

# Using Kotlin for Android Development

Kotlin is a great fit for developing Android applications, bringing all of the advantages of a modern language to the Android platform without introducing any new restrictions:

- **Compatibility:** Kotlin is fully compatible with JDK 6, ensuring that Kotlin applications can run on older Android devices with no issues. The Kotlin tooling is fully supported in Android Studio and compatible with the Android build system.
- **Performance:** A Kotlin application runs as fast as an equivalent Java one, thanks to very similar bytecode structure. With Kotlin's support for inline functions, code using lambdas often runs even faster than the same code written in Java.
- **Interoperability:** Kotlin is 100% interoperable with Java, allowing to use all existing Android libraries in a Kotlin application. This includes annotation processing, so databinding and Dagger work too.
- **Footprint:** Kotlin has a very compact runtime library, which can be further reduced through the use of ProGuard. In a [real application](#), the Kotlin runtime adds only a few hundred methods and less than 100K to the size of the .apk file.
- **Compilation Time:** Kotlin supports efficient incremental compilation, so while there's some additional overhead for clean builds, [incremental builds are usually as fast or faster than with Java](#).
- **Learning Curve:** For a Java developer, getting started with Kotlin is very easy. The automated Java to Kotlin converter included in the Kotlin plugin helps with the first steps. [Kotlin Koans](#) offer a guide through the key features of the language with a series of interactive exercises.

The only sizable difference is that Kotlin removes the superfluous features of Java such as null pointer exceptions. It also removes the necessity of ending every line with a semicolon. In short, Kotlin is much simpler for beginners to try as compared to Java and it can also be used as an “entry point” for Android App Development.