**1. What is RDD?**

RDD is a logical reference of a dataset which is partitioned across many server machines in the cluster. RDDs are Immutable and are self recovered in case of failure. An RDD is a read-only, partitioned collection of records. RDD is a fault-tolerant collection of elements that can be operated on in parallel.

**2. Define Partitions.**

A partition is a logical chunk of a large distributed data set. Spark manages data using partitions that helps parallelize distributed data processing with minimal network traffic for sending data between executors. Because for subsequent transformations on the RDD, there’s a fair amount of shuffling of data across the network. If similar keys or range of keys are stored in the same partition then the shuffling is minimized and the processing becomes substantially fast.

**3. What operations does RDD support?**

RDDs support two types of operations:

*Transformations*: which create a new dataset from an existing one &

*Actions:*  which return a value to the driver program after running a computation on the dataset.

**4. What do you understand by Transformations in Spark?**

Transformations create new RDD from existing RDD. Transformations are executed on demand. That means they are computed lazily.

**5. Define Actions.**

Actions return final results of RDD computations. Actions triggers execution using lineage graph to load the data into original RDD, carry out all intermediate transformations and return final results to Driver program or write it out to file system.