

INFOSOFT IT SOLUTIONS

Training | Projects | Placements

Revathi Apartments, Ameerpet, 1st Floor, Opposite Annapurna Block, Infosoft It solutions,
Software Training & Development Institute, +91 - 9059683947 | +91 - 9182540872

Kotlin

Introduction

- 1. Why Kotlin is a Better Java

Setting Up the Tools

- 1. Using the Command Line Tools
- 2. Installing the JDK for Kotlin
- 3. Installing and Setting Up the IDE: IntelliJ
- 4. Installing the Plugins

Getting Started with Basic Kotlin Programming

- 1. Learning the Kotlin Architecture
- 2. Creating Variables in Kotlin
- 3. Understanding the Different Data Types in Kotlin
- 4. Using Strings in Kotlin
- 5. Understanding Immutability in Kotlin
- 6. Implementing Expressions, Statements, and Assignments
- 7. Using Nullable Variables in Kotlin for Null Safety
- 8. Handling Conditional Program Flow with "if" Statements
- 9. Handling Options Using "when" Statements
- 10. Exploring Kotlin's Advanced Capabilities for Conditional Statements

- 11. Using Arrays and Lists in Kotlin
- 12. Implementing Looping Constructs in Kotlin
- 13. Creating, Declaring, and Using Functions in Kotlin
- 14. Using Parameters and Returning a Value
- 15. Using Extension Functions
- 16. Using Inline Functions
- 17. Using Higher-order Functions

Object-Oriented Programming with Kotlin

- 1. Overview of Programming with Objects
- 2. Creating a Class
- 3. Adding Methods to Classes
- 4. Defining Basic Constructors
- 5. Using Named Parameters and Default Values
- 6. Using Open Classes to Allow Inheritance
- 7. Using Abstract Classes
- 8. Creating and Using Interfaces
- 9. Handling Overriding
- 10. Creating Data Classes
- 11. Using Singletons and Companion Objects
- 12. Using Enumerations
- 13. Grouping Classes into Packages
- 14. Using Imports
- 15. Implementing Information Hiding
- 16. Using Getters and Setters
- 17. Implementing Visibility Modifiers
- 18. Understanding Generics
- 19. Implementing a Generic Stack

- 20. Using Generic Functions

Exploring Kotlin's Interoperability with Java

- 1. Using Java from Kotlin
- 2. Using Kotlin from Java

Exploring Kotlin's Advanced Features

- 1. Understanding Operator Overloading
- 2. Using Annotations
- 3. Using Reflection
- 4. Implementing DSL Construction
- 5. Implementing Destructuring

Using Kotlin for Functional Programming

Hands-on: Converting an Android App from Java to Kotlin

- 1. Using Android Studio
- 2. Starting a New Project with Kotlin
- 3. Converting Existing Java Code to Kotlin
- 4. Using Android APIs with Kotlin

Troubleshooting