RDD -Immutable distribution of objects. It is split into partitions which may be computed on different nodes.

* RDDs can be created in two ways – by loading external dataset or by distributing the collection of objects in the driver program.
* RDDs once created offer two types of operations – Transformatons and actions.
* Transformation – creating a new RDD from the old one
* Action – Creating a result out of the RDD.
* Spark computes RDDs in a lazy fashion, i.e. it computes RDD only when it is first introduced in an action. This way of computation is easier in big data. For example, we defined an RDD and immediately filtered, there’s no use of loading it at first as it will waste the memory because it is immediately going to be filtered. In fact, spark while computing the “first()” function eads until the first time it finds the match and ignores rest of the data

Example -

lines = sc.textFile(“README.md”)

//”lines” is an RDD.

Transformation : val pythonLines = lines.filter(line => line.contains(“Python”))

Action : pythonLines.first()