

## Informatica Big Data for Developers Training

---

### Course Summary

#### Description

This course meets the needs of your organization. Learn to leverage Informatica Big Data Management for the optimization of data warehousing through the offloading of data processing to Hadoop. Also discover options for the enhancement of a data warehouse through accessing NoSQL databases and complex file parsing.

#### Objectives

After successfully completing this course, students should be able to:

- Define “Big Data”
- Identify and prioritize the offloading resource intensive Data Warehouse processes to Hadoop
- Migrate PowerCenter mappings to Big Data Management and ingest data into Hadoop
- Migrate and ingest data into Hadoop using SQOOP and SQL to Mapping
- Describe the Informatica on Hadoop architecture
- Transform data on Hadoop using Informatica polyglot computing
- Differentiate the capabilities of the Informatica engines on Hadoop including Hive MR/Tez, Blaze, and Spark engines
- Leverage the Informatica Smart Executor
- Utilize Informatica and Hadoop monitoring and troubleshooting
- Parse and transform complex data such as JSON, AVRO, and Parquet
- Describe how Informatica parses, reads, and writes NoSQL data collections

#### Audience

This course is designed for Developers and Architects

#### Topics

- Big Data Integration Course Introduction
- Big Data Basics
- Data Warehouse Offloading
- Ingestion and Offload
- Big Data Management Architecture
- Informatica Polyglot Computing in Hadoop
- Mappings, Monitoring, and Troubleshooting
- Hadoop Data Integration Challenges and Performance Tuning
- Data Quality on Hadoop
- Complex File Parsing
- Accessing NoSQL Databases

#### Prerequisites

Prerequisite is Informatica Developer Tool for Big Data Developers course.

#### Duration

Three Days

## Informatica Big Data for Developers Training

---

### Course Outline

#### I. *Big Data Integration Course*

##### *Introduction*

- A. Course Agenda
- B. Accessing the lab environment
- C. Related Courses

#### II. *Big Data Basics*

- A. What is Big Data?
- B. Hadoop concepts
- C. Hadoop Architecture Components
- D. The Hadoop Distributed File System (HDFS)
- E. Purposes of a Name Node & Secondary Name Node
- F. MapReduce
- G. “Yet Another Resource Manager” (YARN) (MapReduce Version 2)

#### III. *Data Warehouse Offloading*

- A. Challenges with traditional Data Warehousing
- B. The requirements of optimal Data Warehouse
- C. The Data Warehouse Offloading Process

#### IV. *Ingestion and Offload*

- A. PowerCenter Reuse Reports
- B. Importin PowerCenter Mappings to Developer
- C. SQOOP
- D. SQL to Mapping capability
- E. Partitioning and parallelism

#### V. *Big Data Management Architecture*

- A. The Big Data world
- B. Build once, deploy anywhere
- C. The Informatica abstraction layer
- D. Polyglot computing
- E. The Smart Executor
- F. Open source and innovation
- G. Connection architecture
- H. Conections to third Party applications

#### VI. *Informatica Polyglot Computing in Hadoop*

- A. Hive MR/Tez
- B. Blaze

- C. Spark
- D. Native
- E. The Smart Executor

#### VII. *Mappings, Monitoring, and Troubleshooting*

- A. Configuring and running a mapping in Native and Hadoop environments
- B. Execution Plans
- C. Monitor mappings
- D. Troubleshoot mappings
- E. Viewing mapping results

#### VIII. *Hadoop Data Integration Challenges and Performance Tuning*

- A. Describe challenges with executing mappings in Hadoop
- B. Big Data Management Performance Tuning
- C. Hive Environment Optimization
- D. Tips

#### IX. *Data Quality on Hadoop*

- A. The Data Quality process
- B. Discover insights into your data
- C. Collaborate and Create Data Improvement Assets
- D. Modify, Manage, and Monitor Data Quality
- E. Self Service Data Quality
- F. Executing Data Quality mappings on Hadoop

#### X. *Complex File Parsing*

- A. The Complex file reader
- B. The Data Processor transformation
- C. The Complex file writer
- D. Performance Considerations: Partitioning
- E. Parsing and processing Avro, Parquet, JSON, and XML file
- F. Data Processor Transformation Considerations

#### XI. *Accessing NoSQL Databases*

- A. CAP Theorem
- B. HBase
- C. MongoDB
- D. Cassandra