1. **What is MapReduce?**

MapReduce is a processing technique and a program model for distributed computing based on java.

MapReduce is a framework using which we can write applications to process huge amounts of data, in parallel, on large clusters of commodity hardware in a reliable manner.

1. **What are the advantages of MapReduce?**

The major advantage of MapReduce is that it is easy to scale data processing over multiple computing nodes.

1. **Explain Map reduce architecture?**

Can refer diagram in PPT

1. **What is Job Tracker and Task tracker? How does they work? (I)**
2. Client application submits job to job tracker.
3. Job Tracker talks to the Name node to determine the location of the data
4. Job Tracker finds the task tracker nodes with available slots at the data or at least near the data.
5. Then Job tracker submits the work to chosen task tracker nodes.
6. Then task tracker nodes are monitored. If do not submit heart beat signal often enough. Then the job tracker decides the particular task tracker is failed and then the work is scheduled on a different Task tracker
7. Task tracker will notify the job tracker when a task fails.
8. The job tracker will decide what to do and it may take actions like
   1. Task may be resubmitted else where
   2. It may mark specific data or record as something to avoid.
   3. It may black list the task tracker as unreliable
9. **What happens if Job tracker fails?**

Job tracker is a SPOF (Single point of failure) if it goes down all running jobs are halted

1. **Explain Map Reduce algorithm?**

The concept of this programming model is to send the code where data resides,

Map reduce program executes in 3 stages.

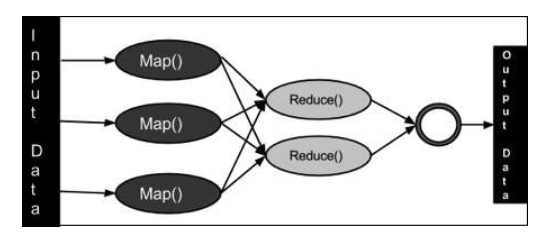
**Map stage:** Map takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs)

**Shuffle/Reduce:** This stage is combination of **shuffle** and **reduce** stage**.** Reduce task, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples.

The reduce task is always performed after the map job.

1. **How Map reduce job works?**

* During a MapReduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster.
* Then MapReduce framework is responsible for monitoring executed map reduce task, copying the data around the cluster between the nodes.
* Mapreduce helps to reduce network traffic Because most of the processing takes place in the node where data is stored.
* Once execution of the task is completed output data is collected and reduced to appropriate result and send it back to Hadoop server.

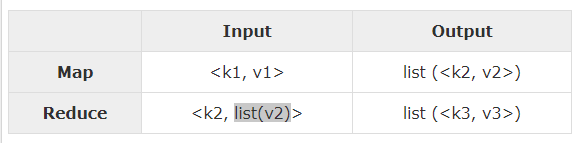


**Inputs and outputs in java perspective:**

Map reduce framework views the input to the job as a set of <key, value> pairs and produces a set of <key, value> pairs as the output of the job, it can also produce out of different types.

The key and the value classes should be in serialized manner by the framework and hence, need to implement the Writable interface.

Additionally, the key classes have to implement the Writable-Comparable interface to support sorting by the framework.



1. When work is completed Job tracker updates its status
2. Client applications can also poll the job tracker for information.
3. **Terminology used in Mapreduce?**

**JobTracker** − Schedules jobs and tracks the assign jobs to Task tracker.

1. ***PayLoad*** − Applications implement the Map and the Reduce functions, and form the core of the job.

**Mapper** − Mapper maps the input key/value pairs to a set of intermediate key/value pair.

**NamedNode** − Node that manages the Hadoop Distributed File System (HDFS).

**DataNode** − Node where data is presented in advance before any processing takes place.

**MasterNode** − Node where JobTracker runs and which accepts job requests from clients.

**SlaveNode** − Node where Map and Reduce program runs.

**Task Tracker** − Tracks the task and reports status to JobTracker.

**Job** − A program is an execution of a Mapper and Reducer across a dataset.

**Task** − An execution of a Mapper or a Reducer on a slice of data.

**Task Attempt** − A particular instance of an attempt to execute a task on a SlaveNode.

Note: Refer https://www.tutorialspoint.com/hadoop/hadoop\_mapreduce.htm