Cloudera Developer Training for Apache Hadoop

Varighet: 5.00 Days   Kurskode: GK3902

Beskrivelse:

HBase is an open-source, non-relational, distributed database that provides a fault-tolerant, scalable way to store massive quantities of data. In this course, Hadoop developers and administrators will gain the skills needed to install and maintain HBase and develop client code.

Målgruppe:

Developers familiar with Apache Hadoop

Agenda:

- In this course you will learn:

- HBase architecture

-

- Use the HBase shell to directly manipulate HBase tables

-

- Design optimal HBase schemas for efficient data storage and recovery

-

- Connect to HBase using the Java API

-

- Bulk-load data into HBase using MapReduce

-

- Administer a HBase cluster

-

- Resolve performance bottlenecks

Forkunnskaper:

- Familiarity with Hadoop's architecture and APIs
- Experience writing basic applications
- Prior programming experience, preferably Java
- Experience with databases and data modeling is helpful, but it is not required

Test og sertifisering

This course is part of the following programs or tracks:

- CCSHB: Cloudera Certified Specialist in Apache HBase (CDH4)

Innhold:

Data Model

- Tables, Row Keys, and Column Families

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

HBase Shell

- Creating and Manipulating Data Using the Command-Line Shell

Cluster Architecture

- HMaster, RegionServers, and Zookeeper

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase

Schema Design

- Creating Column Families

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

HBase API

- Connecting to HBase Using the Java API

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

HBase Administration

- Monitoring HBase Processes

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations

- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

Storage Architecture

- Client Caching

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting

- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

MapReduce and Bulk Loads

- MapReduce Integration

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups

- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

Performance Tuning

- Preventing Network Bandwidth Bottlenecks

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters

Tools

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

HBase Configuration

- Standalone and Distributed Run Modes

- Choosing Column Attributes
- Version and HBase Operations
- Compactions in HBase
- Crash Recovery
- Data Storage and Bloom Filters
- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

- Modifying Rows
- Designing for Locality and Access Patterns
- Detecting and Preventing Hot Spots
- Administrative Actions Using the Java API
- Accessing Data Using the Java API
- Bulk-Load into HBase
- Required Zookeeper Configurations
- Required Configuration Settings
- Performing HBase Backups
- Planning for HBase Capacity
- Java Garbage Collection and HBase Operations
- Tuning for Client Operations
- Logging Locations and Troubleshooting Tools

Labs

- Use the HBase Shell
- Flushes and Compactions
- View and Read HFiles
- Detect Hot Spots
- Write an HBase Program
- HBase Input MapReduce
- Bulk Import into HBase
- Install Distributed HBase
- Pre-Split Regions

Ytterligere informasjon:

For mer informasjon eller kursbooking, vennligst ring oss 22 95 66 00

info@globalknowledge.no

www.globalknowledge.no

Grenseveien 97, 0663 Oslo, PO Box 6256 Etterstad, 0606 Oslo, Norway