CCD-410 Study Guide

This exam focuses on engineering data solutions in MapReduce and understanding the Hadoop ecosystem (including Hive, Pig, Sqoop, Oozie, Crunch, and Flume). Candidates who successfully pass CCD–410 are awarded the Cloudera Certified Hadoop Developer (CCDH) credential.

Recommended Cloudera Training Course

Cloudera Developer Training for Apache Hadoop

Practice Test

CCD–410 Practice Test Subscription

Exam Sections

Each candidate receives 50 - 55 live questions. Questions are delivered dynamically and based on difficulty ratings so that each candidate receives an exam at a consistent level. Each test also includes at least five unscored, experimental (beta) questions.

Infrastructure: Hadoop components that are outside the concerns of a particular MapReduce job that a developer needs to master (25%)
Data Management: Developing, implementing, and executing commands to properly manage the full data lifecycle of a Hadoop job (30%)
Job Mechanics: The processes and commands for job control and execution with an emphasis on the process rather than the data (25%)
Querying: Extracting information from data (20%)

1. Infrastructure Objectives

- Recognize and identify Apache Hadoop daemons and how they function both in data storage and processing.
- Understand how Apache Hadoop exploits data locality.
- Identify the role and use of both MapReduce v1 (MRv1) and MapReduce v2 (MRv2 / YARN) daemons.
- Analyze the benefits and challenges of the HDFS architecture.
- Analyze how HDFS implements file sizes, block sizes, and block abstraction.
- Understand default replication values and storage requirements for replication.
- Determine how HDFS stores, reads, and writes files.
- Identify the role of Apache Hadoop Classes, Interfaces, and Methods.
- Understand how Hadoop Streaming might apply to a job workflow.

2. Data Management Objectives

- Import a database table into Hive using Sqoop.
- Create a table using Hive (during Sqoop import).
- Successfully use key and value types to write functional MapReduce jobs.
- Given a MapReduce job, determine the lifecycle of a Mapper and the lifecycle of a Reducer.
- Analyze and determine the relationship of input keys to output keys in terms of both type and number, the sorting of keys, and the sorting of values.
- Given sample input data, identify the number, type, and value of emitted keys and values from the Mappers as well as the emitted data from each Reducer and the number and contents of the output file(s).
- Understand implementation and limitations and strategies for joining datasets in MapReduce.
- Understand how partitioners and combiners function, and recognize appropriate use cases for each.
- Recognize the processes and role of the the sort and shuffle process.
- Understand common key and value types in the MapReduce framework and the interfaces they implement.
- Use key and value types to write functional MapReduce jobs.

3. Job Mechanics Objectives

- Construct proper job configuration parameters and the commands used in job submission.
- Analyze a MapReduce job and determine how input and output data paths are handled.
- Given a sample job, analyze and determine the correct InputFormat and OutputFormat to select based on job requirements.
- Analyze the order of operations in a MapReduce job.
- Understand the role of the RecordReader, and of sequence files and compression.
- Use the distributed cache to distribute data to MapReduce job tasks.
- Build and orchestrate a workflow with Oozie.

4. Querying Objectives

- Write a MapReduce job to implement a HiveQL statement.
- Write a MapReduce job to query data stored in HDFS.