



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

You have written a Mapper which invokes the following five calls to the `OutputCollector.collect` method:

```
output.collect (new Text ("Apple"), new Text ("Red")) ;
```

```
output.collect (new Text ("Banana"), new Text ("Yellow")) ; output.collect (new Text ("Apple"), new Text  
("Yellow")) ; output.collect (new Text ("Cherry"), new Text ("Red")) ;
```

```
output.collect (new Text ("Apple"), new Text ("Green")) ;
```

How many times will the Reducer's `reduce` method be invoked?

- A. 6
- B. 3
- C. 1
- D. 0
- E. 5

Correct Answer: B

`reduce()` gets called once for each [key, (list of values)] pair. To explain, let's say you called:

```
out.collect(new Text("Car"),new Text("Subaru");
```

```
out.collect(new Text("Car"),new Text("Honda");
```

```
out.collect(new Text("Car"),new Text("Ford");
```

```
out.collect(new Text("Truck"),new Text("Dodge");
```

```
out.collect(new Text("Truck"),new Text("Chevy");
```

Then `reduce()` would be called twice with the pairs

```
reduce(Car, )
```

```
reduce(Truck, )
```

Reference: Mapper `output.collect()`?

QUESTION 2

In the reducer, the MapReduce API provides you with an iterator over Writable values. What does calling the `next ()` method return?

- A. It returns a reference to a different Writable object time.



- B. It returns a reference to a Writable object from an object pool.
- C. It returns a reference to the same Writable object each time, but populated with different data.
- D. It returns a reference to a Writable object. The API leaves unspecified whether this is a reused object or a new object.
- E. It returns a reference to the same Writable object if the next value is the same as the previous value, or a new Writable object otherwise.

Correct Answer: C

Calling `Iterator.next()` will always return the SAME EXACT instance of `IntWritable`, with the contents of that instance replaced with the next value.

Reference: manipulating iterator in mapreduce

QUESTION 3

On a cluster running MapReduce v1 (MRv1), a TaskTracker heartbeats into the JobTracker on your cluster, and alerts the JobTracker it has an open map task slot. What determines how the JobTracker assigns each map task to a TaskTracker?

- A. The amount of RAM installed on the TaskTracker node.
- B. The amount of free disk space on the TaskTracker node.
- C. The number and speed of CPU cores on the TaskTracker node.
- D. The average system load on the TaskTracker node over the past fifteen (15) minutes.
- E. The location of the InputSplit to be processed in relation to the location of the node.

Correct Answer: E

The TaskTrackers send out heartbeat messages to the JobTracker, usually every few minutes, to reassure the JobTracker that it is still alive. These message also inform the JobTracker of the number of available slots, so the JobTracker can stay up to date with where in the cluster work can be delegated. When the JobTracker tries to find somewhere to schedule a task within the MapReduce operations, it first looks for an empty slot on the same server that hosts the DataNode containing the data, and if not, it looks for an empty slot on a machine in the same rack.

Reference: 24 Interview Questions and Answers for Hadoop MapReduce developers, How JobTracker schedules a task?

QUESTION 4

MapReduce v2 (MRv2/YARN) is designed to address which two issues?

- A. Single point of failure in the NameNode.
- B. Resource pressure on the JobTracker.
- C. HDFS latency.



D. Ability to run frameworks other than MapReduce, such as MPI.

E. Reduce complexity of the MapReduce APIs.

F. Standardize on a single MapReduce API.

Correct Answer: BD

YARN (Yet Another Resource Negotiator), as an aspect of Hadoop, has two major kinds of benefits:

*

(D) The ability to use programming frameworks other than MapReduce. / MPI (Message Passing Interface) was mentioned as a paradigmatic example of a MapReduce alternative

*

Scalability, no matter what programming framework you use. Note:

*

The fundamental idea of MRv2 is to split up the two major functionalities of the JobTracker, resource management and job scheduling/monitoring, into separate daemons. The idea is to have a global ResourceManager (RM) and per-application ApplicationMaster (AM). An application is either a single job in the classical sense of Map-Reduce jobs or a DAG of jobs.

*

(B) The central goal of YARN is to clearly separate two things that are unfortunately smushed together in current Hadoop, specifically in (mainly) JobTracker:

/ Monitoring the status of the cluster with respect to which nodes have which resources available. Under YARN, this will be global. / Managing the parallelization execution of any specific job. Under YARN, this will be done separately for each job. The current Hadoop MapReduce system is fairly scalable -- Yahoo runs 5000 Hadoop jobs, truly concurrently, on a single cluster, for a total 1.5 2 millions jobs/cluster/month. Still, YARN will remove scalability bottlenecks

Reference: Apache Hadoop YARN Concepts and Applications

QUESTION 5

Analyze each scenario below and identify which best describes the behavior of the default partitioner?

A. The default partitioner assigns key-values pairs to reducers based on an internal random number generator.

B. The default partitioner implements a round-robin strategy, shuffling the key-value pairs to each reducer in turn. This ensures an even partition of the key space.

C. The default partitioner computes the hash of the key. Hash values between specific ranges are associated with different buckets, and each bucket is assigned to a specific reducer.

D. The default partitioner computes the hash of the key and divides that value modulo the number of reducers. The result determines the reducer assigned to process the key-value pair.



E. The default partitioner computes the hash of the value and takes the mod of that value with the number of reducers. The result determines the reducer assigned to process the key-value pair.

Correct Answer: D

The default partitioner computes a hash value for the key and assigns the partition based on this result.

The default Partitioner implementation is called HashPartitioner. It uses the hashCode() method of the key objects modulo the number of partitions total to determine which partition to send a given (key, value) pair to.

In Hadoop, the default partitioner is HashPartitioner, which hashes a record's key to determine which partition (and thus which reducer) the record belongs in. The number of partitions is then equal to the number of reduce tasks for the job.

Reference: Getting Started With (Customized) Partitioning

[CCD-410 VCE Dumps](#)

[CCD-410 Study Guide](#)

[CCD-410 Exam Questions](#)