G. Rupa Manasa 23-12-2023

Day 18 Assessment

**Basic Spark Commands:**

To start the Spark shell: **$bin/spark-shell**

Read a file from the local system**:**

**scala> val data = sc.txtFile(“data.txt”)**

Create RDD through parallelizing:

**scala> val num = Array(1,2,3,4,5)**

**scala> val NewData = sc.parallellize(num)**

Count Items in RDD:

**scala> NewData.count()**

Read the first 3 Items from RDD:

**Data.take(3)**

Save output/processed data into the text file:

**scala> counts.saveAsTextFile(“output”)**

**Intermediate Spark Commands:**

Filter on RDD:

**scala> val DFData = data.filter(line => line.contains(“yes”))**

Chain Operation:

**scala> data.filter(line => line.contains(“DataFlair”)).count()**

Read the first item from RDD:

**Data.first()**

Count RDD Partitions:

**scala> data. partitions. length**

Cache a File:

**scala> data.cache()**

**Basic Functions of Spark:**

READ:

We can start by loading the files in our data set using the **spark.read.load** command.

CHANGE COLUMN NAMES:

Sometimes, we want to change the name of the columns in our Spark dataframes. We can change column names and perform other transformations on DataFrame columns using the **withColumn** method.

SELECT COLUMNS:

We can also select a subset of columns using the **select** keyword.

SORT or ORDERBY:

We can use the sort or orderBy methods to sort the rows of a DataFrame based on one or more columns.

CAST:

The cast function is used to convert the data type of a column in a DataFrame. It allows you to change the data type of a column to another compatible data type.

FILTER:

The filter operation is used to filter rows of a DataFrame based on a specified condition.

We can filter a dataframe using AND(&), OR(|) and NOT(~) conditions.

GROUPBY:

The groupBy operation is used to group rows of a DataFrame based on one or more columns.

It is typically followed by an aggregation operation to perform computations on the grouped data.

JOINS:

Joins are operations that combine two DataFrames based on a common column or expression.

Spark supports several types of joins, including inner join, left join, right join, and outer join.