

Stackademic · [Follow publication](#)



Great choice. Enjoy your free member-only read.
[Upgrade to read anything on Medium.](#)

[Open in app](#) ↗

≡ **Medium**



Integration of SonarQube with SpringBoot

5 min read · Jan 15, 2024



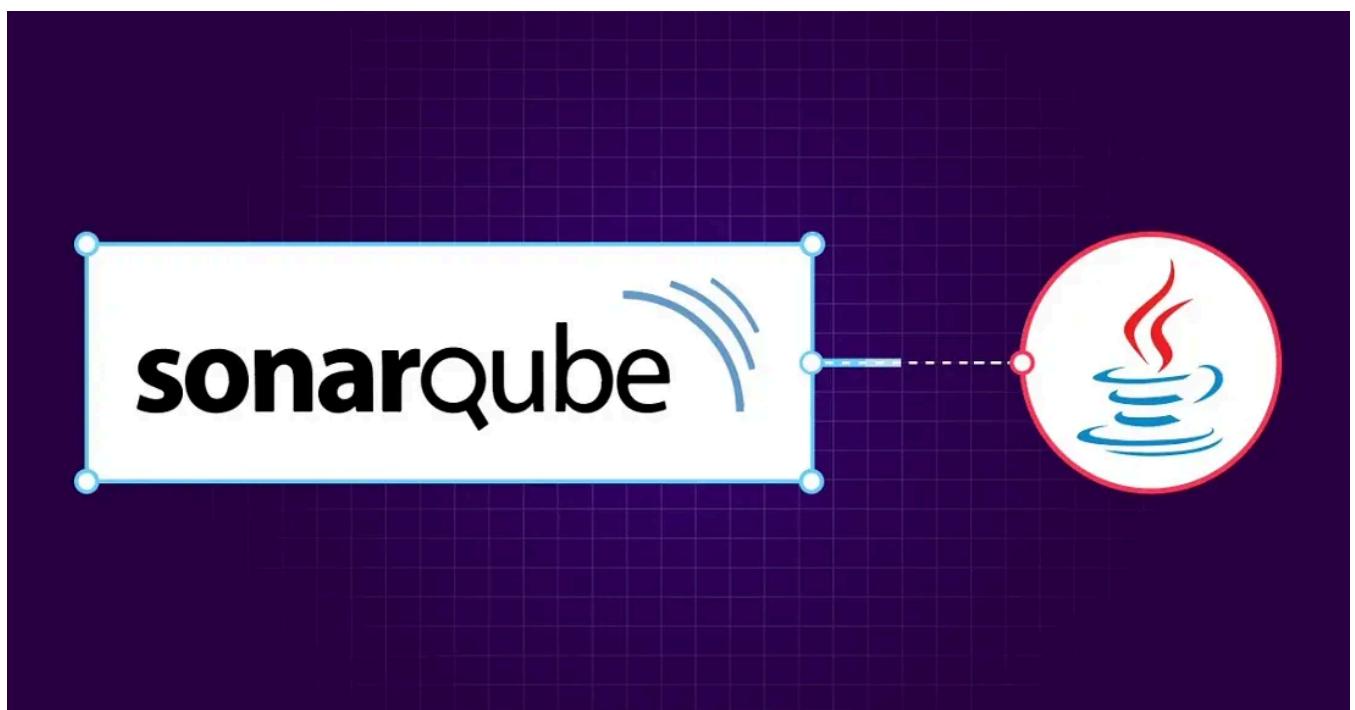
Anusha SP

[Follow](#)

Listen

Share

More



Before knowing how to integrate SonarQube with SpringBoot, let us understand, What is SonarQube? Why it is required ?

What is SonarQube ?

SonarQube formerly known as Sonar, is an open-source platform developed by SonarSource for continuous inspection of code-quality. To perform automatic reviews with static analysis of code to detect bugs or and code smells of around 29 programming languages.

It performs automated code analysis and provides detailed reports on code quality, code coverage, and various code-related issues, including bugs, vulnerabilities and also about code smells.

Why SonarQube ?

There are many benefits of using the SonarQube database for code quality. If you want to empower developers to write more robust and resilient source code, the SonarQube database can help in the following ways:

- Improve quality
- Grow developer skills
- Continuous quality management
- Reduce risk
- Tracking issues
- Provides Code Quality Metrics
- Integration

Integration of SonarQube in SpringBoot applications

Spring boot is a Java based framework that is used to develop web services. It provided many functionalities like Autowire, component scanner, Rest controller, and Microservice environment to develop the web service faster and easier. It also provides stand-alone and production-ready spring applications.

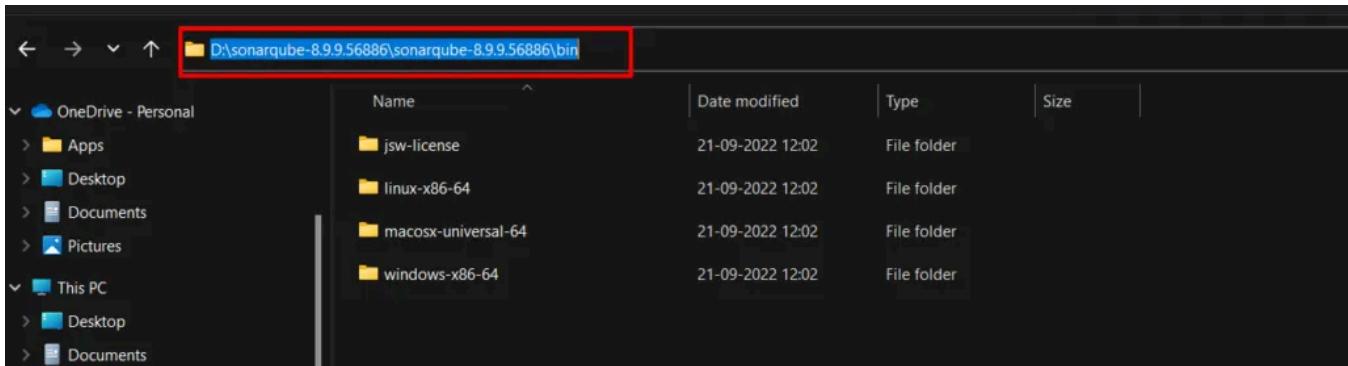
Requirements

- JDK > Java 8
- Maven > 8.6
- SonarScanner

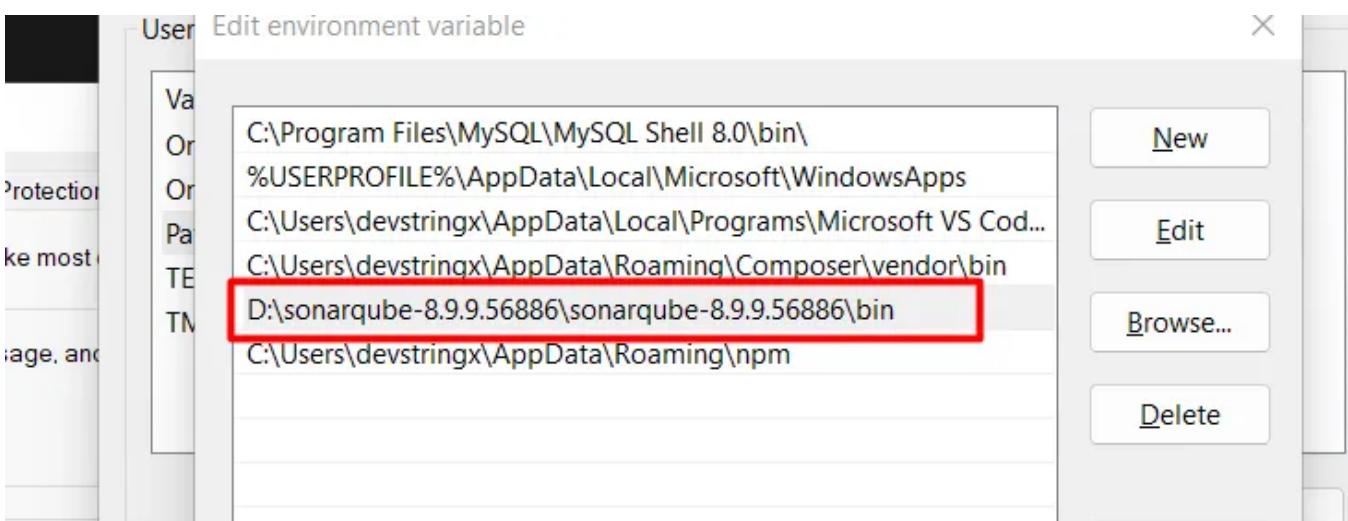
- SonarQube

1. Install and Set Up SonarQube:

- Download and install SonarQube on your server or as a Docker container.
- Download Sonar Qube and extract it to a directory. Copy the path of the bin folder and add this also in the env variable.

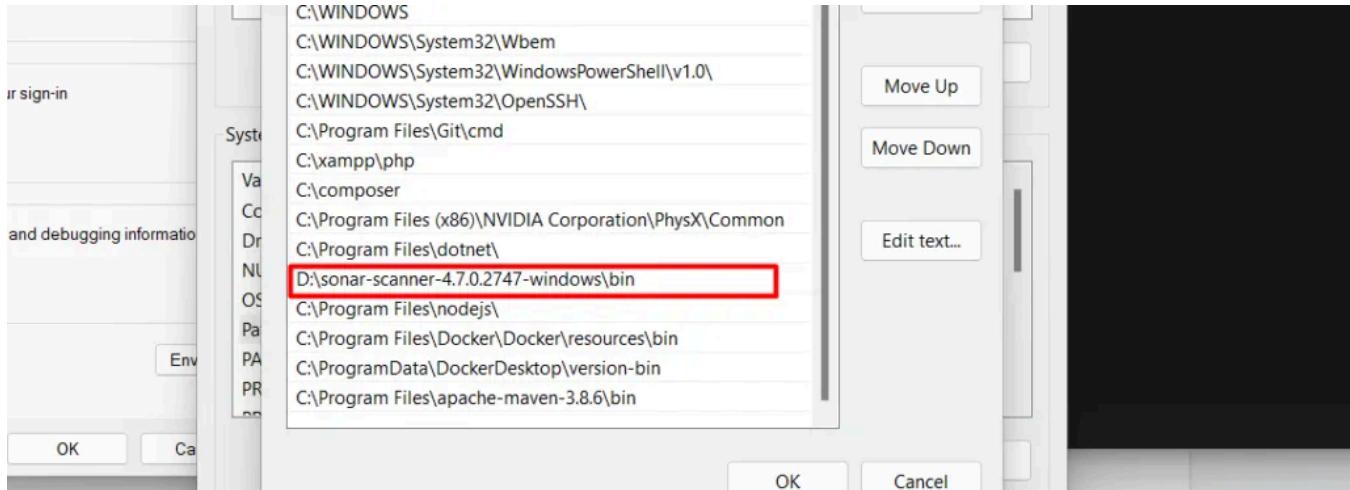


- Add this path to the environment variable



2. Setup SonarScanner

- Download the sonar scanner and unzip it and add the path in the environment variable



- Now you have completed the setup of a third party application so we can proceed with the spring boot application.
- You can find installation instructions on the [official SonarQube Website](#)
- Start the SonarQube server and access the web interface to configure projects and obtain an authentication token.

3. Create a Spring Boot Application With Spring Initializr

- Add the SonarQube Scanner plugin to your project.
- For a Maven-based Spring project, add the following plugin configuration to your `pom.xml`

```
<build>
  <plugins>
    <plugin>
      <groupId>org.sonarsource.scanner.maven</groupId>
      <artifactId>sonar-maven-plugin</artifactId>
      <version>3.9.0.2155</version> <!-- Replace with the latest version -->
    </plugin>
  </plugins>
</build>
```

4. Configure SonarQube properties in your project.

- Add a `sonar-project.properties` file in your project's root directory with the necessary properties, including project key, project name, and SonarQube

server URL.

- For example:

```
sonar.projectKey=my-spring-project
sonar.projectName=My Spring Project
sonar.host.url=http://localhost:9000
```

5. Create a Project in Sonarqube

- After adding the dependencies install the maven dependencies in your application.
- Now we will create a project in Sonarqube.
- So let's start the Sonarqube by running the following file in cmd "StartSonar.bat"

```
D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\bin>cd windows-x86-64
D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\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   | 2022-09-30 12:43:03 INFO app[[o.s.a.AppFileSystem] Cleaning or creating temp directory D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\temp
jvm 1   | 2022-09-30 12:43:03 INFO app[[o.s.a.EsSettings] Elasticsearch listening on [HTTP: 127.0.0.1:9001, TCP: 127.0.0.1:6010]
jvm 1   | 2022-09-30 12:43:04 INFO app[[o.s.a.ProcessLauncherImpl] Launch process[[key='es', ipcIndex=1, logFilenamePrefix=es]] from [D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\elasticsearch]: C:\Program Files\Java\jdk-11.0.15\bin\java -XX:+UseG1GC -Djava.io.tmpdir=D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\temp -XX:ErrorFile=..logs/es_hs_err.pid%log -Des.networkaddress.cache.ttl=60 -Des.networkaddress.ca
che-negative.ttl=10 -XX:+AlwaysPreTouch -Xss1m -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.nosys=true -Djava.tmpdir=D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\temp -XX:-OmitStackTraceInFastThrow -Dio.netty.unsafe=true -Dio.netty.setOptimization=true -Dio.netty.recycler.maxCapacityPerThread=4 -Dio.netty.allocator.numDirectArenas=0 -Dlog4j.shutdownHookEnabled=false -Dlog4j2.disable.jmx=true -Dlog4j2.f
rmatMsgNoLookups=true -Djava.locale.providers=COMPAT -Xmx512m -Xms512m -XX:MaxDirectMemorySize=256m -XX:+HeapDumpOnOutOfMemoryError -Delasticsearch -Des.path.home=D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\el
asticsearch -Des.path.conf=D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\temp\conf\es -cp lib/* org.elasticsearch.bootstrap.BootstrapElasticsearch
jvm 1   | 2022-09-30 12:43:52 INFO app[[o.s.a.SchedulerImpl] Process[es] is up
jvm 1   | 2022-09-30 12:43:52 INFO app[[o.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running
jvm 1   | 2022-09-30 12:43:52 INFO app[[o.s.a.ProcessLauncherImpl] Launch process[[key='web', ipcIndex=2, logFilenamePrefix=web]] from [D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886]: C:\Program Files\Java\j
dk-11.0.15\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\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.rmi=java.rmi.transport=ALL-UNNAMED -Dhttp.nonProxyHosts=loc
alhost|127.*|*:1 -cp ./lib/sonar-application-8.9.9.56886.jar;D:\sonarqube-8.9.9.56886\lib\jdbc\h2\1.4.199.jar org.sonar.server.app.WebServer D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56
886\temp\sql\process21384301427273159properties
jvm 1   | 2022-09-30 12:46:01 INFO app[[o.s.a.SchedulerImpl] Process[web] is up
jvm 1   | 2022-09-30 12:46:02 INFO app[[o.s.a.ProcessLauncherImpl] Launch process[[key='ce', ipcIndex=3, logFilenamePrefix=ce]] from [D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886]: C:\Program Files\Java\j
dk-11.0.15\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\temp -XX:-OmitStackTraceInFastThrow -add-opens=java.base/java.util=ALL-UNNAMED -X
mx512m -Xms128m -XX:+HeapDumpOnOutOfMemoryError -Dhttp.nonProxyHosts=localhost|127.*|*:1 -cp ./lib/sonar-application-8.9.9.56886.jar;D:\sonarqube-8.9.9.56886\lib\jdbc\h2\1.4.199.jar or
g.sonar.ce.app.GeServer D:\sonarqube-8.9.9.56886\sonarqube-8.9.9.56886\temp\so-process1431708231776256669properties
jvm 1   | 2022-09-30 12:47:14 INFO app[[o.s.a.SchedulerImpl] Process[ce] is up
jvm 1   | 2022-09-30 12:47:14 INFO app[[o.s.a.SchedulerImpl] SonarQube is up
```

- When your sonar Qube up visits this URL in the browser <http://localhost:9000/>
- The first-time password for the sonarqube will be the
- Username: admin and Password: admin
- Reset this password just after login

6. Now create a project

- Click on add project and select manually

- Add your project name and click setup

All fields marked with * are required

Project key * 

Up to 400 characters. Allowed characters are alphanumeric, '-' (dash), '_' (underscore), '.' (period) and ':' (colon), with at least one non-digit.

Display name * 

Up to 255 characters

Set Up

- Add the project name in the token and click generate

We initialized your project on SonarQube, now it's up to you to launch analyses!

1 Provide a token

Generate a token

Generate

Use existing token

The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point of time in your user account.

- Then click the maven tab and you will see the maven command like below

2 Run analysis on your project

What option best describes your build?

Maven Gradle .NET Other (for JS, TS, Go, Python, PHP, ...)

Execute the Scanner for Maven from your computer

Running a SonarQube analysis with Maven is straightforward. You just need to run the following command in your project's folder

```
mvn sonar:sonar \
-Dsonar.projectKey=myproject12 \
-Dsonar.host.url=http://localhost:9000 \
-Dsonar.login=400624b3acb88fc4b1ce3dc880fffb88876fa5ccc
```

Copy

Please visit the official documentation of the Scanner for Maven for more details.

Once the analysis is completed, this page will automatically refresh and you will be able to browse the analysis results.

- Edit this command and remove the extra line break and forward slash

```
mvn sonar:sonar -Dsonar.projectKey=myproject12 -Dsonar.host.url=http://localhost:9000 -Dsonar.login=400624b3acb88fc4b1ce3dc880fffb88876fa5ccc
```

- Run the command in cmd at the root of your project

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Microsoft Windows [Version 10.0.22000.978]
(c) Microsoft Corporation. All rights reserved.

D:\java\New folder\demo>mvn sonar:sonar -Dsonar.projectKey=myproject12 -Dsonar.host.url=http://localhost:9000 -Dsonar.login=400624b3acb88fc4b1ce3dc880ff80876fa5ccc
[INFO] Scanning for projects...
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/3.2.0/maven-clean-plugin-3.2.0.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/3.2.0/maven-clean-plugin-3.2.0.pom (5.3 kB at 3.1 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/35/maven-plugins-35.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/35/maven-plugins-35.pom (9.9 kB at 21 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/35/maven-parent-35.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/35/maven-parent-35.pom (45 kB at 67 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/apache/25/apache-25.pom (21 kB at 43 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/apache/25/apache-25.pom
[INFO] Analysis report uploaded in 639ms
[INFO] ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard?id=myproject12
[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/ce/task?id=AYONW4OU34Q3dk0GtnQ4
[INFO] Analysis total time: 53.632 s
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 02:16 min
[INFO] Finished at: 2022-09-30T15:09:04+05:30
[INFO] -----

```

- Now refresh the sonar dashboard

The screenshot shows the SonarQube dashboard for the project 'myproject12'. The top navigation bar includes links for PHP, Upscale, Oracle, codeGraffer, Devstringx, easyagile.com, JSON Grid, Spring, and initialization. The main header displays the project name 'sonarqube' and the current branch 'master'. A search bar allows searching for projects. The dashboard features a 'QUALITY GATE STATUS' section showing a green 'Passed' status with the message 'All conditions passed'. Below this, the 'MEASURES' section provides detailed metrics: 0 Bugs (Reliability A), 0 Vulnerabilities (Security A), 0 Security Hotspots (Reviewed, Security Review A), 1 Code Smells (Maintainability A), 0.0% Coverage (Coverage on 2 Lines to cover, Unit Tests), 0.0% Duplications (Duplications on 71 Lines, Duplicated Blocks), and 0 Duplicated Blocks.

7. You can also run using SonarQube Analysis Command:

- Execute the SonarQube analysis by running the following command in your project's directory:

```
mvn sonar:sonar
```

- Replace **mvn** with **gradle** if you are using Gradle for your project.

8. View SonarQube Reports:

- Access the SonarQube web interface using <http://localhost:9000> to view detailed code analysis reports for your Spring project.

9. Set Up Continuous Integration:

- Configure your CI/CD pipeline (e.g., Jenkins, Travis CI, GitLab CI) to automatically run SonarQube analysis on code commits and pull requests.

I hope this article is helpful. Thank you for reading the article. Please clap, share and comment. it will encourage me to write more such articles. Do share your valuable suggestions, I appreciate your honest feedback!

Stackademic

Thank you for reading until the end. Before you go:

- Please consider clapping and following the writer! 🙌
- Follow us [X](#) | [LinkedIn](#) | [YouTube](#) | [Discord](#)
- Visit our other platforms: [In Plain English](#) | [CoFeed](#) | [Venture](#)

[Sonarqube](#)[Spring](#)[Java](#)[Code Review](#)[Bugs](#)[Follow](#)

Published in Stackademic

73K followers · Last published 10 hours ago

Stackademic is a learning hub for programmers, devs, coders, and engineers. Our goal is to democratize free coding education for the world.

[Follow](#)

Written by Anusha SP

2.1K followers · 3 following

I write what I read and practice !!!

More from Anusha SP and Stackademic



Most Asked
Java 8
Interview
Coding Questions
(Part-1)



In Dev Genius by Anusha SP

Java 8 Coding and Programming Interview Questions and Answers

It has been 8 years since Java 8 was released. I have already shared the Java 8 Interview Questions and Answers and also Java 8 Stream API...

Jan 31, 2023

1.6K

22



...



In Stackademic by aisynd

The Best Programming Language for 2026 (According to Reality, Not Twitter)

Choosing the best programming language for 2026 is harder than choosing a JavaScript framework—and that's saying something. Every year...

Jan 17 553 33



...

Rue

Exploring memory safety that's easier to use

Get Started Read the Spec

Early Stage Familiar Syntax Native Compilation

In Stackademic by Deo Wetton

11 Days, 100,000 Lines of Code: A Rust Veteran Teams Up with Claude to Build a New Language from...

"If you have no funding, no team—just one person—can you still build a programming language?"

Jan 15 332 5



...



 Anusha SP

Synechron Interview Questions

Hello everyone! I recently had the opportunity to attend a technical interview at Synechron, and I wanted to share my experience to help...

Aug 27, 2025 12 1



...

See all from Anusha SP

See all from Stackademic

Recommended from Medium

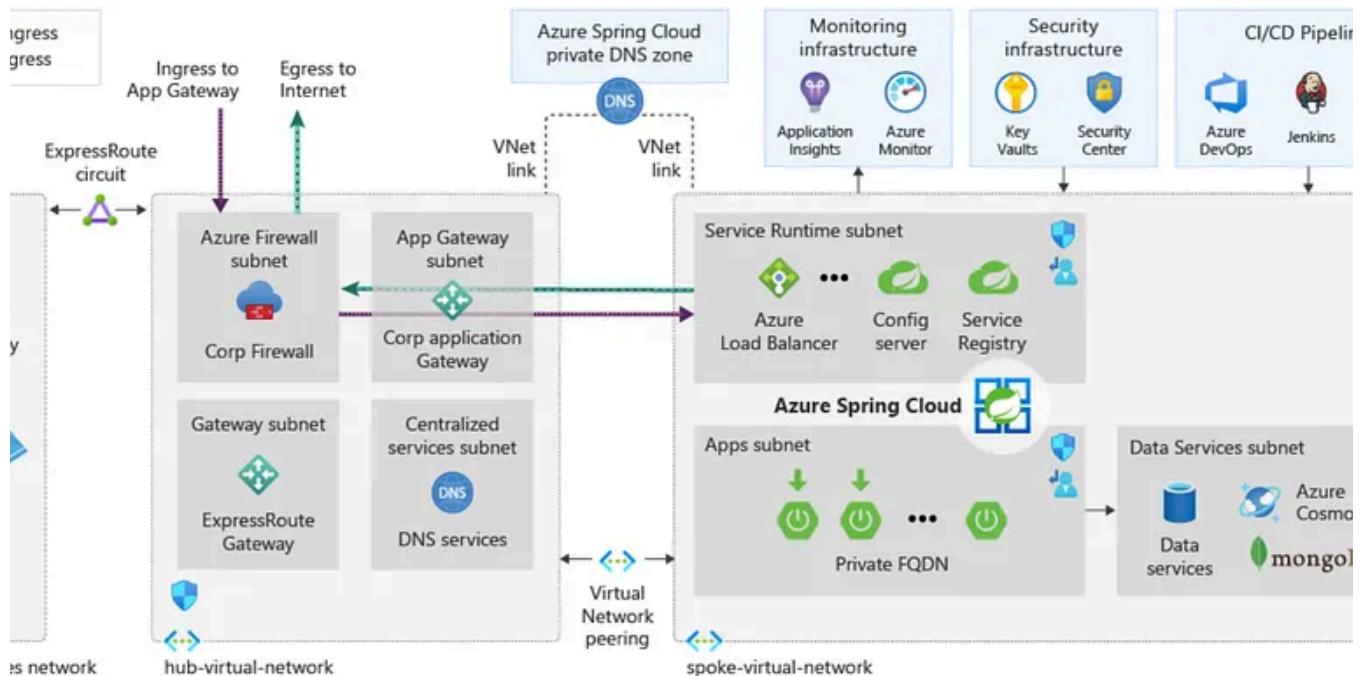


Umesh Kumar Yadav

Token, Session, Cookie, JWT, OAuth2—I can't tell the difference!

Recently, I've noticed that some people easily confuse the concepts of Token, Session, Cookie, JWT, and OAuth2.

Dec 10, 2025 179 2



Jaytech

Game-Changing Java & Spring Boot Trends for 2026

Most important Java & Spring Boot trends for 2026 🚀

Jan 4 403 5



Java 21+ Virtual Threads

Handling 100K Requests

with Java Virtual Threads

⚡ Ultra Lightweight • 🚦 High Throughput • 💡 Scalable • 🎯 Project Loom

```
Executors.newVirtualThreadPerTaskExecutor()
```

In @Override by Ahmet Emre DEMİRŞEN

Handling 100k Requests with Java Virtual Threads

Have you ever wondered how your Java application would fare under a sudden surge of 100,000 concurrent requests? The thought alone can send...

3d ago 115 1

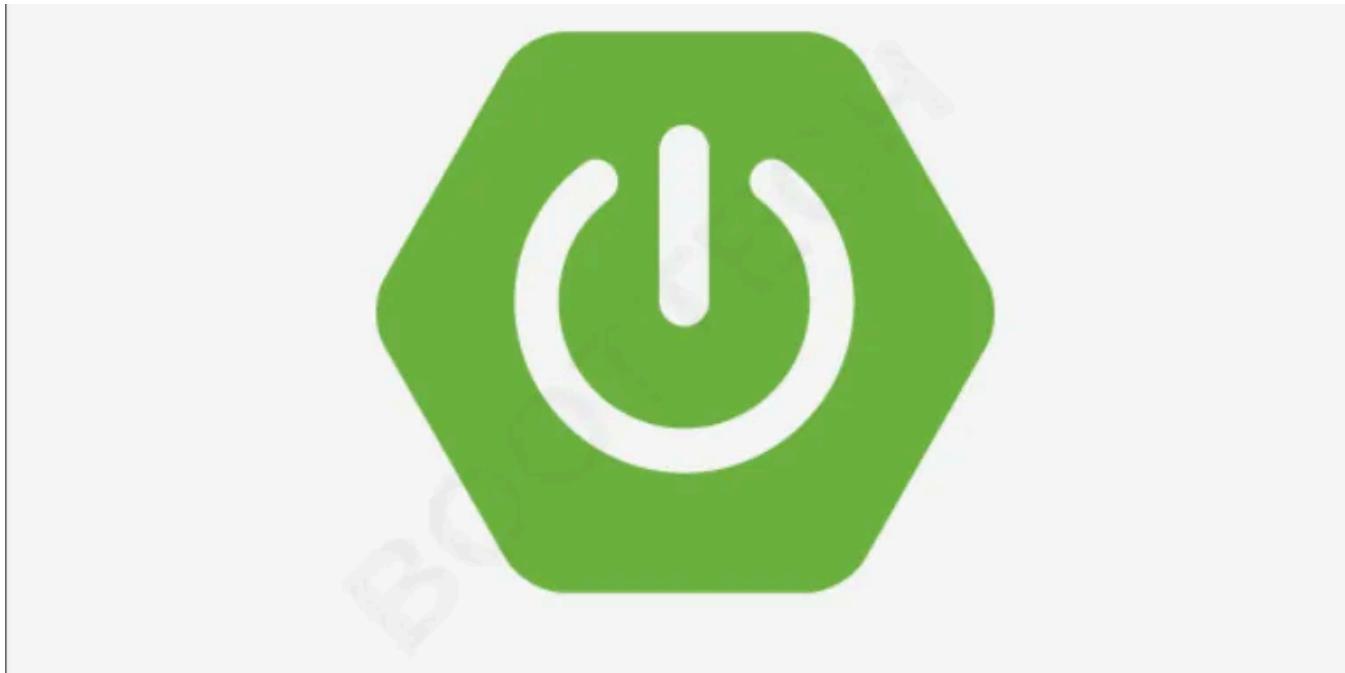


Code With Sunil | Code Smarter, not harder

10 Real-World Spring Boot Architecture Tips Every Developer Should Know

If you are not a Member — Read for free here

2d ago 132

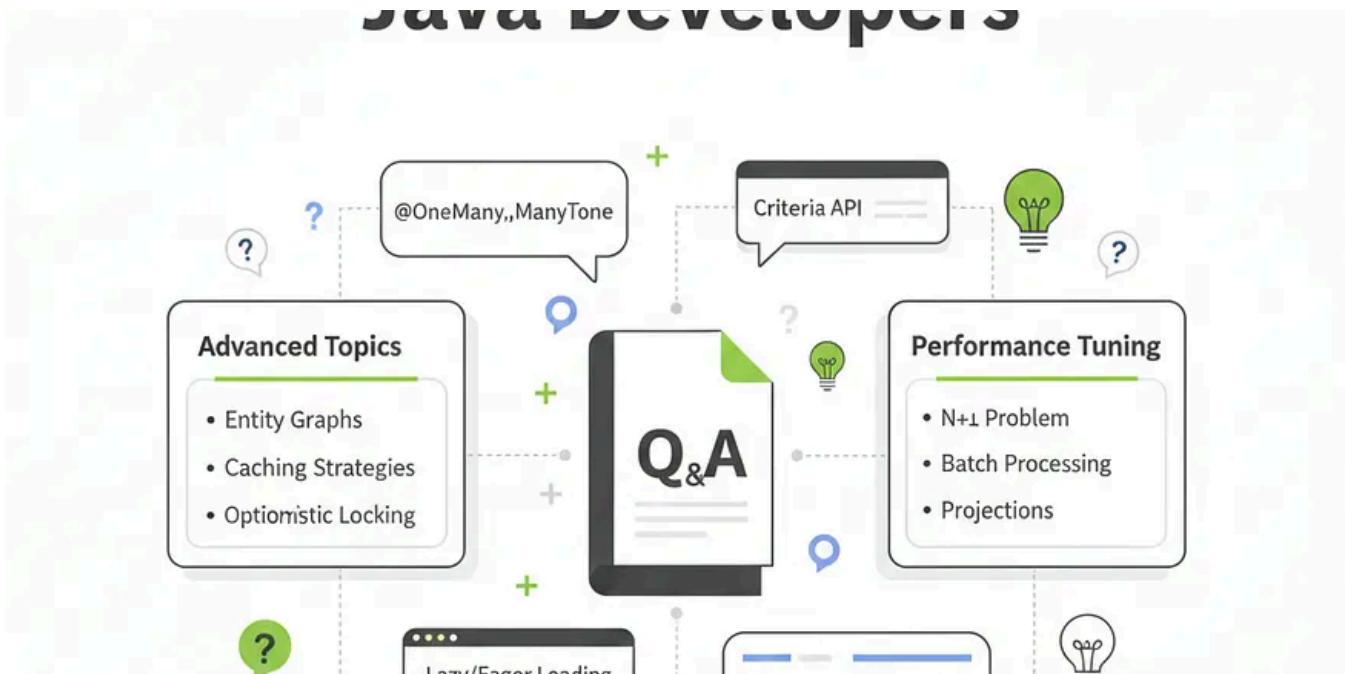


Eric Anicet

Building Robust API Clients with RestClient in Spring Boot 3.X

In this story, we'll explore how to use the RestClient class in Spring Boot 3 to perform GET, POST, PUT, and DELETE operations against REST...

Oct 6, 2025 1





JPA Interview Questions and Answer for Experienced Java Developers — Part -1

JPA Interview Questions and Answer

star Jan 6

wave 8



...

See more recommendations