

# Kotlin Essentials

## Syllabus



Email: [hello@bignerdranch.com](mailto:hello@bignerdranch.com)

P: 770-817-6373

## 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

- Java Android developers who are interested in transitioning their projects to Kotlin
- 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
- Junior 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 conditionals and control flow statements and gain the additional support of control flow statements as assignable expressions
- Learn about nullable and non-nullable types and how to work with null safely
- Understand exceptions and how they may be safely handled

## Collections and Functions

- Learn how to represent series of data with mutable and read-only collection types
- See how to work with data using for loops, while loops, break, and continue
- 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, function references, and function inlining by example
- Use functional paradigms to clean up code, including map, flatMap, filter, and reduce, as well as how to define your own functional operators

## 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
- Understand extension functions to safely add behavior to classes you don't control

## Advanced Kotlin

- Learn how to interoperate with Java classes and supports Java compatibility
- Take a peek at the future of multithreading with coroutines
- Use Ktor to stand up a simple web server in Kotlin
- Leverage structured concurrency and the Flow API to build safe, efficient streams