Mr. Kalyan, Big Data Solution Architect,
Apache Contributor, 11+ years of IT exp, 7+ years of Big Data exp,
Cloudera CCA175 Certified Consultant, IIT Kharagpur, Gold Medalist

Hadoop Course Content (Hadoop-1.x, -2.x & -3.x) (Development and Administration)

Introduction to Big Data and Hadoop

- **Big Data**
  - What is Big Data?
  - Why all industries are talking about Big Data?
  - What are the issues in Big Data?
    - Storage
      - What are the challenges for storing big data?
    - Processing
      - What are the challenges for processing big data?
  - What are the technologies support big data?
    - Hadoop
    - Spark
    - Data Bases
      - Traditional
      - NO SQL
- **Hadoop**
  - What is Hadoop?
  - Why Hadoop?
  - History of Hadoop
  - Hadoop Use cases
  - Advantages and Disadvantages of Hadoop
- Importance of Different Ecosystems of Hadoop
- Importance of Integration with other Big Data solutions
- Big Data Real time Use Cases
- Batch vs Real Time Big Data Analytics
- Real Time Analytics
  - Streaming Data – Storm / Kafka / Flume
  - In Memory Data - Spark

HDFS (Hadoop Distributed File System)

- **HDFS architecture**
  - **Name Node**
    - Importance of Name Node
    - What are the roles of Name Node
    - What are the drawbacks in Name Node
  - **Secondary Name Node**
    - Importance of Secondary Name Node
    - What are the roles of Secondary Name Node
    - What are the drawbacks in Secondary Name Node
  - **Data Node**

- Importance of Data Node
- What are the roles of Data Node
- What are the drawbacks in Data Node

❖ **Data Storage in HDFS**
- How blocks are storing in DataNodes
- How replication works in Data Nodes
- How to write the files in HDFS
- How to read the files in HDFS

❖ **HDFS Block size**
- Importance of HDFS Block size
- Why Block size is so large?
- How it is related to MapReduce split size

❖ **HDFS Replication factor**
- Importance of HDFS Replication factor in production environment
- Can we change the replication for a particular file or folder
- Can we change the replication for all files or folders

❖ **Accessing HDFS**
- CLI(Command Line Interface) using hdfs commands
- Java Based Approach

❖ **HDFS Commands**
- Importance of each command
- How to execute the command
- Hdfs admin related commands explanation

❖ **Configurations**
- Can we change the existing configurations of hdfs or not?
- Importance of configurations

❖ **How to overcome the Drawbacks in HDFS**
- Name Node failures
- Secondary Name Node failures
- Data Node failures

❖ Where does it fit and Where doesn't fit?

❖ Exploring the Apache HDFS Web UI

❖ **How to configure the Hadoop Cluster**
- How to add the new nodes (Commissioning)
- How to remove the existing nodes (De-Commissioning)
- How to verify the Dead Nodes
- How to start the Dead Nodes

❖ **Hadoop 2.x.x version features**
- Introduction to Namenode federation
- Introduction to Namenode High Availabilty with NFS
- Introduction to Namenode High Availabilty with QJM

❖ Difference between Hadoop 1.x, Hadoop 2.x and Hadoop 3.x versions

MAPREDUCE

- **Map Reduce architecture**
  - **JobTracker**
    - Importance of JobTracker
    - What are the roles of JobTracker
    - What are the drawbacks in JobTracker
  - **TaskTracker**
    - Importance of TaskTracker
    - What are the roles of TaskTracker
    - What are the drawbacks in TaskTracker
  - Map Reduce Job execution flow
- **Data Types in Hadoop**
  - What are the Data types in Map Reduce
  - Why these are importance in Map Reduce
  - Can we write custom Data Types in MapReduce
- **Input Format's in Map Reduce**
  - Text Input Format
  - Key Value Text Input Format
  - Sequence File Input Format
  - NLine Input Format
  - Importance of Input Format in Map Reduce
  - How to use Input Format in Map Reduce
  - How to write custom Input Format's and its Record Readers
- **Output Format's in Map Reduce**
  - Text Output Format
  - Sequence File Output Format
  - Importance of Output Format in Map Reduce
  - How to use Output Format in Map Reduce
  - How to write custom Output Format's and its Record Writers
- **Mapper**
  - What is mapper in Map Reduce Job
  - Why we need mapper?
  - What are the Advantages and Disadvantages of mapper
  - Writing mapper programs
- **Reducer**
  - What is reducer in Map Reduce Job
  - Why we need reducer ?
  - What are the Advantages and Disadvantages of reducer
  - Writing reducer programs
- **Combiner**
  - What is combiner in Map Reduce Job
  - Why we need combiner?
  - What are the Advantages and Disadvantages of Combiner
  - Writing Combiner programs
- **Partitioner**
  - What is Partitioner in Map Reduce Job
  - Why we need Partitioner?

- What are the Advantages and Disadvantages of Partitioner
- Writing Partitioner programs

❖ **Distributed Cache**
- What is Distributed Cache in Map Reduce Job
- Importance of Distributed Cache in Map Reduce job
- What are the Advantages and Disadvantages of Distributed Cache
- Writing Distributed Cache programs

❖ **Counters**
- What is Counter in Map Reduce Job
- Why we need Counters in production environment?
- How to Write Counters in Map Reduce programs

❖ **Importance of Writable and Writable Comparable Api's**
- How to write custom Map Reduce Keys using Writable
- How to write custom Map Reduce Values using Writable Comparable

❖ **Joins**
- **Map Side Join**
  - What is the importance of Map Side Join
  - Where we are using it
- **Reduce Side Join**
  - What is the importance of Reduce Side Join
  - Where we are using it
- What is the difference between Map Side join and Reduce Side Join?

❖ **Compression techniques**
- Importance of Compression techniques in production environment
- Compression Types
  - NONE, RECORD and BLOCK
- Compression Codecs
  - Default, Gzip, Bzip2, Snappy and LZO
- Enabling and Disabling these techniques for all the Jobs
- Enabling and Disabling these techniques for a particular Job

❖ **Map Reduce Schedulers**
- FIFO Scheduler
- Capacity Scheduler
- Fair Scheduler
- Importance of Schedulers in production environment
- How to use Schedulers in production environment

❖ **Map Reduce Programming Model**
- How to write the Map Reduce jobs in Java
- Running the Map Reduce jobs in local mode
- Running the Map Reduce jobs in pseudo mode
- Running the Map Reduce jobs in cluster mode

❖ **Debugging Map Reduce Jobs**
- How to debug Map Reduce Jobs in Local Mode.
- How to debug Map Reduce Jobs in Remote Mode.

❖ **Data Locality**
- What is Data Locality?
- Will Hadoop follows Data Locality?

❖ **Speculative Execution**

- What is Speculative Execution?
- Will Hadoop follows Speculative Execution?

- **Map Reduce Commands**
  - Importance of each command
  - How to execute the command
  - Mapreduce admin related commands explanation
- **Configurations**
  - Can we change the existing configurations of mapreduce or not?
  - Importance of configurations
- Writing Unit Tests for Map Reduce Jobs
- Configuring hadoop development environment using Eclipse
- Use of Secondary Sorting and how to solve using MapReduce
- How to Identify Performance Bottlenecks in MR jobs and tuning MR jobs.
- Map Reduce Streaming and Pipes with examples
- Exploring the MapReduce Web UI

YARN (Next Generation Map Reduce)

- What is YARN?
- What is the importance of YARN?
- Where we can use the concept of YARN in Real Time & it's powered projects
- What is difference between YARN and Map Reduce
- **Yarn Architecture**
  1. Importance of **Resource Manager**
  2. Importance of **Node Manager**
  3. Importance of **Application Manager**
  4. Yarn Application execution flow
- Installing YARN on both windows & Linux
- Exploring the YARN Web UI
- Examples on YARN

Apache PIG

- Introduction to Apache Pig
- Map Reduce Vs Apache Pig
- SQL Vs Apache Pig
- Different data types in Pig
- **Modes Of Execution in Pig**
  - Local Mode
  - Map Reduce Mode
- **Execution Mechanism**
  - Grunt Shell
  - Script
  - Embedded
- **UDF's**
  - How to write the UDF's in Pig
  - How to use the UDF's in Pig
  - Importance of UDF's in Pig
- **Filter's**
  - How to write the Filter's in Pig

- How to use the Filter's in Pig
- Importance of Filter's in Pig

❖ **Load Functions**
- How to write the Load Functions in Pig
- How to use the Load Functions in Pig
- Importance of Load Functions in Pig

❖ **Store Functions**
- How to write the Store Functions in Pig
- How to use the Store Functions in Pig
- Importance of Store Functions in Pig

❖ Transformations in Pig
❖ How to write the complex pig scripts
❖ How to integrate the Pig and Hbase

Apache HIVE

❖ Hive Introduction
❖ **Hive architecture**
- Driver
- Compiler
- Optimizer
- Semantic Analyzer

❖ Hive Query Language(Hive QL)
❖ SQL VS Hive QL
❖ Hive Installation and Configuration
❖ Hive DLL and DML Operations
❖ **Hive Services**
- CLI
- Hiveserver
- Hwi

❖ **Metastore**
- embedded metastore configuration
- external metastore configuration

❖ **UDF's**
- How to write the UDF's in Hive
- How to use the UDF's in Hive
- Importance of UDF's in Hive

❖ **UDAF's**
- How to use the UDAF's in Hive
- Importance of UDAF's in Hive

❖ **UDTF's**
- How to use the UDTF's in Hive
- Importance of UDTF's in Hive

❖ How to write a complex Hive queries
❖ What is Hive Data Model?
❖ **Partitions**
- Importance of Hive Partitions in production environment
- Limitations of Hive Partitions
- How to write Partitions

- **Buckets**
  - Importance of Hive Buckets in production environment
  - How to write Buckets
- **SerDe**
  - Importance of Hive SerDe's in production environment
  - How to write SerDe programs
- How to integrate the Hive and Hbase
- How to integrate the Hive and Spark

Cloudera Impala

- Introduction to Impala
- Impala Examples
- Hive vs Impala

Apache Zookeeper

- Introduction to zookeeper
- Pseudo mode installations
- Zookeeper cluster installations
- Basic commands execution

Apache HBase

- HBase introduction
- HBase usecases
- **HBase basics**
  - Importane of Column families
  - Basic CRUD operations
    - create
    - scan / get
    - put
    - delete / deleteall / drop
  - Bulk loading in Hbase
- **HBase installation**
  - Local mode
  - Psuedo mode
  - Cluster mode
- **HBase Architecture**
  - HMaster
  - HRegionServer
  - Zookeeper
- **Mapreduce integration**
  - Mapreduce over HBase

Apache Phoenix

- Introduction to Phoenix
- Installing Phoenix
- Integrating with Hbase
- Comparing Hbase & Phoenix
- Practice on Phoenix examples

Apache Cassandra

- Introduction to Cassandra
- Installing Cassandra
- Practice on Cassandra examples

MongoDB

- Introduction to MongoDB
- Installing MongoDB
- Practice on MongoDB examples

Apache SQOOP

- Introduction to Sqoop
- MySQL client and Server Installation
- Sqoop Installation
- How to connect to Relational Database using Sqoop
- Examples on Import and Export Sqoop commands

Apache FLUME

- Introduction to flume
- Flume installation
- Flume Architecture
  - Agent
  - Sources
  - Channels
  - Sinks
- Practice on Flume examples

Apache Kafka

- Introduction to Kafka
- Installing Kafka
- Practice on Kafka examples

Apache OOZIE

- Introduction to oozie
- Oozie installation
- Executing different oozie workflow jobs
- Monitering Oozie workflow jobs

Pre-Requisites for this Course

- Java Basics like OOPS Concepts, Interfaces, Classes and Abstract Classes etc (Free Java classes as part of the course)
- SQL Basic Knowledge (Free SQL classes as part of the course)
- Linux Basic Commands (Provided in our blog)

Spark and Scala Content as part of Hadoop Course

Introduction of Scala

- What is Scala?
- Why Scala?
- Advantages of Scala?
- Using the Scala REPL(Read Evaluate print loop)
- What is Type Inference
- Interoperability between Scala and Java

Scala using Command Line

- Installing Java & Scala
- Interactive Scala
- Writing Scala Scripts
- Compiling Scala Programs

Basics of Scala

- Defining Variables
- Defining Functions
- String Interpolation
- IDE for Scala

Scala Type Less, Do More

- Semicolons
- Variable Declarations
- Method Declarations
- Type Inference
- Immutability
- Operators
- Precedence Rules
- Literals
- Arrays, Lists, Maps, Tuples

Expressions and Conditionals

- If expressions
- If-Else expressions
- For Loops
- While Loops
- Do-While Loops
- Conditional Operators
- Pattern Matching

Functional Programming in Scala

- What is Functional Programming?
- Different types of functions in Scala
  - Anonymous functions
  - Named functions
  - Curried functions
- Recursions

Object-Oriented Programming in Scala

- How to create a Class
- How to create a Case Class
- How to create a Object
- Constructors in Scala
- Fields in Classes

Introduction to Spark

- What is Spark
- Why Spark
- Who Uses Spark
- Brief History of Spark
- Storage Layers for Spark
- Spark vs Mapreduce
  - Why Spark is 100 times faster than MapReduce
- **Difference between Spark-1.x and Spark-2.x**
- **Unified Stack of Spark**
  - Spark Core
  - Spark Sql
  - Spark Streaming
  - Spark MLLib
  - Spark GraphX
- **Spark Architecture explanation**
  - Master Slave architecture
  - Spark Driver
  - Workers
  - Executors
- **Installation of Spark in different modes**
  - Local mode
  - Pseudo mode
- Introduction Spark WebUI
- Spark Job Execution flow

Basics of Spark

- Creating the **Spark Context**
- Creating the **Spark Conf**
- Creating the **Spark Session**
- **Caching** Overview
- Distributed Persistence
- Deploying Applications with **spark-submit**

Resilient Distributed Dataset (RDD)

- What is RDD
- Creating RDDs
  - Using collections
  - Using datasets (text, csv, tsv, ...)
- **RDD Operations**
  - Transformations
  - Actions
- Working with Key/Value Pairs
- Creating Pair RDDs
- **Transformations on Pair RDDs**
  - Aggregations
  - Joins
  - Sorting Data

Loading and Saving Your Data

- Loading Data using RDD
- Saving Data using RDD

Apache Spark SQL

- What is the importance of **Spark SQL**
- Working with Spark SQL **DataSets**
- Working with Spark SQL **DataFrames**
- Practice on Spark **SQL Context**
- Practice on Spark **SparkSession**
- Practical examples on **Spark SQL**
  - Aggregations
  - Joins
  - Sorting Data
- **Spark SQL Integrations**
  - Spark and Hive interation
  - Spark and RDBMS interation
- **Processing different files using Spark SQL**
  - Text, Json, Csv, Tsv, Parquet

Administration topics:

- **Hadoop Installations (Windows & Linux)**
  - Local mode (hands on installation on ur laptop)
  - Pseudo mode (hands on installation on ur laptop)
  - Cluster mode (hands on 4**0+** node cluster setup in our lab)
  - Nodes Commissioning and De-commissioning in Hadoop Cluster
  - Jobs Monitoring in Hadoop Cluster
  - Fair Scheduler (hands on installation on ur laptop)
  - Capacity Scheduler (hands on installation on ur laptop)
- **Hive Installations**
  - Local mode (hands on installation on ur laptop)
    - With internal Derby
  - Cluster mode (hands on installation on ur laptop)
    - With external Derby
    - With external MySql
  - Hive Web Interface (HWI) mode (hands on installation on ur laptop)
  - Hive Thrift Server mode (hands on installation on ur laptop)
  - Derby Installation (hands on installation on ur laptop)
  - MySql Installation (hands on installation on ur laptop)
- **Pig Installations**
  - Local mode (hands on installation on ur laptop)
  - Mapreduce mode (hands on installation on ur laptop)
- **Hbase Installations**
  - Local mode (hands on installation on ur laptop)
  - Psuedo mode (hands on installation on ur laptop)
  - Cluster mode (hands on installation on ur laptop)
    - With internal Zookeeper
    - With external Zookeeper
- **Zookeeper Installations**
  - Local mode (hands on installation on ur laptop)
  - Cluster mode (hands on installation on ur laptop)
- **Sqoop Installations**
  - Sqoop installation with MySql (hands on installation on ur laptop)
  - Sqoop with hadoop integration (hands on installation on ur laptop)
  - Sqoop with hive integration (hands on installation on ur laptop)
  - Sqoop with hbase integration (hands on installation on ur laptop)
- **Flume Installation**
  - Psuedo mode (hands on installation on ur laptop)
- **Oozie Installation**
  - Psuedo mode (hands on installation on ur laptop)

- **Advanced Technologies Installations**
  - Spark
  - Cassandra
  - MongoDB
  - Kakfa
  - Mahout
- **Cloudera Hadoop Distribution installation**
- **HortonWorks Hadoop Distribution installation**

Advanced and New technologies architectural discussions

- Spark / Flink (Real time data processing)
- Storm / Kafka / Flume (Real time data streaming)
- Cassandra / MongoDB (NOSQL database)
- Solr (Search engine)
- Nutch (Web Crawler)
- Lucene (Indexing data)
- Mahout (Machine Learning Algorithms)
- Ganglia, Nagios (Monitoring tools)
- Cloudera, Hortonworks, MapR, Amazon EMR (Distributions)
- How to crack the Cloudera / Hortonworks certification questions

Cloudera Distribution

- Introduction to Cloudera
- Cloudera Installation
- Cloudera Certification details
- How to use cloudera hadoop
- What are the main differences between Cloudera and Apache hadoop

Hortonworks Distribution

- Introduction to Hortonworks
- Hortonworks Installation
- Hortonworks Certification details
- How to use Hortonworks hadoop
- What are the main differences between Hortonworks and Apache hadoop

Amazon EMR

- Introduction to Amazon EMR and Amazon EC2
- How to use Amazon EMR and Amazon EC2
- Why to use Amazon EMR and Importance of this

Hadoop ecosystem Integrations:

- Hive and Spark integration
- Hive and HBase integration
- Pig and HBase integration
- Sqoop and RDBMS integration
- Hbase and Phoenix integration

- Flume and Phoenix integration
- Kakfa and Phoenix integraion

Free Big Data Workshops:

- Spark & Scala
- Cassandra
- MongoDB
- Search engine & E-commerce solutions
- Big Data Analytics (R, Mahout, Spark ML)

Real Time Big Data Projects

- We willl be sharing **Weekly based Big Data Assignments**
- We willl be sharing **End-to-End Big Data Projects**
- We are providing **Big Data Project Practice on Our Lab**
- We are providing **Important Recorded Videos on Our YouTube Channel**
- Any information search in **Google / YouTube** by keyword is **'Kalyan Hadoop'**

What we are offering to you:

- Hadoop installation on both **Windows & Linux**
- **Free Weekly Online Hadoop Certification**
- **Real Time Big Data projects will be shared**
- **Free Big Data Workshops** on **new & advanced technologies**
- Hands on MapReduce programming around **20+** programs these will make you to perfect in MapReduce both concept-wise and programmatically
- Hands on **5 POC's** will be provided (These POC's will help you perfect in Hadoop and it's ecosystems)
- Hands on practical 4**0+** Node hadoop cluster setup in our Lab.
- Well documented **Hadoop materia**l with all the topics covering in the course
- Well documented **Hadoop blog** contains frequent interview questions along with the answers and latest updates on Big Data technology.
- Discussing about **hadoop interview questions & answers** daily base.
- **Resume preparation** with POC's or Project's based on your experience.