

APACHE-HADOOP-DEVELOPER^{Q&As}

Hadoop 2.0 Certification exam for Pig and Hive Developer





Pass Hortonworks APACHE-HADOOP-DEVELOPER Exam with 100% Guarantee

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

<https://www.leads4pass.com/apache-hadoop-developer.html>

100% Passing Guarantee
100% Money Back Assurance

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

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



QUESTION 1

You write MapReduce job to process 100 files in HDFS. Your MapReduce algorithm uses TextInputFormat: the mapper applies a regular expression over input values and emits key- values pairs with the key consisting of the matching text, and the value containing the filename and byte offset. Determine the difference between setting the number of reduces to one and settings the number of reducers to zero.

- A. There is no difference in output between the two settings.
- B. With zero reducers, no reducer runs and the job throws an exception. With one reducer, instances of matching patterns are stored in a single file on HDFS.
- C. With zero reducers, all instances of matching patterns are gathered together in one file on HDFS. With one reducer, instances of matching patterns are stored in multiple files on HDFS.
- D. With zero reducers, instances of matching patterns are stored in multiple files on HDFS. With one reducer, all instances of matching patterns are gathered together in one file on HDFS.

Correct Answer: D

Explanation: * It is legal to set the number of reduce-tasks to zero if no reduction is desired.

In this case the outputs of the map-tasks go directly to the FileSystem, into the output path set by `setOutputPath(Path)`. The framework does not sort the map-outputs before writing them out to the FileSystem.

* Often, you may want to process input data using a map function only. To do this, simply set `mapreduce.job.reduces` to zero. The MapReduce framework will not create any reducer tasks. Rather, the outputs of the mapper tasks will be the final output of the job.

Note:

Reduce

In this phase the `reduce(WritableComparable, Iterator, OutputCollector, Reporter)` method is called for each pair in the grouped inputs.

The output of the reduce task is typically written to the FileSystem via `OutputCollector.collect(WritableComparable, Writable)`.

Applications can use the Reporter to report progress, set application-level status messages and update Counters, or just indicate that they are alive.

The output of the Reducer is not sorted.

QUESTION 2

MapReduce v2 (MRv2/YARN) splits which major functions of the JobTracker into separate daemons? Select two.

- A. Health states checks (heartbeats)

- B. Resource management
- C. Job scheduling/monitoring
- D. Job coordination between the ResourceManager and NodeManager
- E. Launching tasks
- F. Managing file system metadata
- G. MapReduce metric reporting H. Managing tasks

Correct Answer: BC

Explanation: 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.

Note:

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.

Reference: Apache Hadoop YARN ?Conceptsand; Applications

QUESTION 3

Which process describes the lifecycle of a Mapper?

- A. The JobTracker calls the TaskTracker's configure () method, then its map () method and finally its close () method.
- B. The TaskTracker spawns a new Mapper to process all records in a single input split.
- C. The TaskTracker spawns a new Mapper to process each key-value pair.
- D. The JobTracker spawns a new Mapper to process all records in a single file.

Correct Answer: B

Explanation: For each map instance that runs, the TaskTracker creates a new instance of your mapper.

Note:

*

The Mapper is responsible for processing Key/Value pairs obtained from the InputFormat. The mapper may perform a number of Extraction and Transformation functions on the Key/Value pair before ultimately outputting none, one or many Key/Value pairs of the same, or different Key/Value type.

*

With the new Hadoop API, mappers extend the `org.apache.hadoop.mapreduce.Mapper` class. This class defines an `Identity` map function by default - every input Key/Value pair obtained from the InputFormat is written out.

Examining the `run()` method, we can see the lifecycle of the mapper:

```
/**
 *
 * Expert users can override this method for more complete control over the
 *
 * execution of the Mapper.
 *
 * @param context
 *
 * @throws IOException
 */
public void run(Context context) throws IOException, InterruptedException { setup(context);
while (context.nextKeyValue()) {
map(context.getCurrentKey(), context.getCurrentValue(), context); }
cleanup(context);
}
setup(Context) - Perform any setup for the mapper. The default implementation is a no-op method.
map(Key, Value, Context) - Perform a map operation in the given Key / Value pair. The default implementation calls Context.write(Key, Value)
cleanup(Context) - Perform any cleanup for the mapper. The default implementation is a no-op method.
```

Reference: Hadoop/MapReduce/Mapper

QUESTION 4

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

- A. You will not be able to compress the intermediate data.
- B. You will no 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 reducers.

Correct Answer: C

Explanation: 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](#)

QUESTION 5

Consider the following two relations, A and B.

```
A = LOAD 'data1' AS (a1:int,a2:chararray);
DUMP A;
(1,apple)
(3,orange)
(4,peach)
(2,cherry)

B = LOAD 'data2' AS (b1:chararray,b2:int);
DUMP B;
(Jim,2)
(Brian,4)
(Kim,0)
(Terry,3)
(Chris,2)
```

A Pig JOIN statement that combined relations A by its first field and B by its second field would produce what output?

- A. 2 Jim Chris 2 3 Terry 3 4 Brian 4
- B. 2 cherry 2 cherry 3 orange 4 peach
- C. 2 cherry Jim, Chris 3 orange Terry
4 peach Brian
- D. 2 cherry Jim 2 2 cherry Chris 2 3 orange Terry 3 4 peach Brian 4

Correct Answer: D

[APACHE-HADOOP-
DEVELOPER PDF Dumps](#)

[APACHE-HADOOP-
DEVELOPER Practice Test](#)

[APACHE-HADOOP-
DEVELOPER Exam
Questions](#)