Java with Springboot Advance Training

Day - 1

-
  - Java Fundamentals - Object-oriented Design
-
  - Basics
  - Coherence, Coupling, and God Classes
  - What Is an Object?
-
  - Module Overview
  - More Class Relationships
  - Encapsulation
  - Assigning Responsibilities
  - Demo: Exercising the System
  - Demo: the Single Responsibility Principle
  - Demo: Defending Encapsulation
  - Abstraction: Encapsulation or Inheritance?

-
  - Module Overview
  - Don't Repeat Yourself
  - The Liskov Substitution Principle
  - Overriding and Dynamic Dispatch
  - Demo: Dynamic Dispatch
  - Using Overridden Methods
  - Understanding Inheritance
  - Abstract Classes and the Open-closed Principle
-
  - Problems with println()
  - Money and Delegation: Getters and Setters Are Evil
-
  - The Is-a Test
  - Simple Extension and Template Method
  - Delegation Under Inheritance: Shape
  - Liskov Substitution: Stack, Ver. 1
  - Fragile Base Classes: Stack, Ver. 2

-
  - Module Overview
  - Interfaces What Are They, and Why Do We Need Them?
  - Demo: Paying through an Interface
  - Demo: The Fragile Base Class Problem
  - Delegation and the Decorator Pattern
  - The Strategy Pattern
  - The Interface Segregation Principle
  - The Dependency Inversion Principle
-
  - Getting Started
  - Learning Path
  - Spring Boot Overview
  - Features of Spring Boot
  - Summary

Day - 2

-
  - Defining Contracts
  - Fixing Fragility
  - Dependency Inversion
-
  - The Open Closed Principle
  - Builder
-
  - Overview
  - Demo: Spring Initializr
  - Demo: Spring Boot CLI
  - Demo: Auto Configuration
  - Spring Boot Annotations
  - Spring Boot Properties
  - Spring Boot Profiles
  - Summary

-
  - Overview
  - H2 Database
  - Why Use Templating Engines?
  - ORM with JPA
  - Entities
  - Demo: H2 Integration
  - Demo: H2 Console
  - Demo: Partials
  - Summary
-
  - Dependency Injection
  - Summing Up
-
  - Course Overview
  - What is Object Orientation?
  - From Use Cases to Conceptual Classes
  - Class Relations
  - Deriving Software Classes, Part 1
  - Deriving Software Classes, Part 2
  - Demo: Instance and Static Methods

-
  - Overview
  - MVC Design Pattern
  - Demo: MVC in Action
  - Packaging and Deployment
  - Demo: Uber Jar
  - Summary
-
  - Overview
  - REST Architecture
  - Demo: RESTful Web Application
  - Demo: Testing with URL
  - Response Formats
  - Exception Handling
  - Demo: Exception Handling
  - Summary
-
  - Overview
  - Testing Overview
  - Unit Testing
  - Integration Tests