



# Spring Boot

## Table of content:

### Introduction

- How to make the best of this course
- Full Stack Developer Learning Path

### Software Setup

- Install Java
- Install STS
- Configure JDK in STS
- Install MySQL and MySQL workbench
- Launch MySQL Workbench
- Windows Only -Install MySQL
- Windows Only - Install MySQL Workbench
- Install Postman
- Download the completed projects
- Download Assignment Solutions
- Slides Used In The Course

### Basics

- The problems with the traditional spring application
- Spring Boot Features
- Spring Boot In Action
- Basics

### First Project

- Create a Spring Boot Project
- Spring Boot Starters in POM
- @SpringBootApplication
- @SpringBootTest
- Do Dependency Injection
- Testing
- Spring Boot In Action
- Spring Boot Project
- Spring Data JPA

### Introduction

- Create the project
- Add Spring Data JPA Dependency
- Maven Dependencies
- Create Model Class
- Create the Repository
- Testing
- Show SQL
- Spring Data JPA

- Spring Data JPA

## Create REST CRUD API

- Rest Introduction
- REST Web Services using Spring Boot
- Create the DB Table
- Create the project
- Create the Model and Repository
- Mark with JPA annotations
- Create the Controller
- Get Single Product
- Create Update and Delete
- Configure The Data Source
- Run the application
- Running outside STS
- How Spring Boot Auto Configuration Works ?
- Configure Context Path
- Test Create FindAll
- Test FindById and Update
- Changing the Embedded Server
- Create REST CRUD API
- Spring Web and REST API
- Add context-path and change the server

## Creating a REST Client

- Introduction
- RestTemplate GET method
- POST method
- PUT Method
- REST Client

## Profiles

- Introduction
- Configure REST URL
- Create and Activate a Profile
- Activate Profile through VM argument
- Profiles

## Logging

- Use Logging
- Log to file
- Change Log Level

## Health Checks and Metrics

- Introduction
- Enabling health checks
- Expose Health Details
- Add Build Info

- Expose other endpoints
- Custom Health Information
- Enable health checks and customize info endpoint
- Health Checks and Metrics

### Spring Security

- Enable Security

### Thymeleaf

- Introduction
- Thymeleaf in action
- Special Syntax
- Sending data to template
- Create a template
- Disable Cache
- Sending Object Data
- Rendering multiple records
- Create a HTML Form
- Process Form Data
- Templates

### Database Caching

- Introduction
- Steps to enable caching
- Add Maven Dependencies
- Configure Caching
- Enable and Use Caching
- Caching in action
- Caching

### Spring Batch

- Introduction
- Batching API
- Create Project
- Create Reader
- Implement Processor Writer and Listener
- Configure the beans
- Configure the Step
- Configure the Job
- Write a Test
- Troubleshooting Updates
- Batching in action
- Configuring chunk size
- CSV to Database
- Create the project and model
- Create the CSV
- Implement Reader
- Implement Processor

- Implement Writer
- Configure the DataSource
- Configure Step and Job
- Write a Test
- CSV to DB in action
- Spring Batch

#### Unit Testing using MockMvc

- Introduction
- Create the test
- Use MockMvc
- Expect Results
- Refactoring
- Test Create
- Test Update and Delete

#### Messaging and Spring JMS

- What is Messaging
- Why Messaging
- Two Message Models
- What is JMS
- Spring JMS
- Install Apache ActiveMQ
- Create the project and message sender
- Create Message Listener
- Test
- More about Spring JMS
- Spring JMS

#### Swagger REST Documentation

- Introduction
- Spring boot and Swagger
- Swagger in action
- Change URI
- Swagger annotations
- Advanced Config Reference

#### Validations

- Introduction
- Validations in actions

#### REST File Upload and Download

- Introduction
- Completed project for download
- Create the project
- Implement File Upload
- Test
- Implement and Test File Download

- RestTemplate APIs
- Upload Client
- Download Client

## Spring Reactive Programming

- Introduction
- Create Reactive Project
- Mono and Flux
- Use a Consumer
- Few More methods
- Use a Subscriber
- Configure Batching
- Implement the usecase
- Spring Webflux
- Create a reactive rest api
- Create a Web Layer

## Use Reactive MongoDB

- Introduction
- MongoDB Introduction
- Install Mongo
- Mongo Commandline
- Mongo Reactive API
- Create the project
- Create model and repository
- Implement the API

## Reactive Testing

- Introduction
- Create First Test
- expectNextCount
- assertNext
- Couple more methods
- Unit Testing APIs
- Write a Unit Test
- Unit Test VaccineController
- Mockito Verify
- Unit Test addProduct
- Unit Test getProducts

## Rsockets

- Introduction
- Four Models
- Rsocket Server API
- Create the RSocket Server Project
- Implement Request Response Endpoint
- Client API
- Setup the client project

- Create the RSocket Client
- Use Logging
- Test Request Response
- Fire and Forget Server
- Fire and Forget Client
- Request Stream Server
- Request Stream Client

#### Using YAML Configuration

- Introduction
- Switch to YAML

