Apache Spark with Java & Spring

32 hours

Course Overview:

Moving forward to work with Spark you raise a lot of question:

- Do you fear Big Data? Do you think BigData is such a complicated topic that it requires you to master many new concepts, technologies and paradigms?
- You may have heard that to code in Spark, you need to be a Scala or Python Ninja?
- Perhaps a friend told you that the Spark Java API is limited, bulky and inconvenient?
- Inversion of Control, design patterns, Java frameworks such as Spring, JUnit or Maven/Gradle are not your friends anymore?
- Are you concerned that all your previous Java experience is useless in the world of Big Data technologies?

We have good news for you!

Using Apache Spark - the open source Big Data processing framework, you can relax and start using your existing development expertise to solve Big Data problems.

Writing Spark with Java can be very elegant and very familiar for developers like you, with strong JAVA background.

This training is custom made for Java developers who want to solve Big Data challenges in the most efficient and simple way, using one the world's most advanced and popular BigData framewoks – Spark!

As part of this training, we will review Spark's comprehensive APIs and cover its capabilities. We will also demonstrate that you can do anything you want with Spark and still use the same techniques and knowledge you've gained as a Java master.

We will learn how to build our own infrastructure with Spring framework in order to make the development process in Spark much easier and comfortable.

Who Should Attend:

Experienced Java developers

Course Contents:

Introduction to Big Data

- Data locality
- Map reduce
- New approach
- Hadoop implementations
- Cloudera solutions overview

Hadoop ecosystem

- HDFS
- HBase
- YARN

- Hue
- Sqoop
- Flume
- Hive
- Impala
- Oozie

Minimum SCALA you still need to know, for writing spark on Java

- Scala syntax
- Singleton objects
- Tuples

Java 8 overview

- Callback method pattern
- Lambda expression
- Method reference
- Functional Programming style
- New methods and classes in Java 8

Lombok

- Annotation Processors
- Pojo annotations
- Functional annotations
- Delombok

Spring

- IOC / Inversion of control concepts
- Bean declarations
- Spring context
- Profiles / @Conditional
- Spring Test
- BeanPostProcessors – writing custom annotations for spark

Spark

- Spark intro
- Running spark in cluster (spark submit)
- Spark API
- RDD
    - Transformations
    - Actions
    - Creation
    - Testing
    - File types (SCV, parquet)
- Comparing RDD API (Scala vs Java)

- Partitioning
- Broadcast and accumulators
- Developing @Broadcast annotation
- DataFrames API
    - SqlContext
    - Working with schema files
    - Json, avro
    - DataFrame methods
    - functions API
    - writing etl process
    - creating schema
    - creating dataframe from rdd and from SCV files
    - sql usage on dataframes
    - developing custom annotations with Spring for dataframes
- Spark and Junit
- Writing tests for spark with spring runner
- Separating production from test environment with spring @Conditional
- Full stack application (from spark to rest service)
- Migration to Spark 2
    - Datasets
    - SparkSession
- Spark streaming introduction