

Android & Kotlin Essentials

Training Syllabus

Course Overview

If you need your team to write Android applications in Kotlin and they have little to no experience with either, this course is just what you need. In just 6.5 days, we'll have students building Android applications using Kotlin.

Who Should Take This Course

- Developers who want or need to create Android apps using the Kotlin programming language but have little to no experience with Kotlin and little to no experience with Android.
- Students should already be familiar with basic programming concepts: variables, statements, functions, arrays, data structures and common programming problems.

Syllabus

Kotlin Programming

- Understand writable & read-only properties, take a look at Kotlin's type system, and learn about nullable and non-nullable types.
- Learn about the standard functions included with Kotlin's standard library: let, apply, also, takeIf, and run
- Take a closer look at collections, including destructuring, type checking, smart casting, and type parameter constraints
- Use the function type to define functions that accept other functions, and learn how to optimize them by using the inlining feature of the Kotlin compiler
- Use the functional programming features of Kotlin to clean up code, including map, flatMap, filter, and reduce, as well as how to define your own functional operators and higher order functions.

Objects, Inheritance, and Extensions

- Work with object-oriented programming concepts
- Learn about primary and secondary constructors, equality, and preconditions for object construction
- Learn how to use data classes and enums to represent data and algebraic data types
- Override classes, and define abstract, open or closed classes
- Use Kotlin's built-in singleton implementation
- Understand extension functions to safely add behavior to classes you don't control.

Advanced Kotlin

- Learn how to interoperate with Java classes.
- Explore Kotlin's platform independence
- Use coroutines to do work in the background
- Use extensions with receivers to define a domain-specific language

Android Foundations

- Understand what an Activity is and how they are used to construct Android apps.

- Master the activity lifecycle to elegantly respond to device changes.
- Maintain UI state across rotation and process death
- Use the debugger to find and fix bugs.
- Safely communicate across Activities.
- Explore the differences between versions of Android and use new features while maintaining compatibility with older versions.

Full-Featured Android

- Use Fragments to create reusable components and to respond elegantly to different types of devices.
- Display lists of items performantly with the RecyclerView API.
- Peek behind the scenes at Android's layout system and understand how to use layout tools such as ConstraintLayout.
- Customize a dialog to present modal content to the user.
- Add navigation and menu options to the Toolbar.

Beyond the Basics

- Use Room to access a database to performantly store application data and explore the implications of Android's threading model.
- Master the main thread on Android. Understand how to communicate between threads to present data to the user.
- Seamlessly communicate with other applications using Android's intent system.
- Respond to device types to create an app that is optimized for both phones and tablets.
- Increase the potential user base of an app by supporting multiple languages and improving the accessibility.
- Explore the MVVM architecture, DataBinding, and unit testing to improve the quality of your code.
- Focus on the design of an application using Android's styles and themes APIs. Construct XML Drawables to further customize how users see an app.

Connecting to the Web

- Download, parse, and present data from a web service.
- Learn more about the Toolbar on Android and how to add custom views.
- Perform background work safely in your application.

- Receive and respond to system events.
- Display content from a web site in your application.

Adding Some Fun

- Learn how to create your own custom views when the default widgets just won't do.
- Receive and respond to raw touch events.
- Understand how to increase the delightfulness of an app through animation.