



CCD-410^{Q&As}

Cloudera Certified Developer for Apache Hadoop (CCDH)

Pass Cloudera CCD-410 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.passapply.com/ccd-410.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Cloudera
Official Exam Center

- ⚙ **Instant Download** After Purchase
- ⚙ **100% Money Back** Guarantee
- ⚙ **365 Days** Free Update
- ⚙ **800,000+** Satisfied Customers





QUESTION 1

A client application creates an HDFS file named foo.txt with a replication factor of 3. Identify which best describes the file access rules in HDFS if the file has a single block that is stored on data nodes A, B and C?

- A. The file will be marked as corrupted if data node B fails during the creation of the file.
- B. Each data node locks the local file to prohibit concurrent readers and writers of the file.
- C. Each data node stores a copy of the file in the local file system with the same name as the HDFS file.
- D. The file can be accessed if at least one of the data nodes storing the file is available.

Correct Answer: D

HDFS keeps three copies of a block on three different datanodes to protect against true data corruption.

HDFS also tries to distribute these three replicas on more than one rack to protect against data availability issues. The fact that HDFS actively monitors any failed datanode(s) and upon failure detection immediately schedules re-replication of blocks (if needed) implies that three copies of data on three different nodes is sufficient to avoid corrupted files.

Note:

HDFS is designed to reliably store very large files across machines in a large cluster. It stores each file as a sequence of blocks; all blocks in a file except the last block are the same size. The blocks of a file are replicated for fault tolerance. The block size and replication factor are configurable per file. An application can specify the number of replicas of a file. The replication factor can be specified at file creation time and can be changed later. Files in HDFS are write-once and have strictly one writer at any time. The NameNode makes all decisions regarding replication of blocks. HDFS uses rack-aware replica placement policy. In default configuration there are total 3 copies of a datablock on HDFS, 2 copies are stored on datanodes on same rack and 3rd copy on a different rack.

Reference: 24 Interview Questions and Answers for Hadoop MapReduce developers , How the HDFS Blocks are replicated?

QUESTION 2

Given a directory of files with the following structure: line number, tab character, string:

Example: 1 abialkijfkaoasdfjksdlkjhqweroij 2 kadjfjhuwqounahagtnbvaswslmnbfgy 3 kjfteiomndscxeqalkzhtopedkfsikj



You want to send each line as one record to your Mapper. Which InputFormat should you use to complete the line:
`conf.setInputFormat (____.class) ; ?`

- A. SequenceFileAsTextInputFormat
- B. SequenceFileInputFormat
- C. KeyValueFileInputFormat
- D. BDBInputFormat

Correct Answer: C

<http://stackoverflow.com/questions/9721754/how-to-parse-customwritable-from-text-in-hadoop>

QUESTION 3

Can you use MapReduce to perform a relational join on two large tables sharing a key? Assume that the two tables are formatted as comma-separated files in HDFS.

- A. Yes.
- B. Yes, but only if one of the tables fits into memory
- C. Yes, so long as both tables fit into memory.
- D. No, MapReduce cannot perform relational operations.
- E. No, but it can be done with either Pig or Hive.

Correct Answer: A

Note:

*

Join Algorithms in MapReduce A) Reduce-side join B) Map-side join C) In-memory join / Striped Striped variant variant / Memcached variant

*

Which join to use? / In-memory join > map-side join > reduce-side join / Limitations of each? In-memory join: memory
Map-side join: sort order and partitioning

Reduce-side join: general purpose

QUESTION 4

Identify which best defines a SequenceFile?

- A. A SequenceFile contains a binary encoding of an arbitrary number of homogeneous Writable objects
- B. A SequenceFile contains a binary encoding of an arbitrary number of heterogeneous Writable objects



- C. A SequenceFile contains a binary encoding of an arbitrary number of WritableComparable objects, in sorted order.
- D. A SequenceFile contains a binary encoding of an arbitrary number key-value pairs. Each key must be the same type. Each value must be the same type.

Correct Answer: D

SequenceFile is a flat file consisting of binary key/value pairs.

There are 3 different SequenceFile formats:

Uncompressed key/value records.

Record compressed key/value records - only `\\values\\` are compressed here.

Block compressed key/value records - both keys and values are collected in `\\blocks\\` separately and compressed. The size of the `\\block\\` is configurable.

Reference: <http://wiki.apache.org/hadoop/SequenceFile>

QUESTION 5

What is the disadvantage of using multiple reducers with the default HashPartitioner and distributing your workload across you cluster?

- A. You will not be able to compress the intermediate data.
- B. You will longer be able to take advantage of a Combiner.
- C. By using multiple reducers with the default HashPartitioner, output files may not be in globally sorted order.
- D. There are no concerns with this approach. It is always advisable to use multiple reduces.

Correct Answer: C

Multiple reducers and total ordering

If your sort job runs with multiple reducers (either because `mapreduce.job.reduces` in `mapred-site.xml` has been set to a number larger than 1, or because you've used the `-r` option to specify the number of reducers on the command-line), then by default Hadoop will use the HashPartitioner to distribute records across the reducers. Use of the HashPartitioner means that you can't concatenate your output files to create a single sorted output file. To do this you'll need total ordering,

Reference: Sorting text files with MapReduce

[CCD-410 Practice Test](#)

[CCD-410 Study Guide](#)

[CCD-410 Braindumps](#)