Syllabus for HADOOP/BIGDATA (HADOOP 2)

Course Duration for Hadoop/Big Data Course

- **8 Weekends**

Objective For Hadoop/Big Data Course

- **To become a complete Hadoop/Big Data Professional**

Eligibility for Advanced JAVA Programming Course

- **Any Technical Graduates or Undergraduate (BSc, BCS, BCA, BE, B Tech, MSc, MCS, MCA, M Tech)**

Course overview For Hadoop/Big Data Course

Hadoop/Big Data Course

Hadoop Basic Concepts

- What is Hadoop?
- The Hadoop Distributed File System
- How Hadoop Map Reduce Works
- Anatomy of a Hadoop Cluster

Setting up a Hadoop Cluster

- Make a fully distributed Hadoop Cluster
- Network Topology
- Cluster Specification and installation
- Hadoop Configuration

Hadoop Daemons

- Master Daemons
- Name node
- Job Tracker
- Secondary name node
- Slave Daemons
- Data node
- Task tracker

HDFS (Hadoop Distributed File System)

- Blocks and Splits
- Input Splits
- HDFS Splits
- Methods of accessing HDFS
- JAVA Approach
- CLI Approach
- Cluster Architecture and Block Placement
- Data Replication
- Hadoop Rack Awareness
- High data availability
- Data Integrity
- Programming Practices
- Developing Maps Reduce Programs in
- Local Mode
- Running without HDFS and Map reduce
- Pseudo-distributed mode
- Running all daemons in a single node
- Fully distributed mode
- Running daemons in dedicated nodes

Syllabus for HADOOP/BIGDATA (HADOOP 2)

Writing a Map Reduce Program

- Examining a sample mapreduce program with several examples
- Basic API Concepts
- The Driver Code
- The Mapper
- The Reducer
- The configure and close methods
- Sequence Files
- Record Reader
- Record writer
- Role of Reporter
- Output Collector
- Processing XML Files
- Counters
- Directly Accessing HDFS
- Tool runner
- Using the Distributed Cache

Common Map Reduce Algorithms

- Sorting, Searching and Indexing
- Word Co-occurrence
- Identity Mapper
- Identity Reducer
- Exploring well-known problems using Map Reduce applications

Debugging Map Reduce Programs

- Testing with MR Unit
- Logging
- Other Debugging Strategies

Advanced Map reduce Programming

- A recap of the Map reduce Flow
- The Secondary Sort
- Customized Input formats and Output formats

Introduction to YARN

- What is YARN?
- Why YARN?
- Advantages of YARN
- YARN Daemons
- Resource Manager
- Node Manager
- Application Master
- Classic Mapreduce Vs YARN
- Anatomy of a YARN application run
- Scheduling in YARN
- Fair Scheduler
- Capacity Scheduler
- YARN as a platform for multiple applications
- Supported YARN applications

Syllabus for HADOOP/BIGDATA (HADOOP 2)

Hadoop Ecosystem

Overview of Spark

- What is Spark?
- Hadoop & Spark
- Features of Spark
- Spark Ecosystems
- Spark Streaming
- Spark SQL
- Spark MLib
- Spark Architecture
- Resilient Distributed Datasets
- How to install Spark
- How to run Spark
- How to interact with Spark
- Spark Web Console
- Shared Variables
- Spark Applications
- Word Count Application

Hive

- Hive Concepts
- Hive architecture
- Create database, access it from java client
- Buckets
- Partition
- Joins in hive
- Inner Joins
- Outer Joins
- Hive UDF

Impala

- Introducing Cloudera Impala
- Impala Benefits
- How Cloudera Impala works with CDH
- Primary Impala Features
- Impala Concepts and Architecture
- Components of the Impala Server
- The Impala Daemon
- The Impala Statestore
- The Impala Catalogue Service
- Overview of the Impala SQL Dialect
- How Impala fits into the Hadoop Ecosystem
- How Impala works with Hive
- Overview of Impala Metadata and Metastore
- How Impala uses HDFS

PIG

- Pig basics
- PIG Vs Map reduce and SQL
- PIG Vs Hive
- Write sample Pig Latin Scripts
- Modes of running PIG
- Running in Grunt shell
- Pig UDFs
- Pig Macros

Flume

- Flume Concepts
- Create a sample application to capture logs from Apache using Flume

Syllabus for HADOOP/BIGDATA (HADOOP 2)

Sqoop

- Getting Sqoop
- A sample import
- Database Imports
- Controlling the Import
- Imports and Consistency
- Direct-mode Imports
- Performing an export

CDH Enhancements

- Name Node High-Availability
- Name node federation
- Fencing

Project

Project Analysis using HADOOP will provide an efficient way of analyzing data using HDFS and Map Reduce fundamentals. The data can be utilized in several analysis. HADOOP allows user to process large amount of such data. There could be several complex use cases which can be easily answered by PIG and HIVE and other eco-systems.