pip install pyspark from pyspark.sql.types import StructType, StringType, IntegerType, DoubleType, LongType from pyspark.sql import SparkSession spark = SparkSession.builder.appName('new york exchange').getOrCreate() schema_nyse = StructType().add("exchange_name",StringType(),True).add("stock_id",StringType(),True).add("stock_dt",StringType(),True).add("open",DoubleType print(schema_nyse) df_with_schema = spark.read.format("csv").option("header","False").schema(schema_nyse).load("/content/sample_data/NYSE.csv") df_with_schema.printSchema() df_with_schema.show() df_with_schema.registerTempTable("nyse") df_StockVol = spark.sql("select stock_id, sum(volume) from nyse group by stock_id") df_StockVol.rdd.getNumPartitions() df_StockVol = df_StockVol.coalesce(1) df_StockVol.write.csv("/content/sample_data/spark-sql1") or df_StockVol.write.mode("overwrite").csv("/content/sample_data/spark-sql1") ---------- After changing the default settings to 100 sqlContext = SQLContext(sc) sqlContext.setConf("spark.sql.shuffle.partitions", "100") df_StockVol = spark.sql("select stock_id, sum(volume) from nyse group by stock_id") df_StockVol.rdd.getNumPartitions() 100 ---------- airlines_DF = spark.read.format("csv").option("header", "True").option("inferSchema", "True").load("/content/sample_data/airlines.csv") airlines_DF.registerTempTable("airlines") YrWiseRev = spark.sql("select year, round(sum(Avg_rev_per_seat*booked_seats)/1000000,2) as total_in_mill from airlines group by year order by total_in_mill desc") YrWisePsx = spark.sql("select year, sum(booked_seats) as total_psx from airlines group by year order by total_psx desc")