+44 7989 401397
info@olsensoft.com

Kotlin Development (4 days)

Course overview

Kotlin is an object-oriented language that can run equally well on the Java Virtual Machine or natively on the operating system. Kotlin is also a functional language, and combines the features and benefits of OO and functional programming. This course provides a fast-paced introduction to the language for developers with experience Java or similar languages (e.g., C# or C++), and then delves deeper into idiomatic uses of Kotlin in practice. The course also shows how to use Kotlin Multiplatform to implement a common codebase that can run on any platform.

What you'll learn

- Writing and running Kotlin programs
- Object orientation in Kotlin
- Functional programming in Kotlin
- Implementing Domain-Specific Languages
- Using concurrency
- Overview of Kotlin multiplatform apps

Prerequisites

- Experience using Java or a similar contemporary object-oriented language

Course details

- Introduction to Kotlin: Overview of Kotlin; Kotlin tools; Writing a simple Kotlin program; Using the interactive shell; Using the online playground
- Core Kotlin Syntax: Types and variables; Literals; Decision making and looping; Exceptions; Enumerations
- Functions: Class-level and top-level functions; Local functions; Extension functions and properties; Variable-argument functions
- Object-Oriented Programming in Kotlin: Classes and objects; Properties, getters and setters; Construction techniques; Open, final, and abstract modifiers
- A Closer Look at Kotlin Types: Nullability; Primitive types; Data classes; Sealed classes; Singleton objects and companion objects; Arrays and collections
- Functional Programming: Overview of functional programming; Lambdas and member references; Functional APIs and collections; Lazy collection operations
- Going Further with Functional Programming: Higher-order functions; Inlining; Control flow; Recursion; Domain-Specific Languages
- The Kotlin Type System: Generics; Constraints; Covariance and contravariance; Defining and using annotations; Reflection

- Concurrency: Creating and synchronizing threads; Callbacks; Futures; Coroutines and channels
- Kotlin Multiplatform: Overview of Kotlin Multiplatform; Tooling up for Kotlin Multiplatform; Creating and running a Kotlin Multiplatform app