



# Introduction to Kotlin

## Course Overview:

The Introduction to Kotlin course will introduce students to the Kotlin programming language. Through a series of interactive lectures and labs, students will develop an awareness of the language and its interoperability with existing Java code and libraries. This begins by providing an overview of Kotlin and then moves into basic type systems. Students will also learn about functions, control flow, operators and expressions and classes, and inheritance. On Day Two students will explore abstract classes and interfaces, extension functions and lambdas. This course is intended for those with little or no prior knowledge of Kotlin.

**Course Duration:** This course will be delivered in 2 days.

### Prerequisites:

- Familiarity with Java 8 or a language with similar features.

### Course Objectives: After this course, you will be able to:

- Write simple programs using the Kotlin language
- Describe Java interoperability with existing code and libraries

### This course does include:

- Basic Kotlin language features and syntax
- Object oriented programming in Kotlin
- Introduction to Functional programming in Kotlin
- Interoperability between Java and Kotlin

**This course does NOT include:**

- Basic OO concepts; Encapsulation, Inheritance, Generalization, Instance vs Static Features
- Java 8 features like Lambdas and Streams

**Course Outline:**

- What is Kotlin?
  - History of Kotlin
  - First Kotlin Program
    - Comments
    - Function syntax
    - No semicolons
    - String interpolation
    - Print Statements
    - Packages/import statements
- Nuts and Bolts
  - Basic Type System
  - val v/s var
  - lateinit
  - Primitive types
    - Int, Double, Long → No implicit casting
    - list/array/set literals
  - Nullability
    - ? v/s !!
    - Elvis operators
  - Control Flow
    - If/Else
      - Smart casting

- Regular casting
  - if as an expression
- Loops
  - For/In loop
  - For each
  - repeat
  - Range
  - Map
  - Filter
- when statement
- Operators and Expressions
- Exceptions
- Classes and Inheritance
  - Any and Nothing
  - Primary/Secondary constructors
  - Init block
  - this keyword
  - Creating instances of a class
  - Properties and Fields
  - Visibility modifiers (private, public, protected, internal)
  - object as singleton
  - Companion objects
  - Abstract classes
  - Interfaces
  - Inner classes
  - Data classes
- Functions
  - Defining functions
  - Calling functions
  - Function parameters

- Default and named parameters
- Overriding functions
- Extension functions
- Extension properties
- Lambdas
- Lambdas with receivers
- Scope Functions
  - let, apply, also, run, with
- Kotlin “Conventions”
  - aka Operator Overloading
- Java Interoperability
  - Kotlin from Java
  - Java from Kotlin