Session 16 : Introduction to Apache Spark

Assignment 1

**Problem Statement**

Given a list of numbers - List[Int] (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)

- find the sum of all numbers

- find the total elements in the list

- calculate the average of the numbers in the list

- find the sum of all the even numbers in the list

- find the total number of elements in the list divisible by both 5 and 3

**Solution:-**

Starting Spark shell

Command - spark-shell



Creating a list and Loading it in Spark.

Commands:-

scala> var intList:List[Int]= List[Int](1, 2, 3, 4, 5, 6, 7, 8, 9, 10);

scala> val numbersRDD = sc.parallelize(intList);

scala> numbersRDD.collect().foreach(println);





RDD numbersRDD is now loaded in Spark.

Solution to the problem statements could be found from data in numbersRDD

**Solution to problem statements:-**

1. find the sum of all numbers

**Command -** scala> numbersRDD.reduce((x,y) => x+y);



1. find the total elements in the list

**Command-** scala> numbersRDD.count();



1. calculate the average of the numbers in the list

**Command -** scala> numbersRDD.reduce((x,y) => x+y).toDouble / numbersRDD.count().toDouble;



1. find the sum of all the even numbers in the list

Step 1- Filter out even numbers from numbersRDD and store in evenNumbersRDD

**Command-** scala> val evenNumbersRDD=numbersRDD.filter(i => (i%2==0));

Step 2- Find sum of numbers in evenNumbersRDD

**Command-** scala> evenNumbersRDD.reduce((x,y) => x+y);



1. find the total number of elements in the list divisible by both 5 and 3
2. Number of elements divisible by both 5 and 3 is zero

**Commands:-**

scala> val divBy5And3RDD=numbersRDD.filter(i => ((i%5==0)&&(i%3==0)));

scala> divBy5And3RDD.count();



1. Number of elements divisible by both 5 or 3 is 5

**Commands:-**

scala> val divBy5Or3RDD=numbersRDD.filter(i => ((i%5==0)||(i%3==0)));

scala> divBy5Or3RDD.count();

scala> divBy5Or3RDD.collect();

