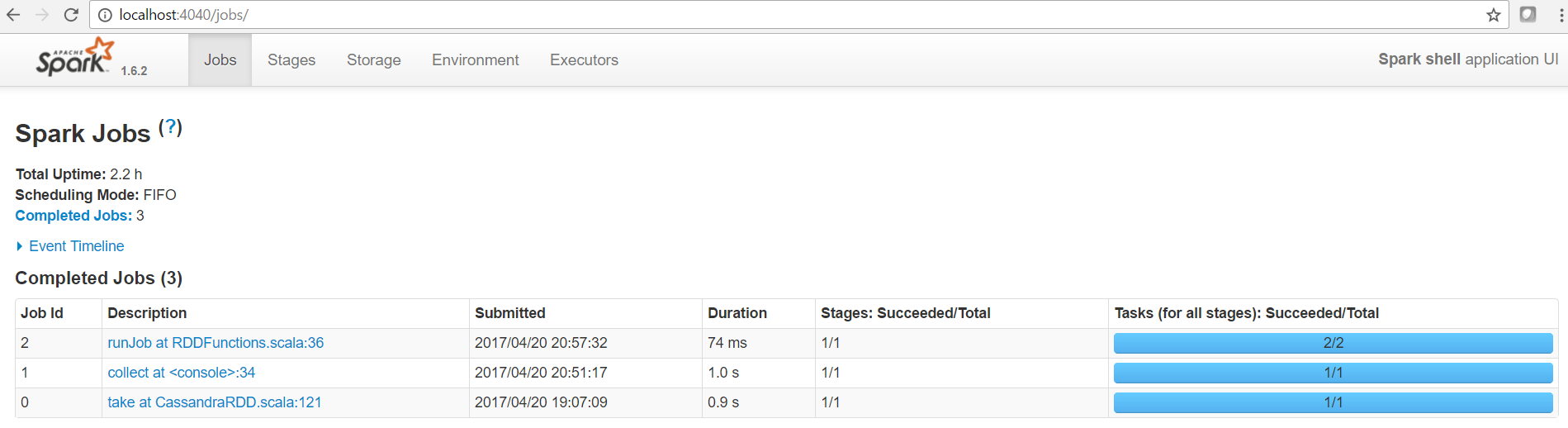
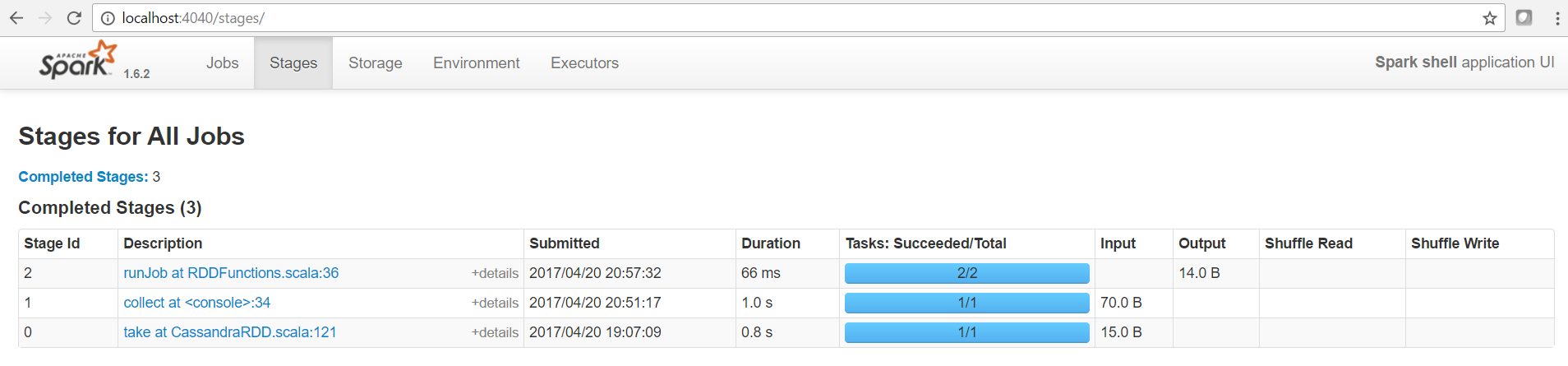
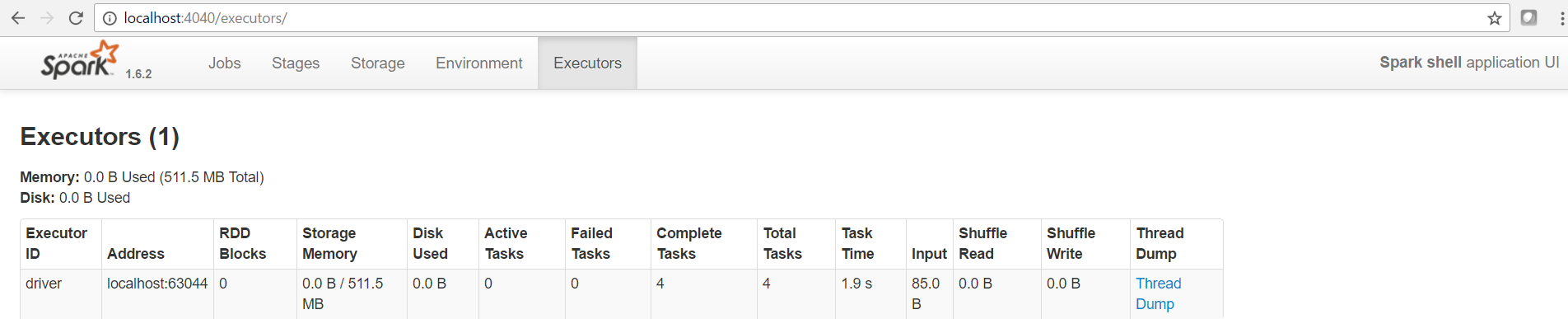
**Part – II – Loading Data from SPARK to CASSANDRA from an input file**

After Part – I is done the SPARK administration console looks like below –







**Saving a sample test data to Cassandra from Spark**

import com.datastax.spark.connector.\_, org.apache.spark.SparkContext, org.apache.spark.SparkContext.\_, org.apache.spark.SparkConf

import com.datastax.spark.connector.\_

import org.apache.spark.SparkContext

import org.apache.spark.SparkContext.\_

import org.apache.spark.SparkConf

scala> import org.apache.spark.\_

import org.apache.spark.\_

scala> import org.apache.spark.streaming.\_

import org.apache.spark.streaming.\_

scala> import org.apache.spark.streaming.StreamingContext.\_

import org.apache.spark.streaming.StreamingContext.\_

scala> val conf = new SparkConf().setMaster("local[2]").setAppName("StreamingTest").set("spark.driver.allowMultipleContexts","true")

conf: org.apache.spark.SparkConf = org.apache.spark.SparkConf@4ff60d4c

scala> val ssc = new StreamingContext(conf, Seconds(10))

ssc: org.apache.spark.streaming.StreamingContext = org.apache.spark.streaming.StreamingContext@4cbd4d9c

scala> val normalfill = ssc.fileStream("file:///D:/spark/normalfill.csv").map(line => line.split(",").map(\_.toInt));

<console>:40: error: missing parameter type

val normalfill = ssc.fileStream("file:///D:/spark/normalfill.csv").map(line => line.split(",").map(\_.toInt));

^

scala> val normalfill = ssc.textFileStream("file:///D:/spark/normalfill.csv").map(line => line.split(",").map(\_.toInt));

normalfill: org.apache.spark.streaming.dstream.DStream[Array[Int]] = org.apache.spark.streaming.dstream.MappedDStream@3262ed56

scala> val rowRDD = normalfill.map(line => (line(0), line(1), line(2)))

rowRDD: org.apache.spark.streaming.dstream.DStream[(Int, Int, Int)] = org.apache.spark.streaming.dstream.MappedDStream@4f468de8

scala> rowRDD.saveToCassandra("int\_ks1","int\_compound")

<console>:45: error: value saveToCassandra is not a member of org.apache.spark.streaming.dstream.DStream[(Int, Int, Int)]

rowRDD.saveToCassandra("int\_ks1","int\_compound")

^

scala> import com.datastax.spark.connector.streaming.\_

import com.datastax.spark.connector.streaming.\_

scala> rowRDD.saveToCassandra("int\_ks1","int\_compound")