Kotlin Fundamentals

Course Duration: 24 Hours **Course code: Android**

1. Course Overview

This course provides a strong foundation in Kotlin programming, the official language for Android app development and a modern alternative to Java. Participants will learn Kotlin syntax, OOP concepts, functional programming features, and practical use cases for building robust applications.

2. What you'll learn?

- Understand Kotlin syntax and language fundamentals.
- Work with variables, data types, operators, and control flow.
- Apply object-oriented programming (OOP) principles in Kotlin.
- Use functions, higher-order functions, and lambdas.
- Explore collections, generics, and null safety.
- Handle exceptions and error management.
- Work with coroutines for concurrency and async programming.
- Apply Kotlin to real-world projects and Android development basics.

3. Target Audience

- Beginners in programming aiming to learn Kotlin.
- Java developers transitioning to Kotlin.
- Students and professionals preparing for Android development.
- Anyone interested in learning a modern JVM language.

4. Pre-Requisites

- Basic understanding of programming concepts recommended.
- No prior knowledge of Kotlin required.
- Familiarity with Java/C# helpful but not mandatory.

5. Course content

Course Overview

- Introduction to the course
- Course Outline

Module 1: Introduction to Kotlin

- Kotlin History
- How Kotlin Program Works?
- Kotlin Environment and the JVM
- Android Studio Environment
- Setting up an Android Studio Project
- Kotlin Hello World Program
- Kotlin Variables
- Kotlin Data Types
- Comments, Annotations
- Lab: Hello World in Kotlin using Android Studio

Module 2: Control Flow Statements

- Introduction
- If Statement
- If-Else Statement
- When Statement and Expression
- For Loops
- While Loops
- Do-While Loops
- Jump Expressions:
- Break Statement
- Continue Statement
- Return Statement

Module 3: Kotlin Classes

- Class Definitions
- Constructors (Named Parameters, Default Values)
- Secondary Constructors
- Simple Properties
- Instance Creation
- Member Functions
- Function Named Parameters
- Function Default Parameters
- Functions Returning Values
- Lab: Books and Bookshop App in Kotlin using Android Studio

Module 4: Objects, Flow of Control, and Further Properties

- Objects
- Flow of Control
- Self Reference: this
- Properties and Their Types
- Property Visibility
- Property Declaration Modifiers
- Nullability
- Nullable Operators
- Lazy Properties
- Instance Equality
- Curly Bracket Syntax
- Lab: Adding Behavior to the Bookshop

Module 5: Inheritance

- Inheritance in Kotlin
- Implementing Inheritance

- Rules for Overriding Functions
- Overriding Properties
- Rules for Polymorphic Variables
- Casting and Inheritance
- The super Variable
- Constructors and Inheritance
- Lab: Defining Inheritance for Book Class

Module 6: Abstract Classes and Interfaces

- Abstract Classes
- Defining an Abstract Class
- Declaring Abstract Functions
- Extending an Abstract Class
- Interfaces in Kotlin
- Interface Definitions
- Implementing Interfaces
- Delegation
- Lab: Implementing Abstract Classes and Interfaces

Module 7: Further Classes

- Four Types of Nested Classes
- Nested Classes
- Member Inner Classes
- Anonymous Classes / Objects
- Inline Classes
- Data Classes
- Data Classes and Destructuring Declarations
- Enumeration Support
- Lab: Implementing Nested Classes and Data Classes

Module 8: Containers and Generics

- Arrays
- Collections Library
- Mutable and Immutable (Read-Only) Collections
- Sets, Lists, and Maps
- Pairs and Triples
- Collection Iteration
- Lab: Working with Containers

Module 9: Functions

- What is Functional Programming?
- Kotlin as a Functional Language
- Defining Functions
- Function Objects
- Function Definitions
- Named Functions, Single Expression & Inline Functions
- Anonymous Functions and Lambdas
- Callable References
- Closures
- Functions and Methods
- Lab: Functional Programming Implementation

Module 10: Higher-Order Functions and Extension Functions

- Higher Order Functions Introduction
- Functions as Parameters
- Using typealias for Function Types
- Functions as Return Types
- Collections & FP: forEach
- Scope Functions (apply, let, also, run, with)
- Conditional Functions (takeIf, takeUnless)

- Extensions
- Extension Functions and Extension Properties
- Looping Over Collections
- Sorting Collections
- Grouping, Folding, Reducing, and Zipping Collections
- Lab: Processing Containers Using Higher Order Functions

Module 11: Packages and Project Structuring

- What is a Package?
- Kotlin Compiler
- Importing
- Kotlin Default Imports
- Visibility Modifiers
- Visibility Modifiers and Constructors
- Kotlin Modules

Module 12: Exception Handling

- Errors & Exceptions
- Exception Types in Kotlin
- Part of the Exception Hierarchy
- Exception Handling
- Kotlin and Checked Exceptions
- Try-Catch Expressions
- Finally and Returned Values
- Defining New Exceptions
- Chained Exceptions