**ASSIGNMENT 13.1**

1. **What is RDD:**

* Spark revolves around the concept of a resilient distributed dataset (RDD), which is a fault-tolerant collection of elements that can be operated on in parallel.
* RDD’s in Spark are immutable distributed collection of objects.
* Each RDD is split into multiple partitions, which may be computed on different nodes of the cluster.
* The RDDs can contain any type of Python, Java or Scala objects, including user defined classes.
* They are not actual data, but they are Objects, which contain information about data residing on the cluster.
* The RDDs try to solve these problems by enabling fault tolerant, distributed In-memory computations.
* In Spark, data is stored in partitions of the RDDs and store in worker nodes (data nodes) which are computed in parallel across all the nodes.
* RDDs load the data for us and are resilient, which means they can be recomputed.
* There are two ways to create RDDs: parallelizing an existing collection in your driver program, or referencing a dataset in an external storage system, such as a shared filesystem, HDFS, HBase, or any data source offering a Hadoop InputFormat

1. **Define partitions**

* One important parameter for RDDs is he number of partitions
* It is the number of parts the data in the RDD will be divided into
* Spark will run one task for each partition of the cluster. So more the number of partitions, more the parallelization achieved on data-processing
* Normally, Spark tries to set the number of partitions automatically based on your cluster. However, you can also set it manually by passing it as a second parameter to parallelize (e.g. sc.parallelize(data, 10)).

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

RDDs support two types of operations:

* Transformations: Transformations create a new dataset from an existing one. For example, map is a transformation that passes each dataset element through a function and returns a new RDD representing the results
* Actions: Actions return a value to the driver program after running a computation on the dataset. For example, reduce is an action that aggregates all the elements of the RDD using some function and returns the final result to the driver program

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

* Transformations are operations on RDD that create a new dataset from an existing one. For example, map is a transformation that passes each dataset element through a function and returns a new RDD representing the results
* All transformations in Spark are lazy, meaning that they do not compute their results right away. Instead, they just remember the transformations applied to some base dataset (e.g. a file). The transformations are only computed when an action requires a result to be returned to the driver program. This design enables Spark to run more efficiently.

1. **Define Actions**

Actions are transformations that return a value to the driver program after running a computation on the dataset. For example, reduce is an action that aggregates all the elements of the RDD using some function and returns the final result to the driver program