

1. What are the main stages of MapReduce?
 - MapReduce phases of execution, namely, splitting, mapping, sorting and shuffling, and reducing.
2. What are the types that can be used to instantiate `Mapper` and `Reducer` templates?
 - The input to the mapper function is in the form of (key, value) pairs, and *Reducer* takes a set of an intermediate key-value pair produced by the mapper as the *input*
3. What does `job.setMapperClass` do?
 - Set the `Mapper` for the job.
4. What does `job.setCombinerClass` do?
 - Set the combiner class for the job.
5. What does `job.setReducerClass` do?
 - Reduces a set of intermediate values which share a key to a smaller set of values.
6. What does `job.setOutputKeyClass` do?
 - Set the key class for the job output data.
7. Why does one need to have `job.waitForCompletion` in the code?
 - its method call returns only when the job gets finished,
8. How does one create a jar container?
 - `jar cf jar-file input-file(s)`
9. Can one apply several Map transformations within one MapReduce job?
 - Yes, as we did in Lab class.