbstractMultiworkerIOReactor.java:351)
JVM 1 at org.apache.http.impl.nio.reactor.PoolingClientConnectionManager.execute(PoolingClientConnectionManager.java:221)
JVM 1 at org.apache.http.impl.nio.client.CloseableHttpAsyncClientBase$1.run(CloseableHttpAsyncClientBase.java:64)
JVM 1 at java.base/java.lang.Thread.run(Thread.java:834)

```

```

jvm 1 at org.elasticsearch.client.RestHighLevelClient.performRequest(RestHighLevelClient.java:1702)
jvm 1 at org.elasticsearch.client.RestHighLevelClient.performRequestAndParseEntity(RestHighLevelClient.java:1672)
jvm 1 at org.elasticsearch.client.ClusterClient.health(ClusterClient.java:119)
jvm 1 at org.sonar.application.es.EsConnectorImpl.getClusterHealthStatus(EsConnectorImpl.java:64)
jvm 1 at org.sonar.application.process.EsManagedProcess.checkStatus(EsManagedProcess.java:90)
jvm 1 at org.sonar.application.process.EsManagedProcess.checkOperational(EsManagedProcess.java:75)
jvm 1 at org.sonar.application.process.EsManagedProcess.isOperational(EsManagedProcess.java:60)
jvm 1 at org.sonar.application.process.ManagedProcessHandler.refreshState(ManagedProcessHandler.java:220)
jvm 1 at org.sonar.application.process.ManagedProcessHandler$EventWatcher.run(ManagedProcessHandler.java:285)
jvm 1 Caused by: java.util.concurrent.ExecutionException: java.net.ConnectException: Timeout connecting to [/127.0.0.1:9001]
jvm 1 at org.elasticsearch.common.util.concurrent.BaseFutureSync.getBaseFutureSync(BaseFutureSync.java:262)
jvm 1 at org.elasticsearch.common.util.concurrent.BaseFutureSync.get(BaseFutureSync.java:249)
jvm 1 at org.elasticsearch.common.util.concurrent.BaseFutureSync.get(BaseFutureSync.java:76)
jvm 1 at org.elasticsearch.client.RestHighLevelClient.performClientRequest(RestHighLevelClient.java:2075)
jvm 1 ... 10 common frames omitted
jvm 1 Caused by: java.net.ConnectException: Timeout connecting to [/127.0.0.1:9001]
jvm 1 at org.apache.http.pool.RouteSpecificPool.timeout(RouteSpecificPool.java:160)
jvm 1 at org.apache.http.nio.pool.AbstractNIOConnPool.requestTimeout(AbstractNIOConnPool.java:628)
jvm 1 at org.apache.http.nio.pool.AbstractNIOConnPool$InternalSessionRequestCallback.timeout(AbstractNIOConnPool.java:894)
jvm 1 at org.apache.http.impl.nio.reactor.SessionRequestImpl.timeout(SessionRequestImpl.java:184)
jvm 1 at org.apache.http.impl.nio.reactor.DefaultConnectingIOReactor.processTimeout(DefaultConnectingIOReactor.java:216)
jvm 1 at org.apache.http.impl.nio.reactor.DefaultConnectingIOReactor.processEvents(DefaultConnectingIOReactor.java:158)
jvm 1 at org.apache.http.impl.nio.reactor.AbstractMultiworkerIOReactor.execute(AbstractMultiworkerIOReactor.java:351)
jvm 1 at org.apache.http.impl.nio.conn.PoolingNHttpClientConnectionManager.execute(PoolingNHttpClientConnectionManager.java:221)
jvm 1 at org.apache.http.impl.nio.client.CloseableHttpAsyncClientBase$.run(CloseableHttpAsyncClientBase.java:64)
jvm 1 at java.base/java.lang.Thread.run(Thread.java:834)
jvm 1 2021.09.29 13:50:50 INFO app[[[o.s.a.scheduler.impl.Process]] is up
jvm 1 2021.09.29 13:50:50 INFO app[[[o.s.a.ProcessLauncherImpl] Launch process[[key='web', ipcIndex=2, logFileNamePrefix=web]] from [C:\Users\Priyanshi\Downloads\sonarqube-9.1.0.47736\ C:\Program Files\Java\jdk-11.0.12\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\Users\Priyanshi\Downloads\sonarqube-9.1.0.47736\temp -XX:OmitStackTraceInFastThrow -add-opens=java.base/java.util=ALL-UNNAMED -add-opens=java.base/java.lang=ALL-UNNAMED -add-opens=java.base/java.io=ALL-UNNAMED -add-opens=java.base/jdk.internal.ref=ALL-UNNAMED -add-opens=java.base/java.lang.invoke=ALL-UNNAMED -add-opens=java.base/java.nio.ch=ALL-UNNAMED -add-opens=java.management=internal=ALL-UNNAMED -Xmx512m -Xms128m -jar C:\Users\Priyanshi\Downloads\sonarqube-9.1.0.47736\lib\jdbch2-h2-1.4.199.jar org.sonar.sc
jvm 1 2021.09.29 13:51:42 INFO app[[[o.s.a.scheduler.impl.Process]] is up
jvm 1 2021.09.29 13:51:42 INFO app[[[o.s.a.ProcessLauncherImpl] Launch process[[key='ce', ipcIndex=3, logFileNamePrefix=ce]] from [C:\Users\Priyanshi\Downloads\sonarqube-9.1.0.47736\ C:\Program Files\Java\jdk-11.0.12\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\Users\Priyanshi\Downloads\sonarqube-9.1.0.47736\temp -XX:OmitStackTraceInFastThrow -add-opens=java.base/java.util=ALL-UNNAMED -add-opens=java.base/jdk.internal.ref=ALL-UNNAMED -add-opens=java.management=internal=ALL-UNNAMED -Xmx512m -Xms128m -jar C:\Users\Priyanshi\Downloads\sonarqube-9.1.0.47736\lib\jdbch2-h2-1.4.199.jar org.sonar.ce.app.CeServer C:\Users\Priyanshi\Downloads\sonarqube-9.1.0.47736\temp\sq-process-39444487414319503
jvm 1 2021.09.29 13:51:42 WARN app[[[startUp] Default administrator credentials are still being used. Make sure to change the password or deactivate the account.
jvm 1 2021.09.29 13:51:42 INFO app[[[startUp]
jvm 1 2021.09.29 13:51:46 INFO app[[[o.s.a.scheduler.impl.Process]] is up
jvm 1 2021.09.29 13:51:46 INFO app[[[o.s.a.scheduler.impl.SonarQube] is up

```

4) Open another command prompt. Run command:

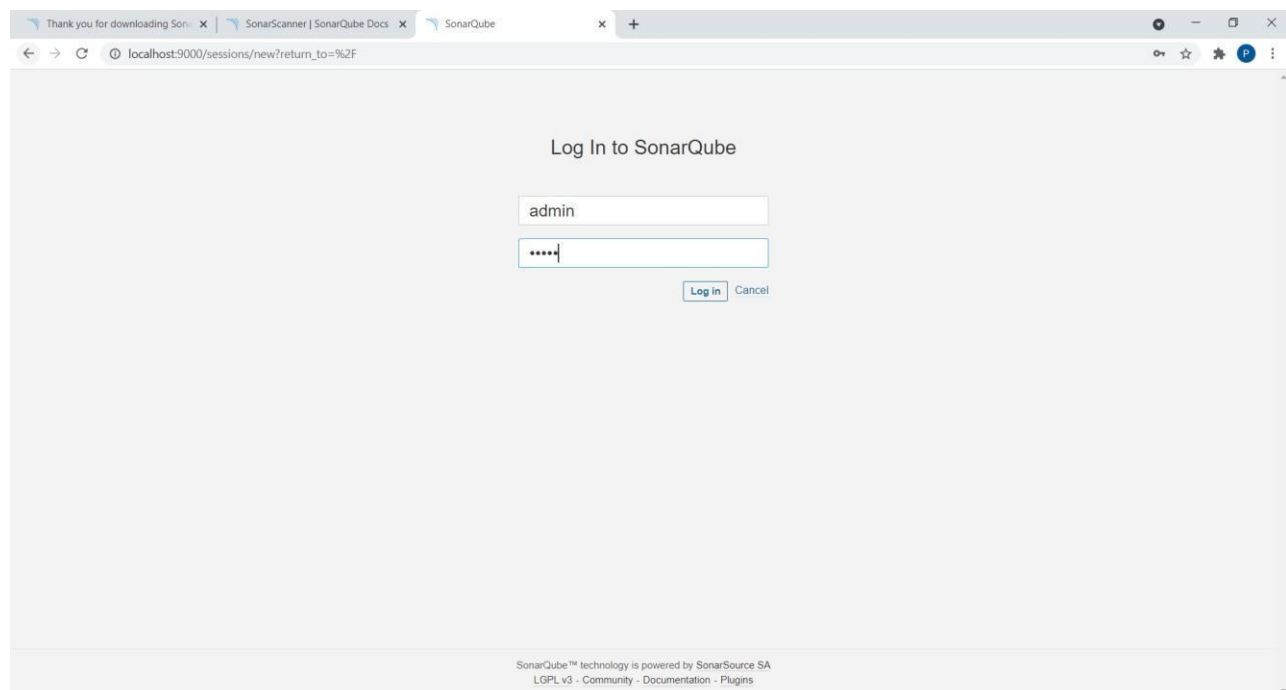
- `cd "sonar-scanner-4.6.2.2472-windows\bin"`
- `sonar-scanner`

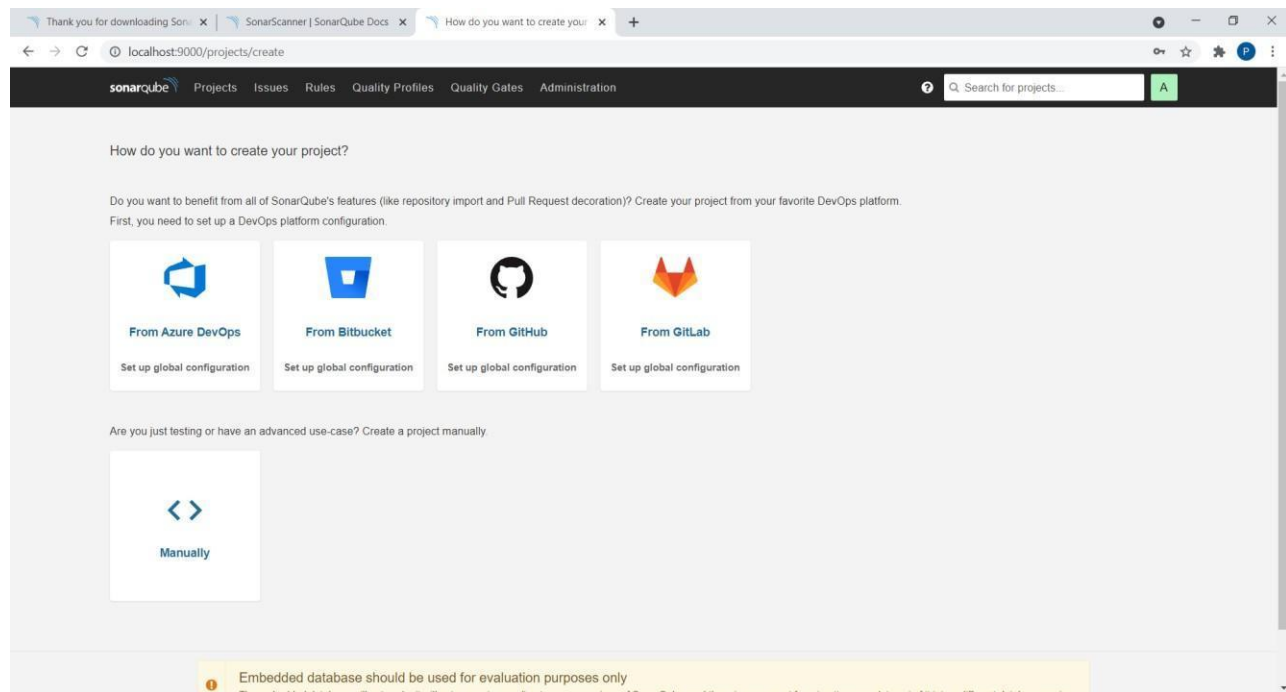
```
Command Prompt
C:\Users\Priyansi\Downloads\sonar-scanner-4.6.2.2472-windows\bin>sonar-scanner
INFO: Scanner configuration file: C:\Users\Priyansi\Downloads\sonar-scanner-4.6.2.2472-windows\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: SonarScanner 4.6.2.2472
INFO: Java 11.0.11 AdoptOpenJDK (64-bit)
INFO: Windows 10 10.0 amd64
INFO: User cache: C:\Users\Priyansi\.sonar\cache
INFO: Scanner configuration file: C:\Users\Priyansi\Downloads\sonar-scanner-4.6.2.2472-windows\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: Analyzing on SonarQube server 9.1.0
INFO: Default locale: "en_IN", source code encoding: "windows-1252" (analysis is platform dependent)
INFO: Load global settings
INFO: -----
INFO: EXECUTION FAILURE
INFO: -----
INFO: Total time: 3.958s
INFO: Final Memory: 5M/20M
INFO: -----
ERROR: Error during SonarScanner execution
ERROR: Not authorized. Analyzing this project requires authentication. Please provide a user token in sonar.login or other credentials in sonar.login and sonar.password.
ERROR:
ERROR: Re-run SonarScanner using the -X switch to enable full debug logging.

C:\Users\Priyansi\Downloads\sonar-scanner-4.6.2.2472-windows\bin>
```

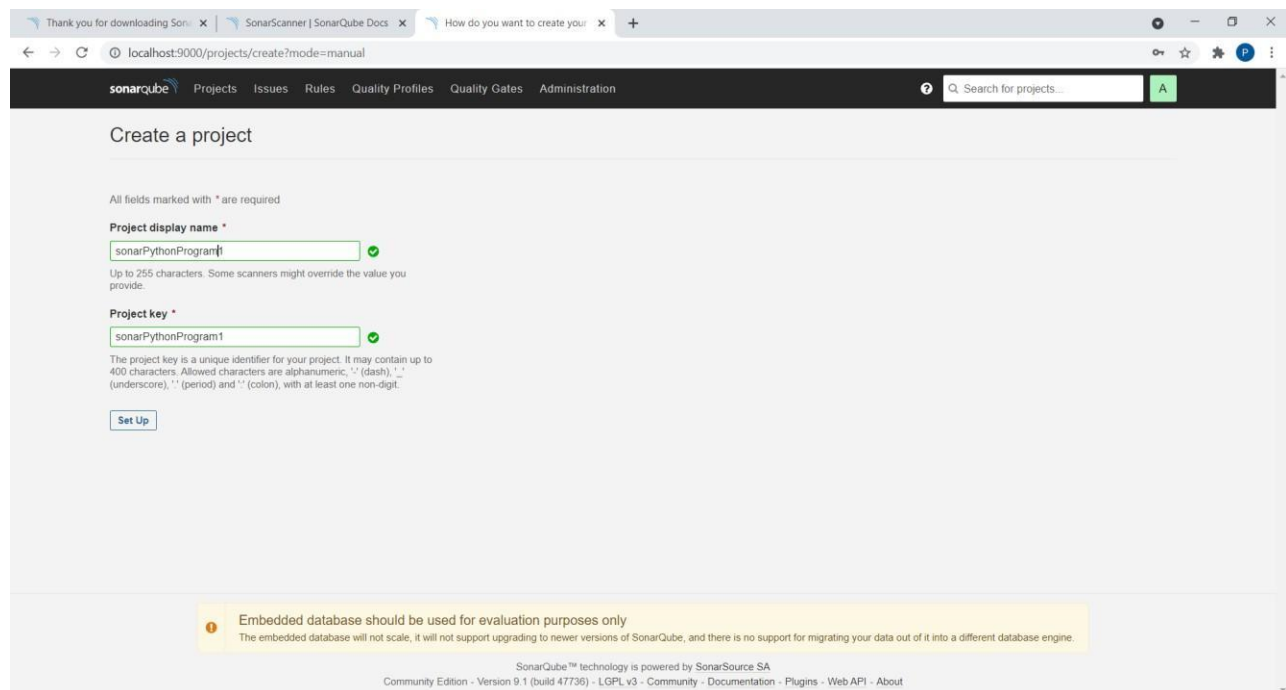
5) Server up and running on localhost:9000

Login using credentials as User: admin and Password: admin and Set a new password

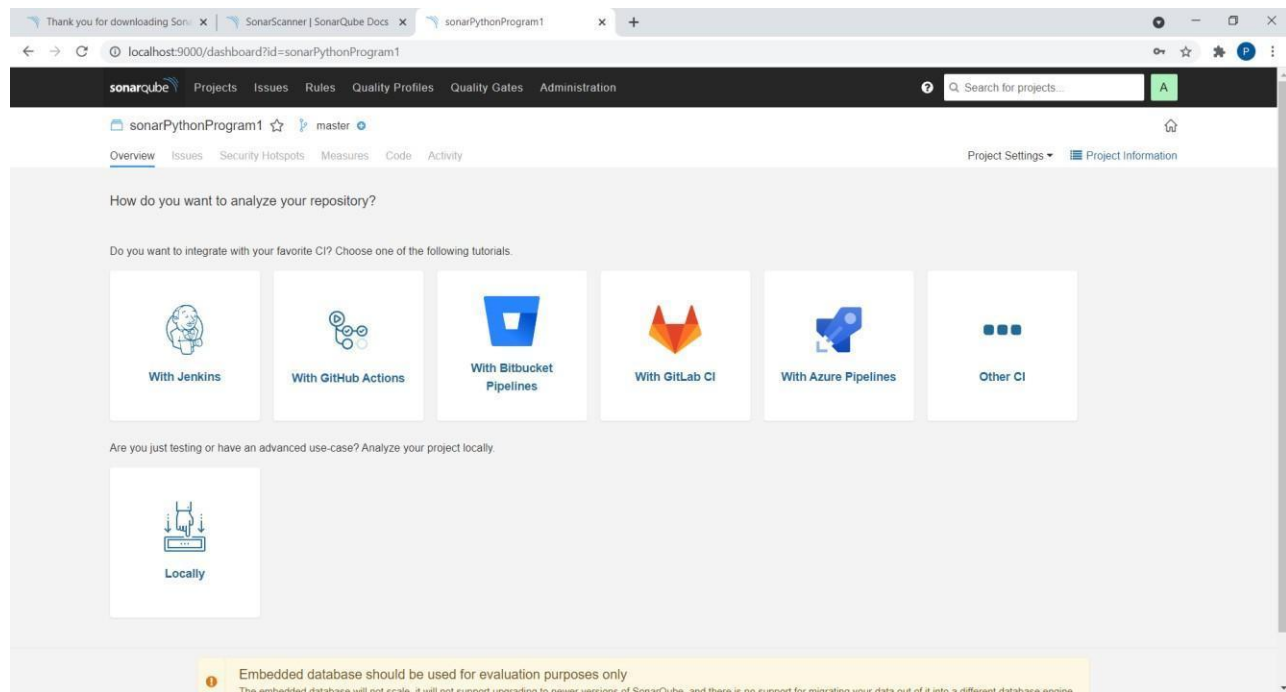




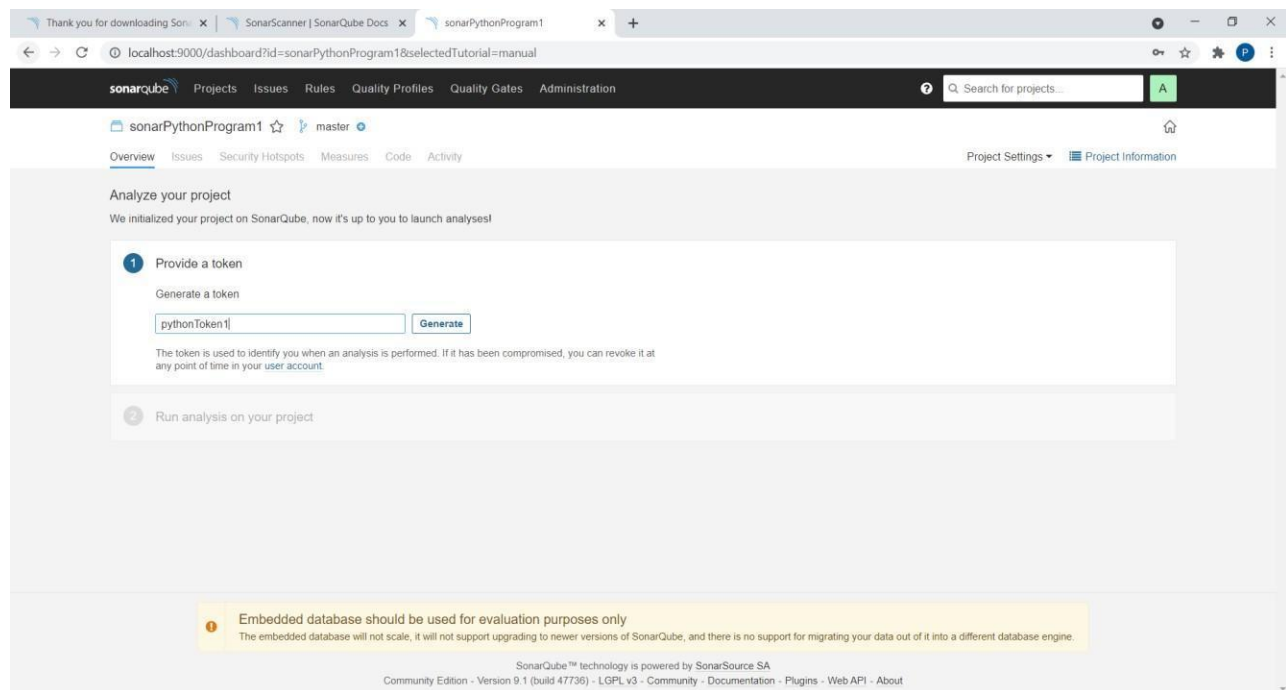
6) Click on Create a project Manually.



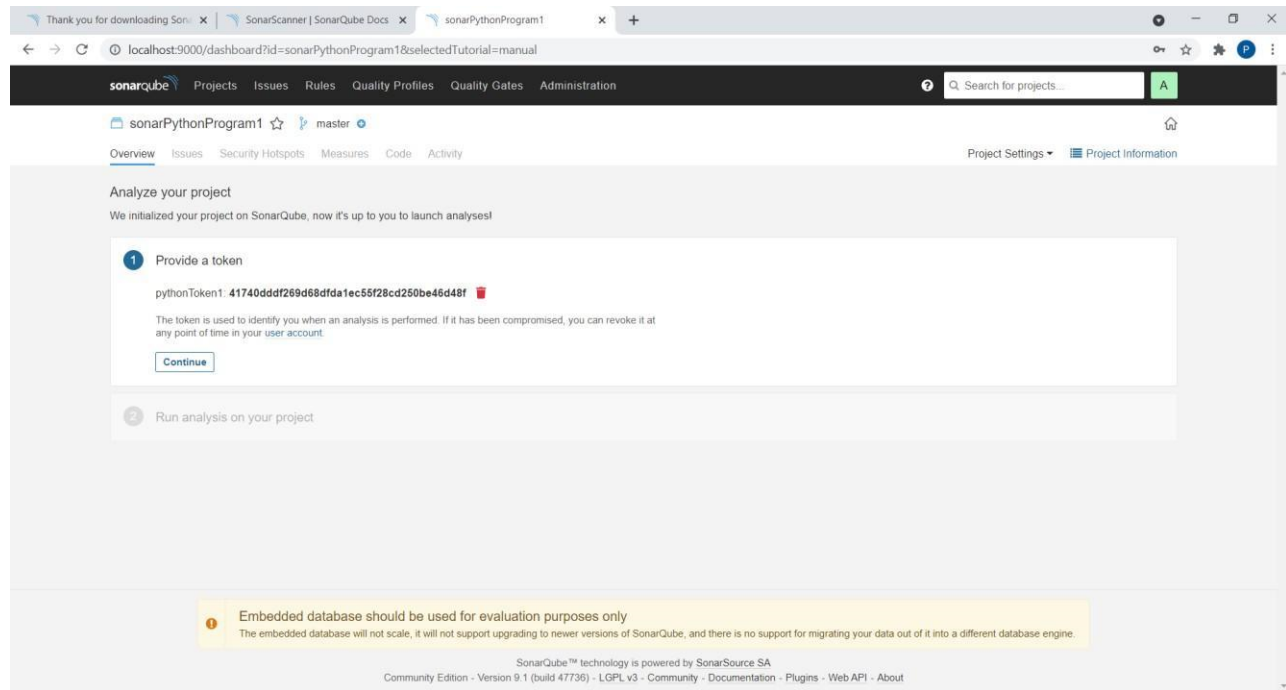
7) Give any Project display name.



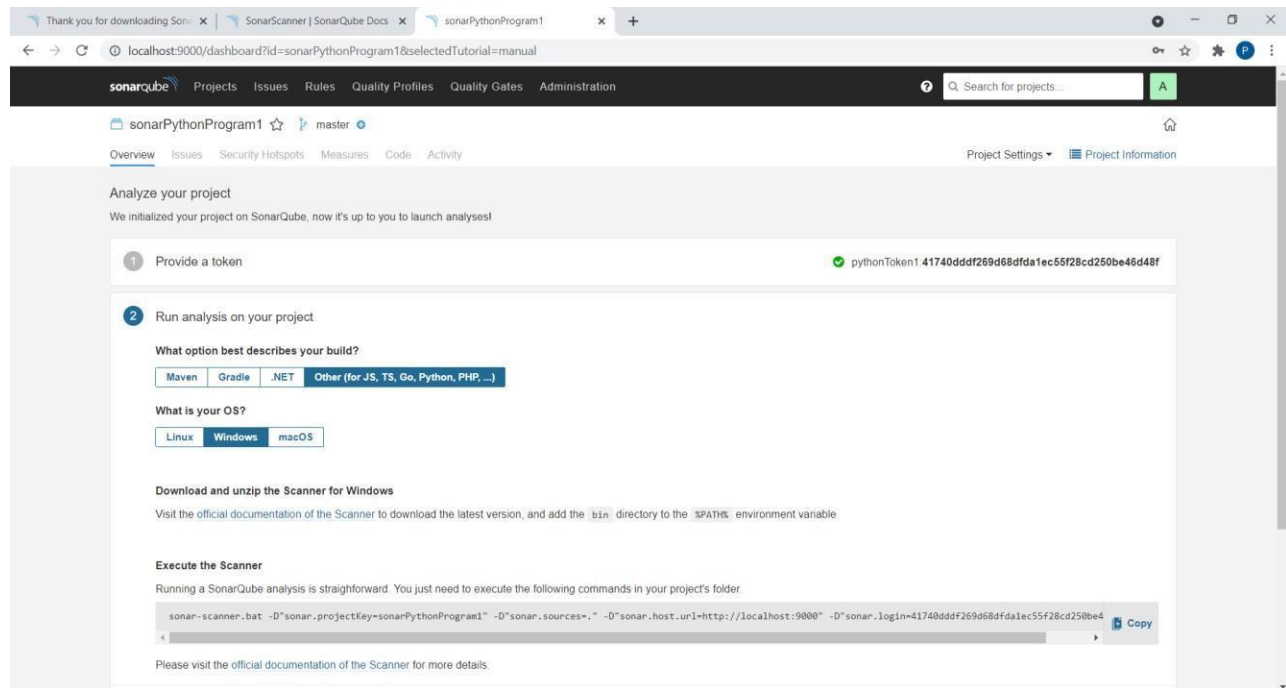
Click on Locally.



9) Give any name to token and click on Generate.



Click on Continue.



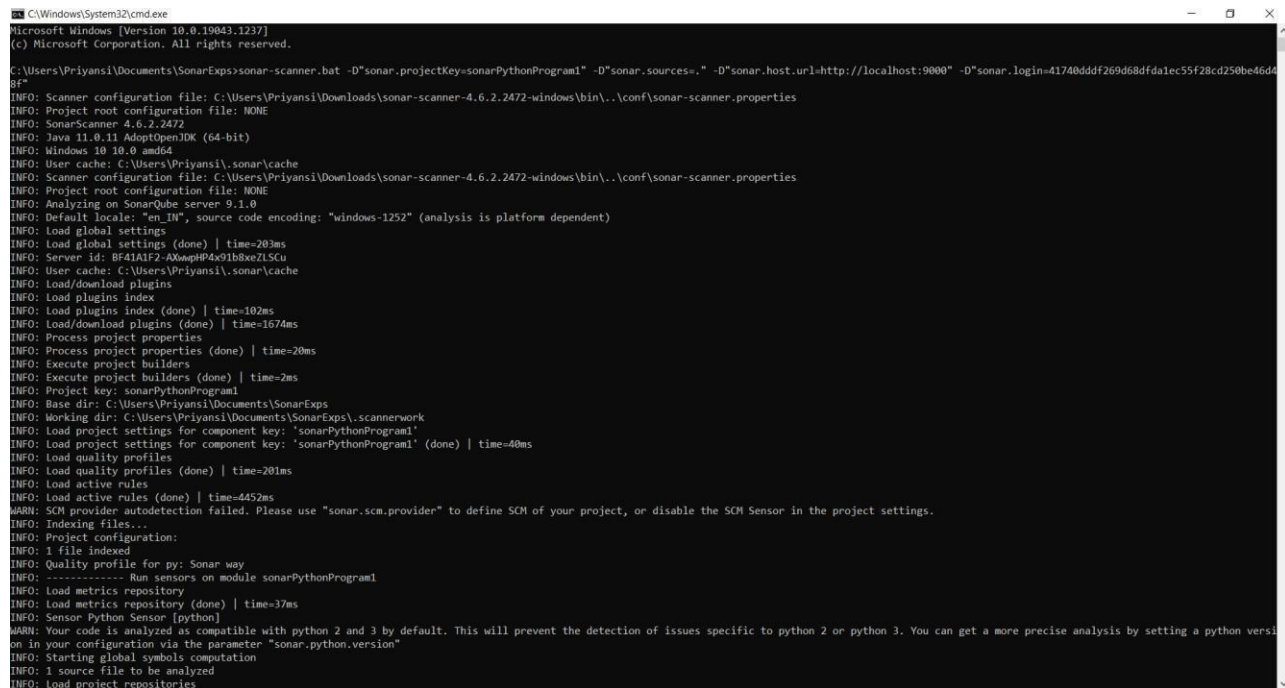
**10) Save a Python program in a folder. class Solution(object):
def romanToInt(self, s):**

```

roman =
{'I':1,'V':5,'X':10,'L':50,'C':100,'D':500,'M':1000,'IV':4,'IX':9,'XL':40,'XC':90,'CD':400,'CM':
900}
i = 0
num = ""
while i < len(s):
    if i+1<len(s) and s[i:i+2] in roman:
        num+=roman[s[i:i+2]]
        i+=2
    else:
        #print(i)
        num+=roman[s[
i]] i+=1
return num
ob1 =
Solution()
print(ob1.romanToInt("III"))
print(ob1.romanToInt("CDXL
III"))

```

11) Open command prompt in this folder and Run program using copied command. “sonar-scanner.bat -D"sonar.projectKey=sonarPythonProgram1" -D"sonar.sources=." -D"sonar.host.url=http://localhost:9000" -D"sonar.login=41740dddf269d68dfda1ec55f28cd250be46d48f"



```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19043.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Priyansi\Documents\SonarExps>sonar-scanner.bat -D"sonar.projectKey=sonarPythonProgram1" -D"sonar.sources=." -D"sonar.host.url=http://localhost:9000" -D"sonar.login=41740dddf269d68dfda1ec55f28cd250be46d48f"

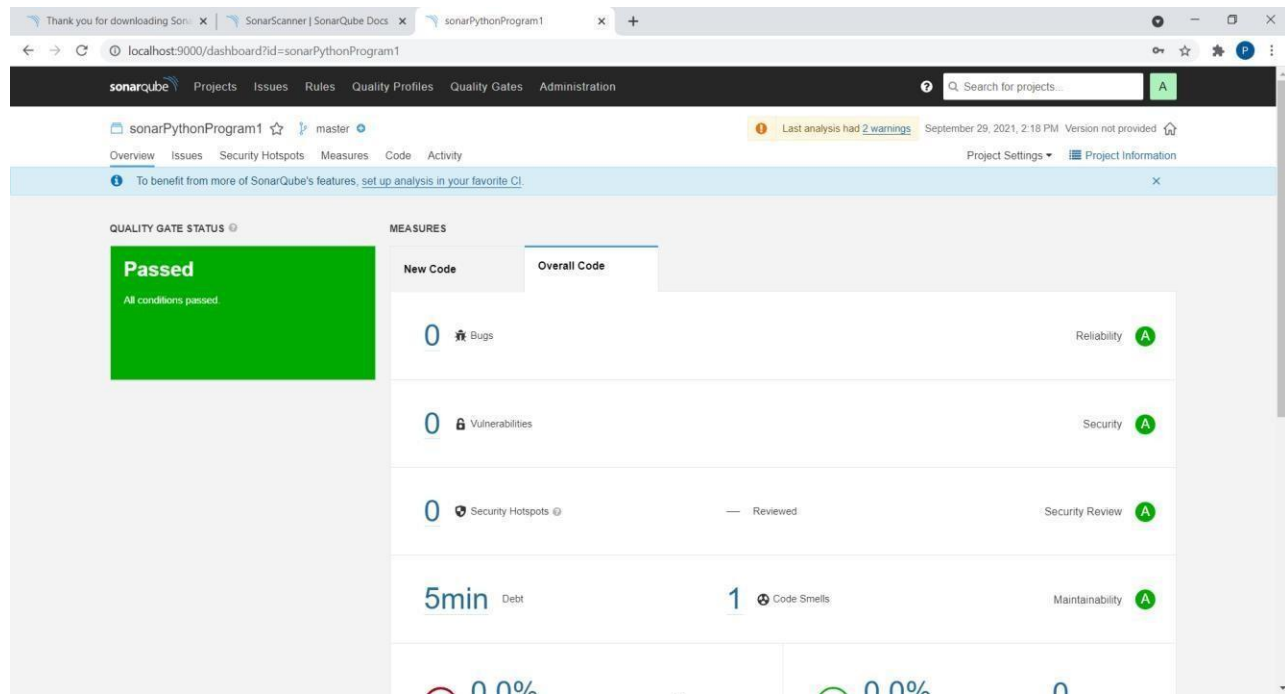
INFO: Scanner configuration file: C:\Users\Priyansi\Downloads\sonar-scanner-4.6.2.2472-windows\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: SonarScanner 4.6.2.2472
INFO: Java 11.0.11 AdoptOpenJDK (64-bit)
INFO: Windows 10 10.0 amd64
INFO: User cache: C:\Users\Priyansi\sonar\cache
INFO: Scanner configuration file: C:\Users\Priyansi\Downloads\sonar-scanner-4.6.2.2472-windows\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: Analyzing on SonarQube server 9.1.0
INFO: Default locale: "en_IN", source code encoding: "windows-1252" (analysis is platform dependent)
INFO: Load global settings
INFO: Load global settings (done) | time=203ms
INFO: Server id: 8f41a1f2-40a4p4x01b8xe7L5Cu
INFO: User cache: C:\Users\Priyansi\sonar\cache
INFO: Load/download plugins
INFO: Load plugins index
INFO: Load plugins index (done) | time=102ms
INFO: Load/download plugins (done) | time=1674ms
INFO: Process project properties
INFO: Process project properties (done) | time=20ms
INFO: Execute project builders
INFO: Execute project builders (done) | time=2ms
INFO: Project key: sonarPythonProgram1
INFO: Base dir: C:\Users\Priyansi\Documents\SonarExps
INFO: Working dir: C:\Users\Priyansi\Documents\SonarExps\scannerwork
INFO: Load project settings for component key: 'sonarPythonProgram1' (done) | time=40ms
INFO: Load quality profiles
INFO: Load quality profiles (done) | time=201ms
INFO: Load active rules
INFO: Load active rules (done) | time=4452ms
WARN: SCM provider autodetection failed. Please use "sonar.scm.provider" to define SCM of your project, or disable the SCM Sensor in the project settings.
INFO: Indexing files...
INFO: Project configuration:
INFO: 1 file indexed
INFO: Quality profile for py: Sonar way
INFO: ----- Run sensors on module sonarPythonProgram1
INFO: Load metrics repository
INFO: Load metrics repository (done) | time=37ms
INFO: Sensor Python Sensor [python]
WARN: Your code is analyzed as compatible with python 2 and 3 by default. This will prevent the detection of issues specific to python 2 or python 3. You can get a more precise analysis by setting a python version in your configuration via the parameter "sonar.python.version"
INFO: Starting global symbols computation
INFO: 1 source file to be analyzed
INFO: Load project repositories

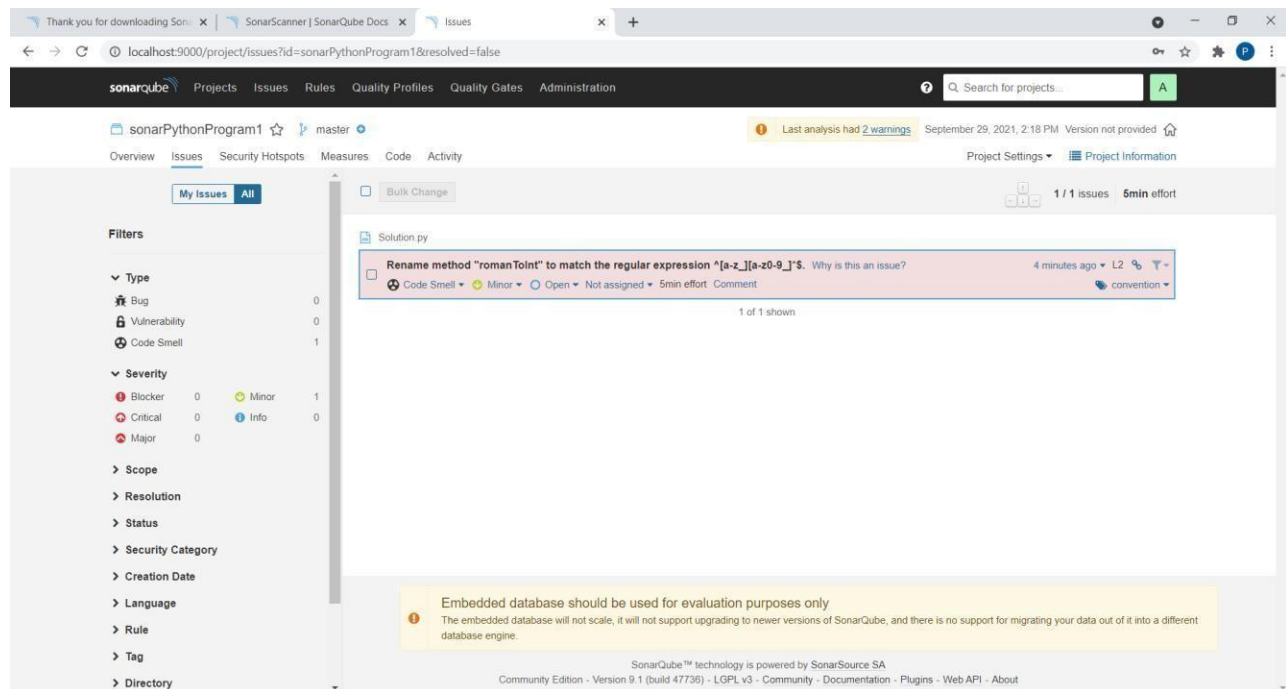
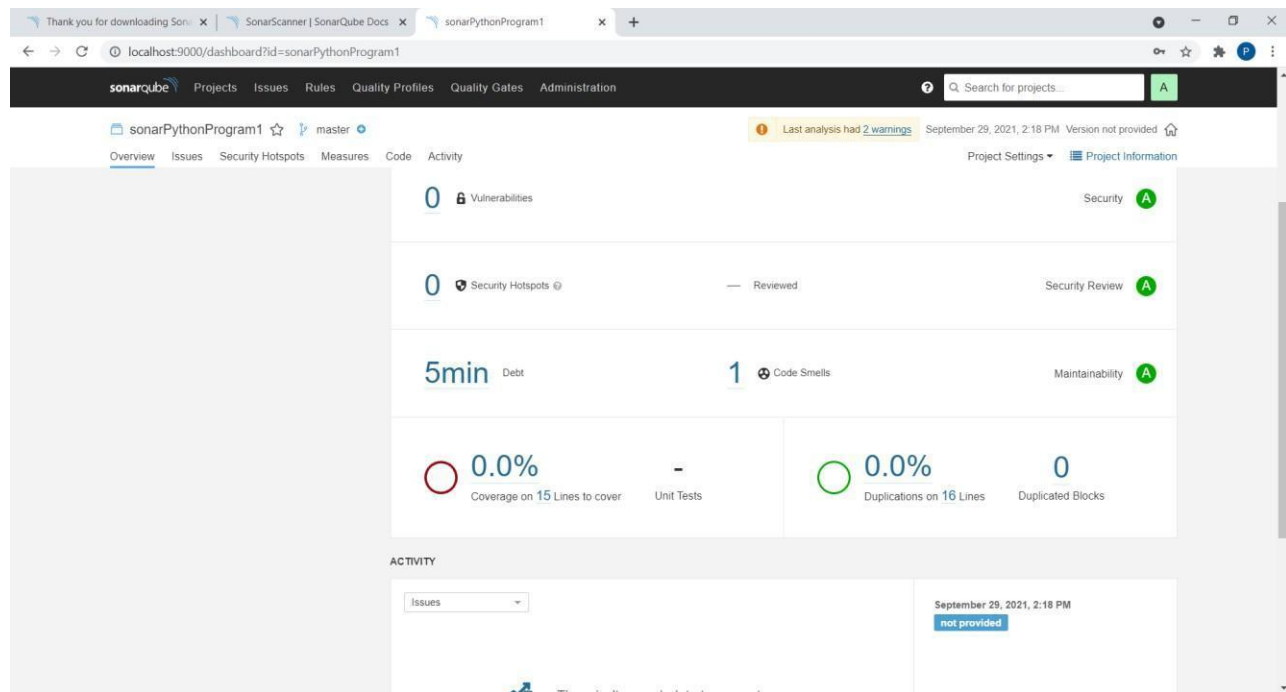
```



```
C:\Windows\System32\cmd.exe
INFO: Sensor HTML [web] (done) | time=2ms
INFO: Sensor VB.NET Project Type Information [vbnet]
INFO: Sensor VB.NET Project Type Information [vbnet] (done) | time=1ms
INFO: Sensor VB.NET Analysis Log [vbnet]
INFO: Sensor VB.NET Analysis Log [vbnet] (done) | time=12ms
INFO: Sensor VB.NET Properties [vbnet]
INFO: Sensor VB.NET Properties [vbnet] (done) | time=0ms
INFO: ----- Run sensors on project
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=12ms
INFO: SCM Publisher No SCM system was detected. You can use the 'sonar.scm.provider' property to explicitly specify it.
INFO: CPD Executor Calculating CPD for 1 file
INFO: CPD Executor CPD calculation finished (done) | time=10ms
INFO: Analysis report generated in 59ms, dir size=103.9 kB
INFO: Analysis report compressed in 19ms, zip size=14.7 kB
INFO: Analysis report uploaded in 76ms
INFO: ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard?id=sonarPythonProgram1
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
INFO: More about the report processing at http://localhost:9000/api/task?id=AWwwV1hx91b8xeZLX8H1
INFO: Analysis total time: 7.502 s
INFO: -----
INFO: EXECUTION SUCCESS
INFO: -----
INFO: Total time: 10.887s
INFO: Final Memory: 79M/30M
INFO: -----
C:\Users\Priyansi\Documents\SonarExps>
```

13) Given below is the inspection of code quality to perform automatic reviews with static analysis of code to detect bugs, code smells, and security vulnerabilities.





Press “Ctrl + C” to stop the server.

