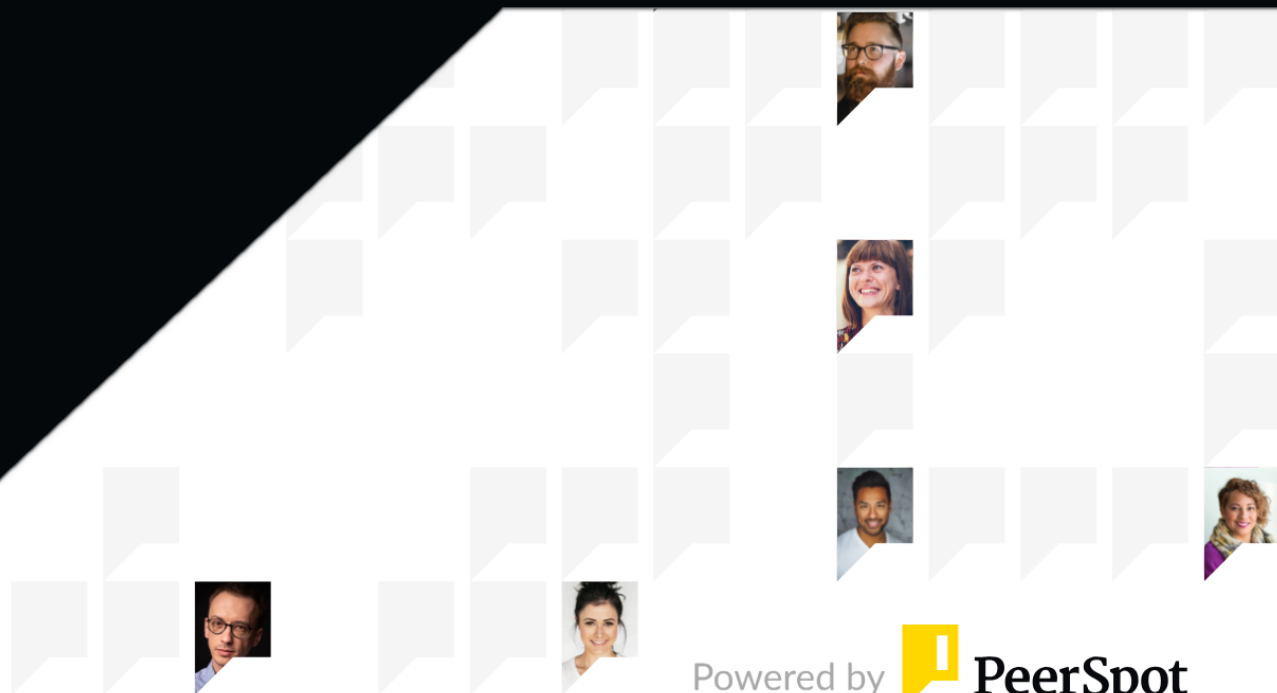


aws marketplace

OpenJDK Java

Reviews, tips, and advice from real users



Powered by  PeerSpot

Contents

Product Recap..... 3 - 4

Valuable Features..... 5 - 12

Other Solutions Considered..... 13 - 15

ROI..... 16 - 18

Use Case..... 19 - 22

Setup..... 23 - 24

Customer Service and Support..... 25 - 26

Other Advice..... 27 - 32

Trends..... 33 - 34

About PeerSpot..... 35 - 36

Product Recap



OpenJDK Java

OpenJDK Java Recap

OpenJDK Java is an open-source implementation of the Java Platform, Standard Edition. It provides developers with a robust platform for creating applications and services, ensuring compatibility and support for Java applications across diverse environments.

As a leading choice for Java development, OpenJDK Java empowers developers with a reliable and high-performance environment. It supports the development of a wide range of applications, from enterprise-level software to mobile apps, thanks to its compatibility and extensive library of tools. OpenJDK Java facilitates seamless integration and offers updates that enhance security and performance, making it a respected choice among developers.

What are the most important features?

- **Open Source:** Free to use, modify, and distribute.
- **Cross-Platform:** Ensures compatibility across different OS.
- **Regular Updates:** Keeps up with security and performance improvements.
- **Comprehensive Libraries:** Extensive collection of class libraries for efficient development.
- **Community Support:** Backed by a vibrant community for assistance and collaboration.

What benefits should users look for in reviews?

- **Performance:** High efficiency even in demanding environments.
- **Scalability:** Can handle applications ranging from small to large scale.
- **Cost-Effective:** No licensing fees for usage and deployment.
- **Innovation:** Continuous improvements and enhancements.

In industries like finance, OpenJDK Java is often implemented for developing secure and scalable banking applications. In healthcare, it is used to create reliable and compliant data management systems, while the tech industry employs it for innovative enterprise solutions. OpenJDK Java consistently supports industry-specific requirements, enabling development teams to deliver customized and robust applications.

Valuable Features

Excerpts from real customer reviews on PeerSpot:



“OpenJDK Java has positively impacted my organization by significantly reducing parsing failures.”



Harshwardhan Gullapalli

AI Engineer at a educational organization with 51-200 employees



“OpenJDK Java is open for everyone, every developer, and every company, and it is free of cost.”



Aadarsh Mishra

Sde li at Greypip Software Pvt. Ltd.



“At the end of the day, Java is a platform and many open-source libraries and projects are added to the technology stack and then we are implementing enterprise applications, web applications, or even console applications just to show that this is something that can cover everything.”



Aradovan Adwan

University Professor at Algebra Bernays University



“OpenJDK Java has positively impacted my organization by providing a stable, reliable, and cost-effective platform for running Java applications.”



Anuja Deore

Assistant Engineer at Wipro Limited



“Using OpenJDK Java has impacted my organization positively, as it has significantly improved productivity by providing a stable and mature platform, while the rich set of libraries and frameworks like Spring Boot and Hibernate accelerates development and reduces the time needed, allowing us to reduce development time by approximately twenty-five percent and cut production incidents and downtime by fifteen percent.”



Shreya Reddy

Software Engineer at a financial services firm with 10,001+ employees



“I can observe savings in my ROI and categorize it as money-saving, as I see about 60% in money savings and 40% in time savings.”



Shailesh Shandilya

Principle Consultant at Verinite



“We have found OpenJDK Java to be highly performant and reliable compared to other Java distributions such as Oracle JDK, and in some cases, even more stable due to its open-source nature and community-driven updates.”



Hussain Gagan

FullStack Developer at EnactOn Technologies

What users had to say about valuable features:

“I really appreciate the inclusion of Java 8 with streams, which allows me to iterate through a list using streams instead of the traditional for-loop. Furthermore, I really appreciate the use of records in OpenJDK Java, which simplifies creating plain Java objects without needing to define so many getters and setters.

“The best features OpenJDK Java offers include Java 8 with the streams API. OpenJDK Java has positively impacted my organization because Java is known for being really powerful and is used daily to support services for many clients around the world..”

Fanon Jupkwo

Consultant at emagine

[Read full review](#)

“The best features OpenJDK Java offers are that it is open source and all the latest Java features can be used in OpenJDK. We used OpenJDK 21 version with those features, including virtual threads and strings which are also part of those features.

“The most valuable open-source aspects and latest Java features for us are virtual threads which were the most frequently used in our application.

“OpenJDK Java has improved our organization because it has some optimization techniques. Since our application was a high throughput application, we utilized these optimizations. We also make use of other Java features which are widely used in Java 17 as well..”

Himanshu Joshi

Senior Software Engineer at Netcracker

[Read full review](#) 

“The best features of OpenJDK Java in my experience include write-once, run-anywhere capability thanks to the JVM, along with powerful performance optimization and garbage collection tuning options for backend systems.

“I have seen significant improvements in system stability and resource utilization thanks to OpenJDK Java's garbage collection and JVM tuning capabilities. For instance, in one project, we were able to reduce memory-related issues by over 50% by fine-tuning the JVM's garbage collection settings.

“The open-source nature of OpenJDK Java is a significant advantage, giving flexibility to choose different builds such as Amazon Corretto or Red Hat builds. Additionally, the frequent release cycles help us adopt new features quickly.

“OpenJDK Java has positively impacted our organization by helping us standardize our backend stack across teams, making onboarding easier. New developers could ramp up faster since Java and OpenJDK Java are widely known and well documented..”

Hussain Gagan

FullStack Developer at EnactOn Technologies

[Read full review](#) 

The best features OpenJDK Java offers include its own garbage collection system, as it automatically removes unused objects, which is an important feature.

This feature has helped my development process because creating too many objects can lead to an out of memory situation, but the garbage collection efficiently deletes unused memory.

“OpenJDK Java stands out for its portability, as it runs on a write once, run everywhere principle due to its virtual memory and JVM, which converts every code into machine code, making it helpful to run code on any device.

“OpenJDK Java has positively impacted my organization by providing a large number of resources to find issues and creating a large environment that helps in many areas.

“Having a large environment and many resources has helped my team specifically when we tried to use Redis components, as we could consult previous blogs to maintain it, which was helpful for integrating other components into our code. .”

HARI KRISHNA M

Software Developer at CognitiveScale

[Read full review](#) 

“The best features OpenJDK Java offers depend on the version. Usually, the last few versions included virtual threads and all the optimizations that benefit applications including Spring applications, etc. Nowadays you don't need to rely on platform threads. You can use virtual threads without a problem and create them in vast numbers, not just hundreds, because asynchronous applications are very important for creating an environment in which you can be responsive, and especially in this AI age where you can combine everything together and try to provide your users with the best features and best capabilities of your system. Of course, the latest version includes the Vector API. I'm looking forward to actually using the best features related to that and all those small perks that are added to every single version, such as lambda expression optimizations, HTTP client, JavaFX that I use, and everything that helps you to be productive and to follow the best practices such as SOLID principles, design patterns, and clean code.

“OpenJDK Java has positively impacted my organization because the word open actually opens everything more to the community. Usually people don't want to be vendor-locked. That's why they need this opportunity to choose their own platform to create an environment where they feel the most comfortable and of course, the most productive. Every next version of OpenJDK, for example, the last few years we experienced optimization in garbage collectors, etc. Someone can also include their own garbage collector or choose from the list of available and supported garbage collectors. They can have this support that will give them full control over their environments. Everything that is added to the stack every six months also makes the whole community vibrant. We eagerly wait for those short-term releases every six months, but at the end of the day, the latest versions are those that have the most impact to the community..”

Aradovan Adwan

University Professor at Algebra Bernays University

[Read full review](#) 

The best features of OpenJDK Java are its cross-platform compatibility, stability, strong performance, automatic memory management, and large ecosystem of libraries and frameworks. Being open-source and widely used in enterprise environments also makes it a reliable choice for development and production workloads.

The feature that I have found most valuable is stability and reliability because in my day-to-day work, it helps ensure that Java applications run consistently in development, testing, and production environments with minimal issues. This reduces troubleshooting time and gives me confidence when deploying applications on Linux servers and containerized platforms.

“The feature worth mentioning is OpenJDK Java's strong ecosystem and community support because it is widely adopted. There is extensive documentation, a large collection of libraries and frameworks, and plenty of community resources available, which makes development, troubleshooting, and learning much easier.

“OpenJDK Java has positively impacted my organization by providing a stable, reliable, and cost-effective platform for running Java applications. Its open-source nature eliminates licensing costs, while its strong performance and compatibility help ensure smooth application deployment and maintenance.

“While I do not have exact figures, one noticeable benefit has been avoiding commercial Java licensing costs by using OpenJDK Java. This allows us to run and maintain Java-based applications without additional runtime licensing expenses while still receiving the performance and stability needed for production workloads. Additionally, its reliability helped reduce downtime and troubleshooting effort, improving overall operational efficiency. .”

Anuja Deore

Assistant Engineer at Wipro Limited

[Read full review](#) 

Other Solutions Considered

“I do not think we were using a different solution as we started using OpenJDK 21 when I joined this project within our organization which was a product development. The upper management was using this only..”

Himanshu Joshi

Senior Software Engineer at Netcracker

[Read full review](#) 

“We were initially using Oracle JDK but switched to OpenJDK Java due to licensing costs and flexibility concerns, as OpenJDK Java offered the same core functionality without the restrictions..”

Hussain Gagan

FullStack Developer at EnactOn Technologies

[Read full review](#) 

“I do not have the answer to whether we evaluated other options before choosing OpenJDK Java because I have been part of a development team, and higher management only chose this. We made use of it..”

Himanshu Joshi

Senior Software Engineer at Netcracker

[Read full review](#) 

“Before choosing OpenJDK Java, we evaluated other options including Oracle JDK and some JVM alternatives, but most were either expensive or less flexible compared to OpenJDK Java..”

Hussain Gagan

FullStack Developer at EnactOn Technologies

[Read full review](#) 

“Before standardizing on OpenJDK Java, we used a more fragmented approach. We had a mix of Python-based text extraction tools and some proprietary PDF parsing libraries. The problem was inconsistency; different document formats failed unpredictably with different tools. We ended up with manual workarounds, with team members extracting data from PDFs that the automated pipeline could not handle reliably. We switched to OpenJDK Java and Apache Tika because we required something battle-tested and stable for production use with financial documents. The Java ecosystem had proven libraries that could handle edge cases and unusual document formats that were giving us trouble. Tika's maturity and the JVM's stability made it the right choice, simplifying our tech stack significantly. Instead of maintaining multiple parsing tools across different languages, we have one solid foundation that reduces operational complexity. In hindsight, it was absolutely the right decision; the consistency and reliability improvements justified the learning curve my team had to undergo coming from a Python-heavy background..”

Harshwardhan Gullapalli

AI Engineer at a educational organization with 51-200 employees

[Read full review](#) 

“I would like to see some alternative tools that will help developers to start development with AI assistance because JetBrains IntelliJ is something that is really powerful, but when you are no longer a student, you need to pay for the tooling. Before, let's say 20 years ago, we had Eclipse or JDeveloper environments, but they are not keeping pace with the competitors. That's why this open keyword should also popularize the whole Java enterprise and community, etc. It will help to have a tool that will be competitive to the other elements because, for example, we had the Scene Builder for JavaFX that was maintained by Oracle, but they abandoned it. So this parallel thread, let's say Gluon JavaFX is now actually producing good results. That's why this branding is very important. If you actually stand behind a tool that will be created for this new era of development with the help of AI and live coding, it could help the platform to be more popular and of course, involve more than 10 million developers. Although Java is already over 30 years old, it will still be a main competitor on the market. For all enterprise development solutions, Java as an open-source technology could still be the number one choice and stay for a long time as number one..”

Aradovan Adwan

University Professor at Algebra Bernays University

[Read full review](#) 

ROI

Real user quotes about their ROI:

“I can observe savings in my ROI and categorize it as money-saving. If I quantify the savings, I see about 60% in money savings and 40% in time savings..”

Shailesh Shandilya

Principle Consultant at Verinite

[Read full review](#) 

“I have seen a return on investment as I think a lot of money has been saved, but I do not have that number currently because higher management handles that in our organization..”

Himanshu Joshi

Senior Software Engineer at Netcracker

[Read full review](#) 

“The ROI has been strong with OpenJDK Java due to zero licensing costs, stable performance, and reduced operational overhead, overall improving efficiency by around 25% across backend teams..”

Hussain Gagan

FullStack Developer at EnactOn Technologies

[Read full review](#) 

While I do not have specific ROI metrics, I have seen value through avoiding commercial Java licensing costs, reducing software expenses, and leveraging a stable, widely supported platform. OpenJDK Java also helps save time by providing reliable performance and a large ecosystem of tools and documentation, which reduces troubleshooting and development effort.

Anuja Deore

Assistant Engineer at Wipro Limited

[Read full review](#) 

“A return on investment has been realized, specifically in terms of money saved and time saved.

“Approximately two hours daily have been saved in terms of maintenance since implementing OpenJDK Java, and in terms of project estimation, about four to six story points are saved..”

Amol Tyagi

AWS solutions architect at Deloitte

[Read full review](#) 

“I have indeed seen a return on investment with OpenJDK Java, particularly in operational efficiency. As mentioned earlier, we reduced manual document processing interventions from around 10 to 15 percent down to 2 to 3 percent. That means our team of three document processors can handle approximately 40 to 50 percent more documents per week without adding headcount. Over a year and a half, that is a significant capacity gain without hiring additional people. On the cost side, being open source means we incurred zero licensing fees. For a financial document processing platform, that represents real money saved since commercial Java solutions would have cost us thousands annually in licensing alone. Development time is another critical metric; our engineers spend significantly less time debugging parsing failures and environment issues. I estimate we save approximately 15 to 20 percent of engineering time that would have gone to infrastructure maintenance, which we redirect toward building new features and improving our LLM extraction accuracy. The speed of processing a document from ingestion to final structured output in under two minutes means we meet aggressive client turnaround SLAs, translating directly to client retention and upsell opportunities..”

Harshwardhan Gullapalli

AI Engineer at a educational organization with 51-200 employees

[Read full review](#) 

Use Case

“My main use case for OpenJDK Java is for various development purposes including desktop, web, integration patterns, interoperability, and everything that is usually included in standard enterprise projects and small demo examples that I use in teaching while instructing students at my courses..”

Aradovan Adwan

University Professor at Algebra Bernays University

[Read full review](#) 

My main use case for OpenJDK Java is to build Java backend code and backend logic. A specific example of a project where I used OpenJDK Java for backend logic is in my current organization, where we are using it to build our Deva code, which is an application that functions as a digital wallet.

In addition to my main use case, while studying, I used JDK to build backend logic for my personal project, which is a form-fill assist application. .”

HARI KRISHNA M

Software Developer at CognitiveScale

[Read full review](#) 

“My main use cases for OpenJDK Java include creating open source projects and working with the Spring Boot framework, as well as projects in the banking industry and various tech projects.

“I use OpenJDK Java to build APIs, and we maintain some legacy codebases using Java 6, Java 8, Java 17, or Java 21. We have both monolithic applications and microservices applications..”

Fanon Jupkwo

Consultant at emagine

[Read full review](#) 

“My main use case for OpenJDK Java is developing a telecom product which requires OpenJDK 21 with Apache Kafka. We are using Apache NiFi, and that was the major requirement for OpenJDK because we were using the Apache NiFi latest version which uses Java 17 and above versions.

“A specific example of how OpenJDK Java fits into our telecom product is that Apache NiFi requires Java 17 and above versions, and OpenJDK is a free version which can be used. We use it and also use virtual threads for Java 21, so we benefit from the features of OpenJDK..”

Himanshu Joshi

Senior Software Engineer at Netcracker

[Read full review](#) 

“I have been using OpenJDK Java for around two years, mainly for building backend services and APIs in microservices architecture.

“My primary use case with OpenJDK Java has been building scalable backend systems, including REST APIs, async job processors, and event-driven services.

“One example is a booking system I worked on where we handled around 50,000 daily requests using OpenJDK Java with proper JVM tuning, which reduced API response times by about 30% and improved throughput significantly without increasing infrastructure costs.

“Beyond APIs, I have also used OpenJDK Java for batch processing jobs and background workers..”

Hussain Gagan

FullStack Developer at EnactOn Technologies

[Read full review](#) 

I have been using OpenJDK Java for the past eight to nine months primarily for developing and running enterprise and web-based applications. Most of my work involves building back-end services, automation scripts, and Java-based tools that integrate with Linux environments and containerized platforms.

Recently, the task where I used OpenJDK Java was deploying a Java-based web application on a Linux server using Apache Tomcat. OpenJDK Java served as the runtime environment, and I used it for application testing, configuration, and troubleshooting to ensure stable performance in production.

“Besides running and supporting Java-based web applications, I also used OpenJDK Java for learning and testing new Java features, troubleshooting application issues, and working with containerized environments such as Docker and Kubernetes. .”

Anuja Deore

Assistant Engineer at Wipro Limited

[Read full review](#) 

Setup

The setup process involves configuring and preparing the product or service for use, which may include tasks such as installation, account creation, initial configuration, and troubleshooting any issues that may arise. Below you can find real user quotes about the setup process.

Setup is straightforward, especially on Linux systems, and the overall cost is mainly related to infrastructure and maintenance rather than software licensing.

Anuja Deore

Assistant Engineer at Wipro Limited

[Read full review](#) 

“The experience with pricing, setup cost, and licensing for OpenJDK Java is straightforward and completely free since it is open-source. We usually install it via package manager or Docker images, making onboarding new services quick..”

Hussain Gagan

FullStack Developer at EnactOn Technologies

[Read full review](#) 

“I do not think there are any challenges with initial setup. However, when installing any application, everything should be set up automatically, similar to how Java is installed. There should be no need to write environment variables. They should already be set up in the system. At the beginner level, developers can only write the code and do not have to use environment variables at all..”

Aadarsh Mishra

Sde li at Greyp Software Pvt. Ltd.

[Read full review](#) 

“We deployed OpenJDK Java on-premises. We run our OpenJDK Java infrastructure on Linux servers hosted in our own data center setup, which is important for our use case because we handle sensitive financial documents for clients including balance sheets, trial balances, and compliance data. Having on-premises infrastructure gives us direct control over data residency and security, crucial for client trust and regulatory compliance. We consider cloud options eventually; Docker containerization makes it theoretically portable to cloud environments if needed. However, we decided to keep it on-premises for data sovereignty reasons. The on-premises setup started well for us because our document volumes are predictable, allowing us to manage infrastructure costs directly. We do not need the auto-scaling flexibility that cloud would offer..”

Harshwardhan Gullapalli

AI Engineer at a educational organization with 51-200 employees

[Read full review](#) 

Customer Service and Support

OpenJDK Java itself does not provide dedicated commercial customer support because it is an open-source project. However, it has a large community, extensive documentation, and active forums. Organizations that require enterprise-grade support can obtain it through vendors such as Red Hat, Oracle, or Eclipse.

Anuja Deore

Assistant Engineer at Wipro Limited

[Read full review](#) 

“Community support is available, and there is also technical support with many discussions happening in different communities since it is an open platform, offering a lot of assistance. In my case, community support has been sufficient, and I did not need to reach out to technical support for the product..”

Shailesh Shandilya

Principle Consultant at Verinite

[Read full review](#) 

“The community support for OpenJDK Java is massive and active. When we encounter issues or have questions, Stack Overflow usually has answers. OpenJDK Java documentation itself is comprehensive, though it is dense and assumes some Java experience. For our team, coming from Python and JavaScript backgrounds, we had to do additional learning, but the resources are there. Specifically, the Apache Tika community has helped us greatly; the documentation for Tika is solid, with good examples, and the community forums are responsive. When we run into edge cases with parsing specific document formats, we typically find discussions or solutions within the community. The trade-off with open source is that we do not pay for dedicated support; however, we are also not dependent on the vendor support queue. We need to be more self-sufficient, but this approach works for our team. We troubleshoot, read documentation, check community discussions, and usually find answers within hours rather than waiting for support tickets..”

Harshwardhan Gullapalli

[Read full review](#) 

AI Engineer at a educational organization with 51-200 employees

Other Advice

“The optimizations impacted our organization significantly since we were using Apache NiFi with OpenJDK 21, and our application throughput was really high. We were able to process 7,000 records per minute, so the throughput was really high and the performance was also good. The JVM was not running high and the performance was really good. I would rate this product a 9 overall..”

Himanshu Joshi

Senior Software Engineer at Netcracker

[Read full review](#) 

“My advice to others looking into using OpenJDK Java is to please go ahead and use it because it is the best. If you are using Python, then you understand the syntax is easy in Python, but the indentation is problematic, whereas OpenJDK Java is extremely awesome; we do not have an indentation issue, and the code is understandable and not that difficult. I rate this product a 10..”

Validated user

Sr cloud engineer at a tech vendor with 201-500 employees

[Read full review](#) 

“Regarding performance, I cannot provide specific details as I am a Java developer focused on development rather than infrastructure concerns.

“It is difficult for me to share specific outcomes regarding OpenJDK Java because I am only a Java developer and this comparison requires broader organizational perspective.

“I think OpenJDK Java is improving the way it handles AI. Years ago, people thought of Python for AI projects, but now many are moving to Java, and I appreciate that Java is becoming a programming language for AI. Regarding governance and security, I think the team at OpenJDK is doing well.

“I cannot speak to the accuracy and reliability of AI outputs in OpenJDK Java.

“I would advise others looking into using OpenJDK Java that it is really powerful, and I enjoy working on the Java ecosystem. The syntax and documentation are great, and our applications are based on OpenJDK Java while serving many clients, so I recommend not hesitating to use OpenJDK Java. I rate this product a ten out of ten..”

Fanon Jupkwo

Consultant at emagine

[Read full review](#) 

OpenJDK Java integrates very well with other technologies in my environment. It works seamlessly with databases through JDBC, supports REST APIs and web services, and fits well into DevOps workflows using tools such as Docker, Kubernetes, [Git](#), and CI/CD pipelines. This compatibility makes it easy to build, deploy, and maintain enterprise applications.

OpenJDK Java updates and patch management are handled through my standard system maintenance process. The infrastructure team regularly applies security patches and version updates during scheduled maintenance windows, with testing performed before deployment to production to ensure application compatibility and stability.

“The learning curve is generally moderate. New developers can learn the basics quickly, thanks to Java's clear syntax and extensive documentation, while more advanced topics such as enterprise frameworks, performance tuning, and large-scale application development take additional experience to master.

“OpenJDK Java supports compliance efforts by providing a secure and well-

maintained runtime environment with regular security updates and patches. While compliance depends mainly on application design and organizational processes, OpenJDK Java helps by offering a stable platform that aligns with enterprise security and governance requirements.

“Personally, I expect OpenJDK Java to continue evolving with improvements in performance, security, and cloud-native development. I am particularly keeping an eye on enhancements for containers and Kubernetes environments, better support for modern application architectures, and continued innovation through regular Java releases while maintaining backward compatibility.

“OpenJDK Java remains important for my future plans because many enterprise applications and backend services rely on Java. Given its stability, strong ecosystem, and open-source nature, I expect it to continue being used for new applications, modernized services, and cloud-native deployments in the future.

“It is generally easy to find developers with OpenJDK Java skills because Java is one of the most widely used programming languages. There is a large talent pool, extensive learning resources, and many developers already have experience with Java-based applications and frameworks.

“I would recommend OpenJDK Java to organizations looking for a reliable, mature, and cost-effective Java platform. It offers strong performance, broad ecosystem support, and no licensing costs. My advice would be to choose an appropriate LTS version, establish a regular patching process, and thoroughly test applications before major version upgrades to ensure compatibility. I have provided this review with an overall rating of nine out of ten. .”

Anuja Deore

Assistant Engineer at Wipro Limited

[Read full review](#) 

“I have found that the learning curve for new developers adopting OpenJDK Java in our team is relatively moderate. They typically already have a background in Java,

allowing them to get up to speed quickly and start contributing to our projects with some guidance and resources.

“I have had a smooth experience integrating OpenJDK Java with other technologies, particularly [Spring Boot](#) and containerization tools such as [Docker](#). OpenJDK Java works seamlessly with [Spring Boot](#), allowing us to develop and deploy applications quickly and efficiently.

“I handle security and updates for OpenJDK Java by regularly checking for updates and applying them as soon as possible, while also making sure to follow best practices for secure coding and configuration. I have found that using tools such as Docker and [Kubernetes](#) helps streamline the process of keeping our deployments up to date and secure.

“We monitor and manage the performance of our applications running on OpenJDK Java using tools such as Java [Mission Control](#) and VisualVM, which provide detailed insights into memory usage, CPU usage, and other key metrics.

“We have found OpenJDK Java to be highly performant and reliable compared to other Java distributions such as Oracle JDK, and in some cases, even more stable due to its open-source nature and community-driven updates. It has consistently delivered high-quality performance and reliability across various applications and deployments.

“We handle version upgrades with OpenJDK Java by regularly checking for updates and applying them as soon as possible to ensure that we have the latest security patches and features.

“Since OpenJDK Java is community-driven, we rely mostly on forums and documentation. For critical systems, we sometimes use vendor-supported builds for SLA coverage.

“I would recommend starting with OpenJDK Java if you are building backend systems, as it is cost-effective and production-ready. I suggest investing some time in understanding JVM tuning early on.

“OpenJDK Java is a very mature and reliable platform, giving enterprise-grade capabilities without the cost overhead, making it an excellent choice for most

backend systems. I would rate OpenJDK Java a nine out of ten..”

Hussain Gagan

FullStack Developer at EnactOn Technologies

[Read full review](#) 

“An example of how I use OpenJDK Java in my teaching and enterprise projects is that since everything I teach is related to Java, I create an object, for example, in the IntelliJ JetBrains environment. Then when a new Java version is released, perhaps version 25 or 26 every six months, I download it through this tool, JetBrains IntelliJ, and configure my project, mostly [Spring](#) projects or vanilla Java projects without any specific frameworks to establish the basic functionality, and then I start developing all the features that are required for this purpose.

“At the end of the day, Java is a platform and many open-source libraries and projects are added to the technology stack and then we are implementing enterprise applications, web applications, or even console applications just to show that this is something that can cover everything. With AI, it's much easier because the best models are using the latest Java versions without any problems, so you can immediately be very productive and at the end of the day, you can deliver your application much faster than you could five or six years ago.

“I want to show many examples of performance, measurements, reports, and automation such as unit testing, CI/CD, and especially integration with AI regarding how I use OpenJDK Java in my teaching and development.

“The efficiency of virtual threads is the first thing that comes to mind when I talk about how virtual threads or any of these features have changed the way I and my students develop applications. Instead of creating platform threads that are very heavy for the system because the memory footprint is 1 MB to 2 MB per thread, with virtual threads, you actually have a footprint of 1 KB, which means that you can create many threads and they can be synchronized, especially if you use asynchronous operations or prefer to have non-blocking changes to every resource

that you are contacting with. This is something that can be done much more easily and you can create a pool of threads that you use efficiently, so you can be much more productive when you use those things. Of course, your system can utilize the hardware resources much more efficiently and that's why this is the future of asynchronous parallel development. For example, when you synchronize virtual threads with your main thread in JavaFX UI, you can actually create something that will run in the background without taking too much processor time and actually increase the responsiveness of every application.

“I want to highlight especially the newest features that are included in the latest releases, such as the Vector API, because I tested the Vector API a few years ago when it was in the incubation period. It was still promising to actually use some features of CPUs to even better optimize the processing of large data. Especially nowadays we use many datasets when we prepare the datasets to train our LLM models. This is something I would like to highlight and I'm looking forward to getting the final version so I can play with it in my own use cases.

“My advice for others looking into using OpenJDK Java is that they need to compare their own solutions to OpenJDK Java and then decide for themselves because it's very important to have a choice, not to be vendor-locked. You should have a choice and compare how good this is accepted, how easy you can actually get all the information, and how good it is accepted across the AI landscape. If the AI generates many examples that can be easily integrated or run on OpenJDK Java, that's the way to go. I give this review a rating of 9 out of 10..”

Aradovan Adwan

University Professor at Algebra Bernays University

[Read full review](#) 

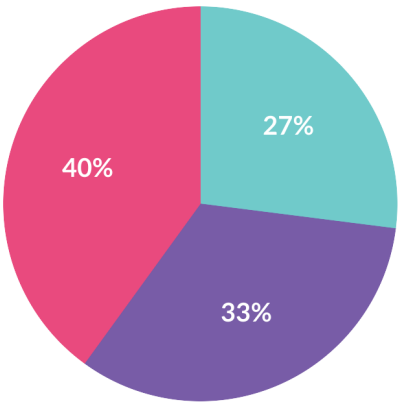
Top Industries

by visitors reading reviews

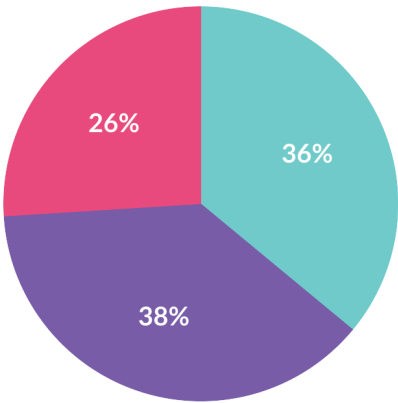


Company Size

by reviewers



by visitors reading reviews



Large Enterprise Midsize Enterprise Small Business

About this buyer's guide

Thanks for downloading this PeerSpot report.

The summaries, overviews and recaps in this report are all based on real user feedback and reviews collected by PeerSpot's team. Every reviewer on PeerSpot has been authenticated with our triple authentication process. This is done to ensure that every review provided is an unbiased review from a real user.

Get a custom version of this report... Personalized for you!

Please note that this is a generic report based on reviews and opinions from the collective PeerSpot community. We offer a [customized report](#) of solutions recommended for you based on:

- Your industry
- Company size
- Which solutions you're already considering

The customized report will include recommendations for you based on what other people like you are using and researching.

Answer a few questions in our short wizard to get your customized report.

[Get your personalized report here](#)

About PeerSpot

PeerSpot is the leading review site for cloud, AI, and business software. We created PeerSpot to provide a trusted platform to share information about software, applications, and services. Since 2012, over 22 million people have used PeerSpot to choose the right software for their business.

PeerSpot helps tech professionals by providing:

- A list of products recommended by real users
- In-depth reviews, including pros and cons
- Specific information to help you choose the best vendor for your needs

Use PeerSpot to:

- Read and post reviews of products
- Access over 30,000 buyer's guides and comparison reports
- Request or share information about functionality, quality, and pricing

Join PeerSpot to connect with peers to help you:

- Get immediate answers to questions
- Validate vendor claims
- Exchange tips for getting the best deals with vendor

Visit PeerSpot: www.peerspot.com

PeerSpot

244 5th Avenue, Suite R-230 • New York, NY 10001

reports@peerspot.com

+1 646.328.1944