**Answer1:**

A RecordReader uses the data within the boundaries created by the input split to generate key/value pairs. In the context of file-based input, the “start” is the byte position in the file where the RecordReader should start generating key/value pairs. The “end” is where it should stop reading records.

**Answer2:**

With reducer count 0, the outputs of the map-tasks will 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.

**Answer3:**

If the join is performed by mapper then it is called as Map-Side join.If the join is performed by reducer then it is called as Reduce-Side join.

**Answer4:**

Conf.setMapperclass sets the mapper class and all the stuff related to map job such as reading a data and generating a key-value pair out of the mapper.

**Answer5:**

Counter provides a way to measure the progress or the number of operations that occur within MapReduce programs. Map-Reduce provide built in counters to measure basic I/O operations, such as FILE\_BYTES\_READ/WRITTEN and Map/Combine/Reduce input/output records.

**Answer6:**

1. Processing of image files.
2. Complex nested condition
3. Processing of flexible schemas
4. Poorly designed XML
5. Files which are not properly delimited like log files

**Answer7:**

1. When hadoop developers need definite driver program control then they should make use of Hadoop MapReduce instead of Pig and Hive.
2. Whenever the job requires implementing a custom partitioner, hadoop developers can choose MapReduce over Pig and Hive.
3. If there already exists pre-defined library of Java Mappers or Reducers for a job then it is a wise decision to use Hadoop MapReduce instead of Pig and Hive.
4. If the hadoop developers require good amount of testability when combining lots of large data sets then they should use MapReduce instead of Pig and Hive.
5. If the application demands legacy code requirements that command physical structure then Hadoop MapReduce is a better option.
6. If the job requires optimization at a particular stage of processing by making the best use of tricks like in-mapper combining then Hadoop MapReduce can prove to be a better coding approach over Pig and Hive.
7. If the job has some tricky usage of distributed cache (replicated join), cross products, groupings or joins then Hadoop MapReduce is a better programming approach over Pig and Hive

**Answer8:**

The combiner function is used as an optimization for the MapReduce job. The combiner function runs on the output of the map phase and is used as a filtering or an aggregating step to lessen the number of intermediate keys that are being passed to the reducer. In most of the cases the reducer class is set to be the combiner class. The difference lies in the output from these classes. The output of the combiner class is the intermediate data that is passed to the reducer whereas the output of the reducer is passed to the output file on disk.