

Why Should You Choose Kotlin?

We'll cover the following



- Why do developers prefer Kotlin?

Why do developers prefer Kotlin?

Kotlin may be the right choice for your current project, or the next one, for many reasons:

- Kotlin delivers on the “less is more” promise—you write less boilerplate code. The less code you write, the less your team has to maintain, resulting in fewer errors to deal with.
- Kotlin gives you the freedom to mix the imperative and functional styles of programming; you pick what’s best for the problem at hand. Also, you may write it in one way and refactor it later as you desire—make it work, then make it better real soon.
- Kotlin offers lots more compile-time safety when compared to a lot of other statically typed languages. The code you write will fail less and fail fast—during compilation rather than runtime. This is one of the reasons why the [Spring](#) team decided to embrace Kotlin.
- Kotlin coroutines makes it a lot easier to create high-performance asynchronous code, compared to what’s available in the Java Development Kit.
- Some of the features that are scheduled to appear in future versions of Java are already in Kotlin—you can experience and benefit from future Java right now by using Kotlin.
- You may intermix Kotlin and Java code in your projects—using Kotlin is not an all-or-nothing proposition.

- You can not only use fluent DSL-like syntax to interact with APIs, like in the Spring Kotlin API, but also design your own APIs to be fluent and expressive for programmers who use your code.
 - You can reduce duplication among parts of your system with Kotlin. For example, the same business rules that check users' input may be compiled to Java bytecode for back-end validations, transpiled to JavaScript for front-end validation, compiled to native binaries to run on targeted platforms like iOS and Android, or to WebAssembly to run within browsers.
 - Finally, Kotlin is a great choice for Android development since it's an official language for that platform.
-

You'll see more about why Kotlin is exciting in the many lessons of this course. Buckle up—it's going to be a fun ride. Let's start by getting the SDK installed so we can start writing Kotlin.