

## U.S. Supreme Court Finds Google's Copying of Oracle's Java API Code a Non-Infringing Fair Use

April 27, 2021 by [Dasha Chestukhin](#)

In *Google LLC v. Oracle America, Inc.*, Case No. 18–956, 593 U. S. \_\_\_\_ (2021), the U.S. Supreme Court held that Google's unauthorized copying of around 11,500 lines of Oracle's computer code was a fair use and therefore not an infringement.

Assuming, but expressly not deciding, that Oracle's copied Java SE program code was copyrightable, the Supreme Court reversed the Federal Circuit's most recent decision and held that Google's copying was a permissible fair use. The Supreme Court's decision placed particular emphasis on the nature of the copied code, which was part of a user interface and therefore "inherently bound together with" both the uncopyrightable idea of the interface's organization and Google's other, independently-created code for its Android platform for smartphones. The Supreme Court also held that Google's recontextualization of Java, which had been created for computers, in the smartphone environment was transformative and not a market substitute for Oracle's code. Finally, Google's wholesale copying of 11,500 lines of code was acceptable because it was tethered to Google's valid purpose of allowing computer programmers familiar with Java to write code for Android.

### Factual Background

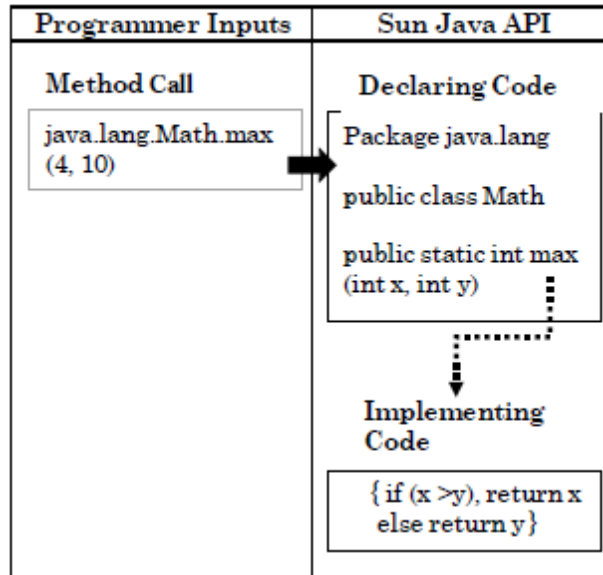
Java SE is a computer program that uses the popular Java computer programming language that was originally developed by Sun Microsystems (Oracle's predecessor). In 2005, Google acquired startup Android Inc. and, subsequently, roughly 100 Google engineers worked for over three years to create Google's Android software platform for smartphones. At the time of Android's acquisition, many software developers understood and wrote programs in the Java programming language, particularly using Sun's popular Java SE platform, which allowed them to write programs that would run on any desktop or laptop computer. To enable these programmers to work with its new Android platform, Google copied verbatim roughly 11,500 lines of code from the Java SE program that are part of a tool called an Application Programming Interface or API. As Google saw it, this API code included the tasks most likely to be useful to Android programmers. Android was released in 2007 and Android-based devices have been successful (producing more than \$42 billion in revenue as of 2015). In 2010, Oracle purchased Sun and soon thereafter sued Google for copyright infringement.

### Technical Background

By way of background, an API is a tool that allows programmers to use prewritten code to build certain functions, rather than writing their own code from scratch. Here is how it works: A programmer enters a string of code called a "method call" (which Oracle did not claim was copyrightable). That method invokes a particular task, which is located within a particular class,

which is in turn located within a particular package. The lines of code that “name the methods, classes, and packages” are “declaring code”—which is what was at issue in the case. The declaring code both constitutes a set of shortcuts for programmers and reflects the precise way in which Java’s creators concretely set out particular tasks. Each method’s declaring code is associated with particular lines of “implementing code,” which actually instructs the computer in the programmer’s application. It was undisputed that Google itself wrote all of Android’s implementing code. Appendix B to the Court’s decision provides a helpful diagram showing the interoperation of the various levels of code:

Sun Java API Diagram



## Procedural Background

After the case was initially filed, the district court judge held that Oracle’s so-called “declaring code” was utilitarian and thus not copyrightable. The Federal Circuit reversed, but left open the possibility that Google might have a fair use defense. The Supreme Court rejected Google’s petition for review of this decision. Upon remand, a jury unanimously found that Google’s use of Oracle’s code was fair. Oracle again appealed and the Federal Circuit overturned the jury’s verdict, holding that Google’s use was not fair as a matter of law. Google filed another petition for the Supreme Court’s review, which was granted.

## Supreme Court Decision

The Supreme Court expressly declined to decide whether Oracle’s API code was copyrightable, accepting that it fell within the definition of what can be copyrightable “purely for argument’s sake.” The Court explained that the “rapidly changing technological, economic and business-related circumstances” cautioned against deciding more than necessary to resolve the parties’ dispute—but then proceeded to issue a fairly sweeping decision based on fair use grounds. The Supreme Court found that all four fair use factors weighed in favor of finding that Google’s use was fair.

The Supreme Court found that the “nature of the copyrighted work” (i.e., the Java API code) weighed in favor of fair use. Although the declaring code was part of a computer program (which is, generally speaking, copyrightable), it differed from other kinds of copyrightable computer

code because it was inextricably bound together with: (1) an uncopyrightable general system of dividing computing tasks; (2) the uncopyrightable idea of organizing tasks; (3) the use of specific commands (method calls), which Oracle did not claim were copyrighted; and (4) copyrightable implementing code, which was all created by Google. Moreover, Java's creators intentionally tried to write declaring code that would be intuitively easy to remember, so as to attract programmers. For these reasons, the Court concluded that declaring code, "if copyrightable at all," was further from the core of copyright than most computer programs (e.g., implementing code), meaning that the "nature of the copyrighted work" weighed in favor of fair use.

The Court then found that the "purpose and character" of Google's use was transformative because it reimplemented the Java API in a new environment, namely, smartphones. Even though Google used the Java API for the same reason that Sun created it (to enable programmers to call up implementing programs to accomplish specific tasks), the Court still found the use transformative because it sought to create "new products" and expand the use of Android-based smartphones. The Court also highlighted the ways in which reimplementing an interface like the Java API can further development of computer programs generally, including by making it easier for programmers to use previously acquired skills. Accordingly, the "purpose and character" of Google's use weighed in favor of fair use.

Third, the Court held that the "amount and substantiality" of the Java API copied by Google weighed in favor of fair use because it was driven by a transformative purpose. Google copied the entirety of about 11,500 lines of Java API code, which constituted 0.4% of the total set of Java API code, including all the implementing code. Despite the verbatim copying, the Court held that it was more appropriate to take into account the several million lines that Google did *not* copy. The Java API declaring code that Google copied was inseparably bound to the implementing code that actually carried out the necessary tasks, and it was undisputed that Google had written its own implementing code. Moreover, Google copied the declaring code to achieve its purpose of created a platform for a different computing environment (smartphones) and because programmers had already learned to work with the Java API system. Because the amount of Google's copying was "tethered to a valid, and transformative, purpose," the Court concluded that the "substantiality" factor weighed in favor of a fair use finding.

Finally, the Court found that the "market effect" of Google's copying weighed in favor of fair use because Oracle failed to meaningfully compete in the smartphone market and because allowing Oracle to enforce its copyright would cause damage to the public (including computer programmers who had invested in learning Java). In particular, the Court reasoned that the jury could have found that Android did not harm the actual or potential markets for Java SE at least in part because Oracle was unable to enter or succeed in the smartphone market and because Android was not a market substitute for Java SE. Further, third parties, like computer programmers, had already invested in learning Java, so permitting Oracle to lock up use of the Java API's declaring code would stifle creative improvements, new applications and new uses developed by those who have already learned to work with that interface.

Justice Breyer wrote the opinion in which five other Justices joined. But Justice Thomas, joined by Justice Alito, wrote a lengthy dissent arguing that the majority should have discussed the copyrightability of the declaring code, and although agreeing that it is indeed copyrightable, arguing that the majority's fair use analysis was flawed.

## Conclusion

Although the Supreme Court's decision is an apparent win for software developers, it should not be read too broadly because it is so fact- and context-specific. Copying computer code now seems allowed to the extent needed to allow interoperability, but it is unclear whether and to what extent copying beyond that would be permissible.

For further information, contact [Dasha Chestukhin](#), [Tom Kjellberg](#) or your CLL copyright attorney.

## [Dasha Chestukhin](#)



### Associate

[Email](#) | 212.790.9251

Dasha's practice encompasses a broad range of intellectual property matters, including trademarks, copyrights, domain names, unfair competition and patents.