2 Answers

Shahzad Masud, Certified Java Programmer
Written Dec 15

It entirely depends on the target platform on which you want to run JDK. Technical differences are a consequence of the goal of each one (OpenJDK is meant to be the reference implementation, open to the community, while Oracle is meant to be a commercial one)

They both have "almost" the same code of the classes in the Java API; but the code for the virtual machine itself is actually different, and when it comes to libraries, OpenJDK tends to use open libraries while Oracle tends to use closed ones.

In my practical experience, OpenJDK works better for large number of users with small request count, while it become worse for small number of user with prolonged (50 seconds + per request cycle). This is an undocumented behaviour, and never seen anywhere other than experienced on some J2EE containers.

2k Views

Upvote　　Downvote　Comment　1

Alan Deikman, More lines of code than I care to think about.
Written Apr 21, 2016

In the past several years I have yet to detect any functional difference between OpenJDK and the official Oracle release. There are of course differences between the versions in each chain but all the latest language features seem to be present in the latest of each variant.

My advice is stick to the Java that your platform normally comes with. On Linux that will be OpenJDK, on Mac I think you get Oracle.   I move projects between those two platforms all the time without any trouble.

Right now I am working with a large open-source project that insists you use Oracle 8 and nothing else. If you try to use OpenJDK and run into trouble they will tell you to switch over before offering any other help.

It is not as if the authors expect OpenJDK not to work or that Oracle is better. For them it is a matter of not wasting time trying to trace down what is different in the development environment.   It eliminates one more variable.