

Kotlin Essentials

Training Syllabus

Course Overview

Discover why Google has pinned Kotlin as the preferred language for Android development with this 2-day course. You will learn Kotlin development techniques and language principles, and will walk away knowing how to migrate existing Java projects to Kotlin.

Who Should Take This Course

- Developers who are looking to explore the Kotlin language
- Developers who wish to see both object-oriented and functional programming paradigms implemented in a modern JVM language
- Developers interested in learning a language with multi-platform applications
- Android developers who want to learn new skills to advance their career
- Teams looking to upgrade their applications from Java to Kotlin.

Syllabus

The Basics

- Understand writable and read-only properties and take a look at Kotlin's type system
- Work with control flow statements as assignable expressions
- Differentiate nullable and non-nullable types and how to work with null safely
- Understand safe exception handling in Kotlin

Collections and Functions

- Learn how to represent series of data with mutable and read-only collection types
- Take a closer look at collections, including destructuring, type checking, smart casting, and type parameter constraints
- Learn about defining class- and file-level functions and specifying their return types
- See how to pass functions as arguments, lambda notation, and function references
- Use functional paradigms to clean up code, including map, flatMap, filter, and reduce, as well as how to define your own functional operators
- Understand type variance and generic type parameters

Objects, Inheritance, and Extensions

- Apply object-oriented programming concepts to Kotlin
- Learn about constructors, equality, and preconditions for object construction
- Learn how to use data classes, enums, and sealed classes to represent data and algebraic data types
- Override class behavior and define abstract, open or closed classes
- Understand extension functions to safely add behavior to classes you don't control

Advanced Kotlin

- Learn how to interoperate with Java classes and support Java compatibility
- Use Ktor to stand up a simple web server in Kotlin
- Leverage coroutines to efficiently and safely move work between threads
- Use Channels to communicate between coroutines
- Represent application state using the Flow API to observe and react to changes in values over time
- Step into Kotlin Multiplatform to see how Kotlin can be used across platforms and stacks