

**Modern JVM Development with Kotlin, Microservices, and Kafka**

**Course Number:** MBL-226
**Duration:** 3 days

**Overview**

This Modern JVM Development with Kotlin, Microservices, and Kafka training course teaches attendees the Kotlin skills and Key Libraries needed to create Microservices in Kotlin. Students will learn how to create and deploy a set of services using basic Docker and Kubernetes. In addition, participants will learn how to deploy services to cloud providers like AWS.

**Prerequisites**

Students must have at least five years of experience in software development using either Java or C#. They should be familiar with the basics of distributed system design and REST, ideally by working on projects using frameworks like Spring Boot and/or the .NET Web API.

**Materials**

All Kotlin training attendees receive comprehensive courseware.

Course outline and materials are copyrighted and owned by [Instil Software](https://instil.co/).

**Software Needed on Each Student PC**

* Windows or Mac minimum 8 GB RAM
* Android Studio installed
* Provided lab files from Accelebrate

**Objectives**

* Program using all the features of the Kotlin language
* Write build files using the new Kotlin DSL in Gradle V5
* Create your own Kotlin DSL’s for custom configuration
* Use Arrow to enhance Kotlin’s support for Functional Types
* Apply TDD, BDD, and Property-Based Testing in Kotlin
* Write RESTful Microservices using Ktor and HTTP4K
* Link Microservices via messaging and event streams in Kafka
* Deploy sets of services via both Docker and Kubernetes
* Access NoSQL data stores from services via reactive APIs

**Outline**

* Introduction
* The Kotlin Language and Key Libraries
	+ Introduction to the Kotlin language and ecosystem
	+ The OO and FP components of the language
	+ Creating DSL’s and ensuring interoperability with legacy Java code
	+ Libraries: KotlinTest, Gradle 5, Arrow, and TornadoFX
* Creating Microservices in Kotlin
	+ Ktor and HTTP4K frameworks to create RESTful services
	+ Unit Testing
	+ Integration via Kafka event broker and NoSQL databases such as Redis
* Sample Project
	+ Create and deploy a set of services to implement a sample system.
	+ Docker and Kubernetes basics
	+ Deploying to cloud providers like AWS
* Conclusion