Section 1: Course Introduction

Section 2: Why Functional Programming and Why to Bother

* Course Resources

Section 3: Lambda and Functional Interface

* First Cute Lambda
* Functional Interfaces
* How Lambda Works Under the Hood
* Imperative V/s Declarative

Summary

Section 4: Practice Lambda with different signatures

* Introduction
* Practice Lambda - 1
* Practice Lambda - 2
* Practice Lambda - 3
* Practice Lambda - 4
* Summary

Section 5: Predefined Functional Interfaces and Practice

* Introduction
* Predefined Functional Interfaces
* Generic Functional Interfaces
* Predicate
* Consumer
* Supplier
* Function
* UnaryOperator
* BiFunction
* BinaryOperator
* Summary

Section 6: Method and Constructor Reference

* Introduction
* Method Reference
* Constructor Reference
* Summary

Section 7: Optional for Null Pointer

* Introduction
* Dangers of Null
* Introducing Optional
* Getting the values Back
* Operators Part - 1
* Operators Part - 2
* Summary

Section 8: Functional Programming in Depth

* Introduction
* What is Functional Programming
* Functional Programming Concepts
* Functions as First class Citizens
* No Side Effects - Pure Functions
* Higher Order Functions
* Referencial Transparency
* Functional Programming Techniques
* Designing APIs In Functional Way
* Chaining
* Composition
* Closures
* Currying
* Lazy Evaluation
* Tail Call Optimisation aka TCO
* Summary

Section 9: Design Patterns in Functional Way

* Introduction
* Iterator Design Pattern
* Strategy Design Pattern
* Decorator Design Pattern
* Creating Fluent Interfaces
* Factory Method Pattern
* Builder Design Pattern
* Command Design Pattern
* Summary

Section 10: Streams and Parallel Streams in depth

* Preview
* Streams Introduction
* Observing the Stream
* Stream Pipeline
* Streams are not data Containers
* Filter Operations
* Map Operations
* Reduce Operations
* Streams are Lazy
* Numeric Streams
* Numeric Streams - Methods
* Bounded Streams
* Infinite Streams
* Stream.of and FlatMap
* Parallel Streams
* Stateless and stateful operations
* Setting Parallelism
* Summary and BrainMapping

Section 11: Creating Streams from Custom Source and Understanding Spliterator Pattern

* Introduction
* What Is Spliterator
* Spliterator Characteristics
* Custom Spliterator
* Summary

Section 12: Collectors for Data Processing in Depth

* Introduction
* What are Collectors
* Terminal Analogous Collectors-1
* Terminal Analogous Collectors-2
* Terminal Analogous Collectors-3
* Downstream Collectors
* Cascading Collectors
* Summary

Section 12: Collectors for Data Processing in Depth

* Introduction
* What are Collectors
* Terminal Analogous Collectors-1
* Terminal Analogous Collectors-2
* Terminal Analogous Collectors-3
* Downstream Collectors
* Cascading Collectors
* Summary

Section 13: Creating Your Own Collectors

* Introduction
* How Collectors Works Internally
* Creating Your Own Collectors - 1
* Creating Your Own Collectors - 2
* Summary

Section 14: Playing with Collections and Map in Functional style

* Preview
* List
* Set
* Map
* Summary

Section 15: Data Structures Functional Way

* Introduction
* Functional Data Structure - What and why
* List
* Queue
* Tree
* Map
* Summary