Apache Spark

Spark is

1)open source

2)parallel

3)Scalable

4)Supports In-memory computation

4)Lazy evaluation

5)Fault Tolerant

6)Immutability

7)Partitioning

8)Persistence

9)Coarse Grained Operations

Spark is used for data analytics much like map-reduce but better than that

Unlike map-reduce which shuffles files on disk spark works in memory it’s fast as well

Features:

1)High-Speed

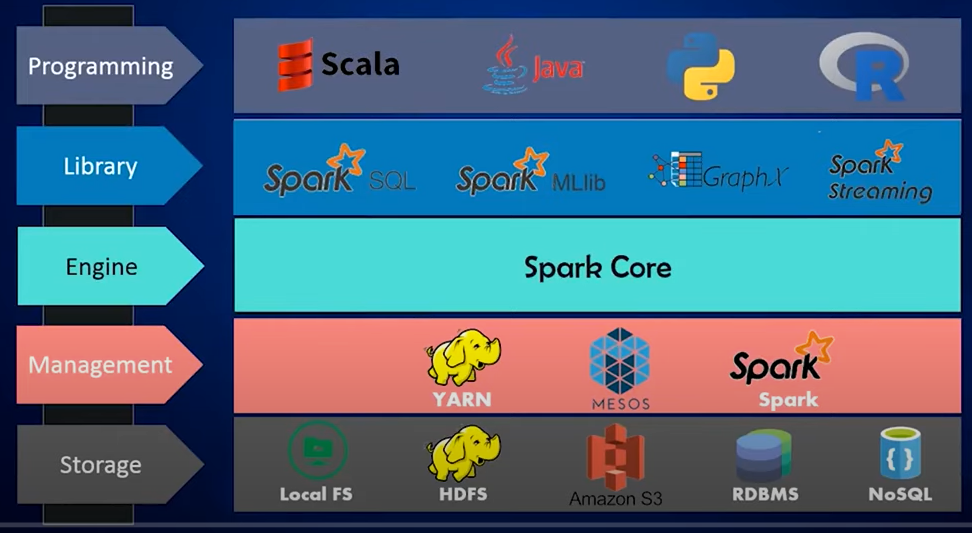
2)Powerful caching

3)Deployment is easy

4)Real-time computation

5)Supports multiple languages

Spark Eco-System:



RDD:

To reduce number of operation performed through hdfs can be achieved through in-memory data computation that’s where RDD’s come into picture RDD’s are fault tolerant fundamental data structure of spark which can handle both schema and schema less structures

Data read into spark🡪Same Data read into RDD🡪 Transformation of data🡪 New RDD is created depending on need🡪The data obtained is stored to persistent storage

Objects in RDD are immutable

RDD has logical partitions and the data is present in multiple partitions

so the multiple data copies are present and when one data is lost in some partition other partition data will help

RDD can be created by

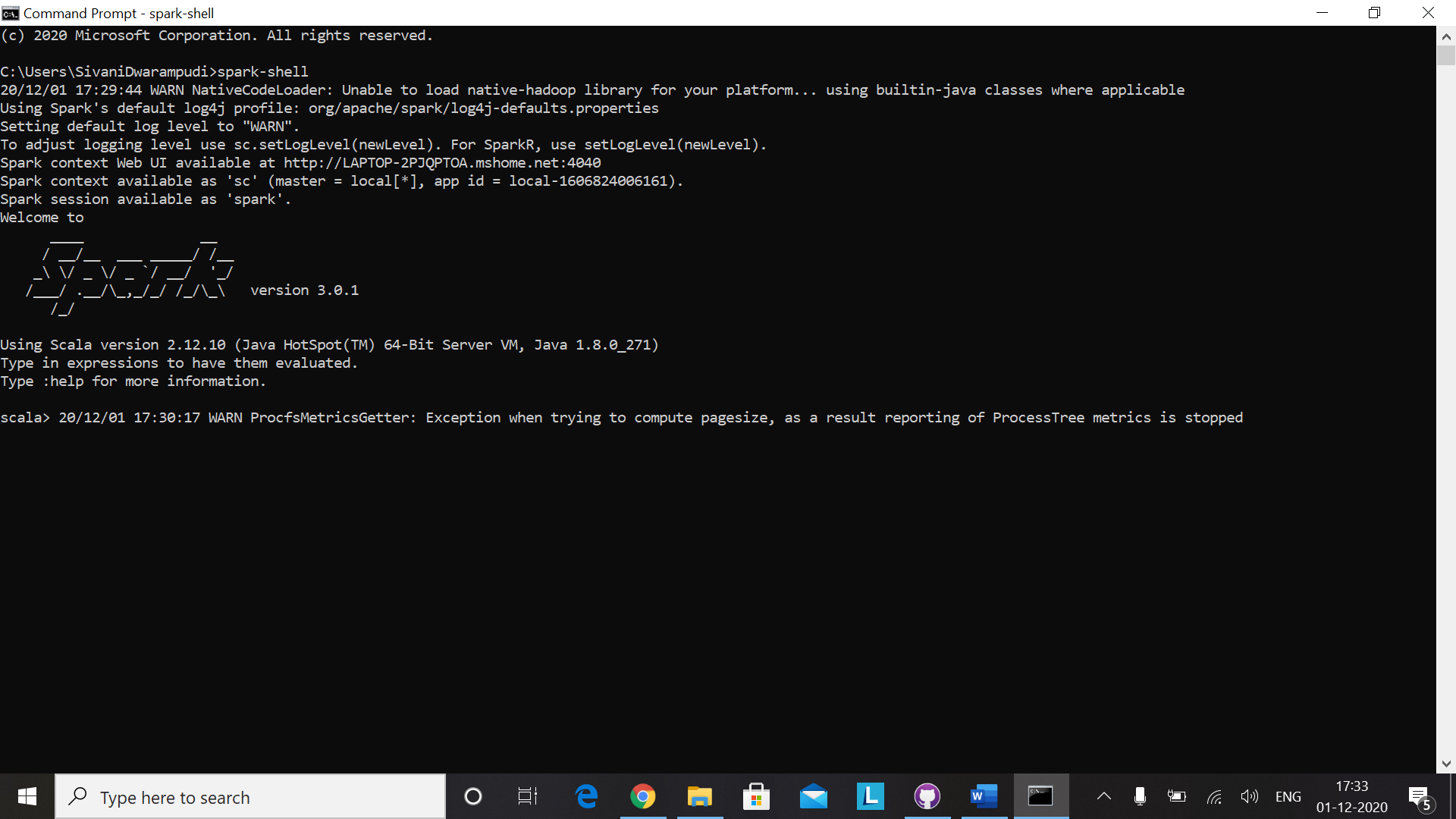
1)parallelized collections

2)From existing RDD’s

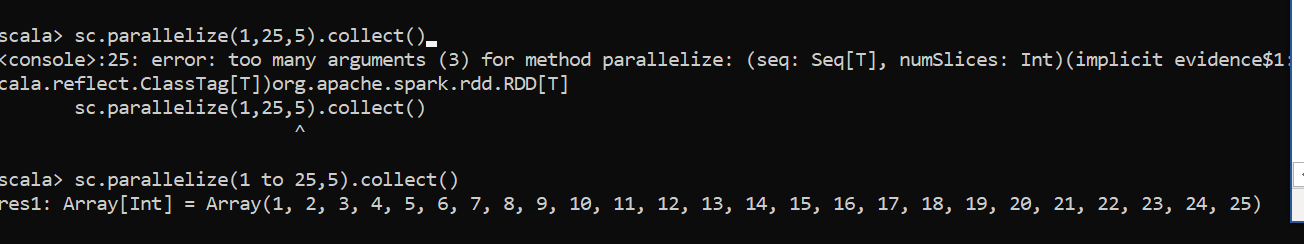
3)External Data

Steps for creating RDD using Parallelized collections:

Start Spark-Shell:

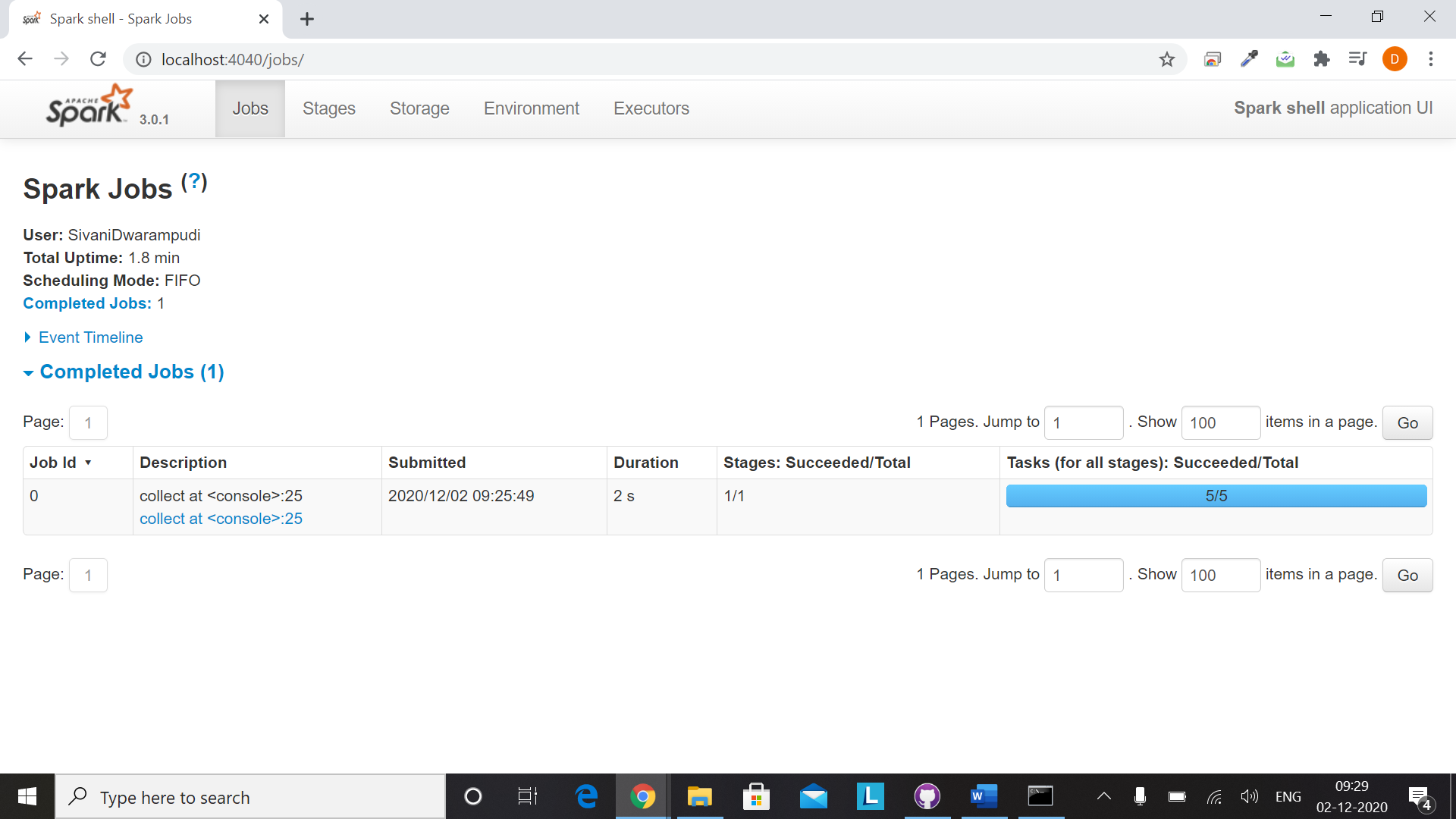


Create RDD:

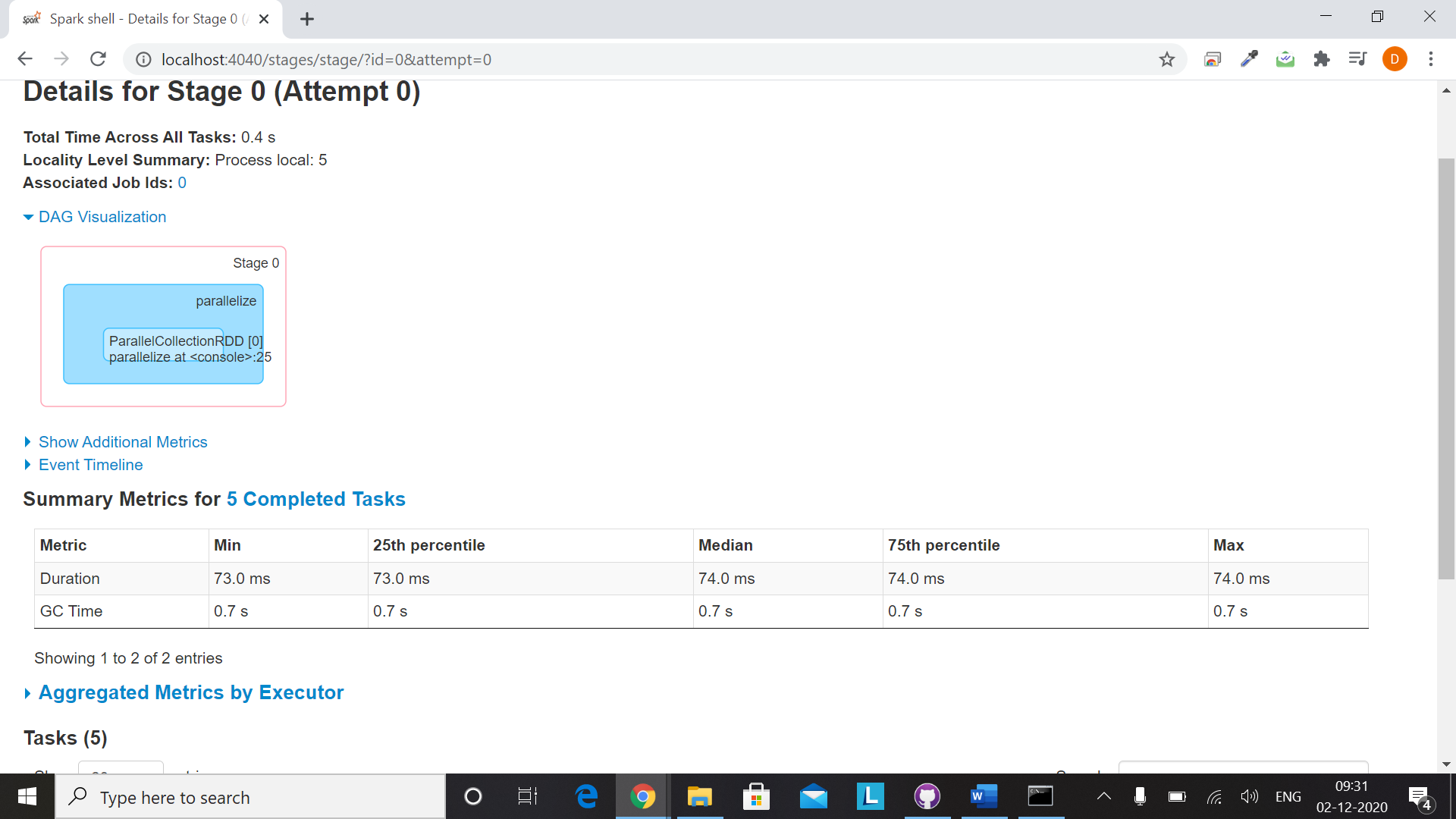


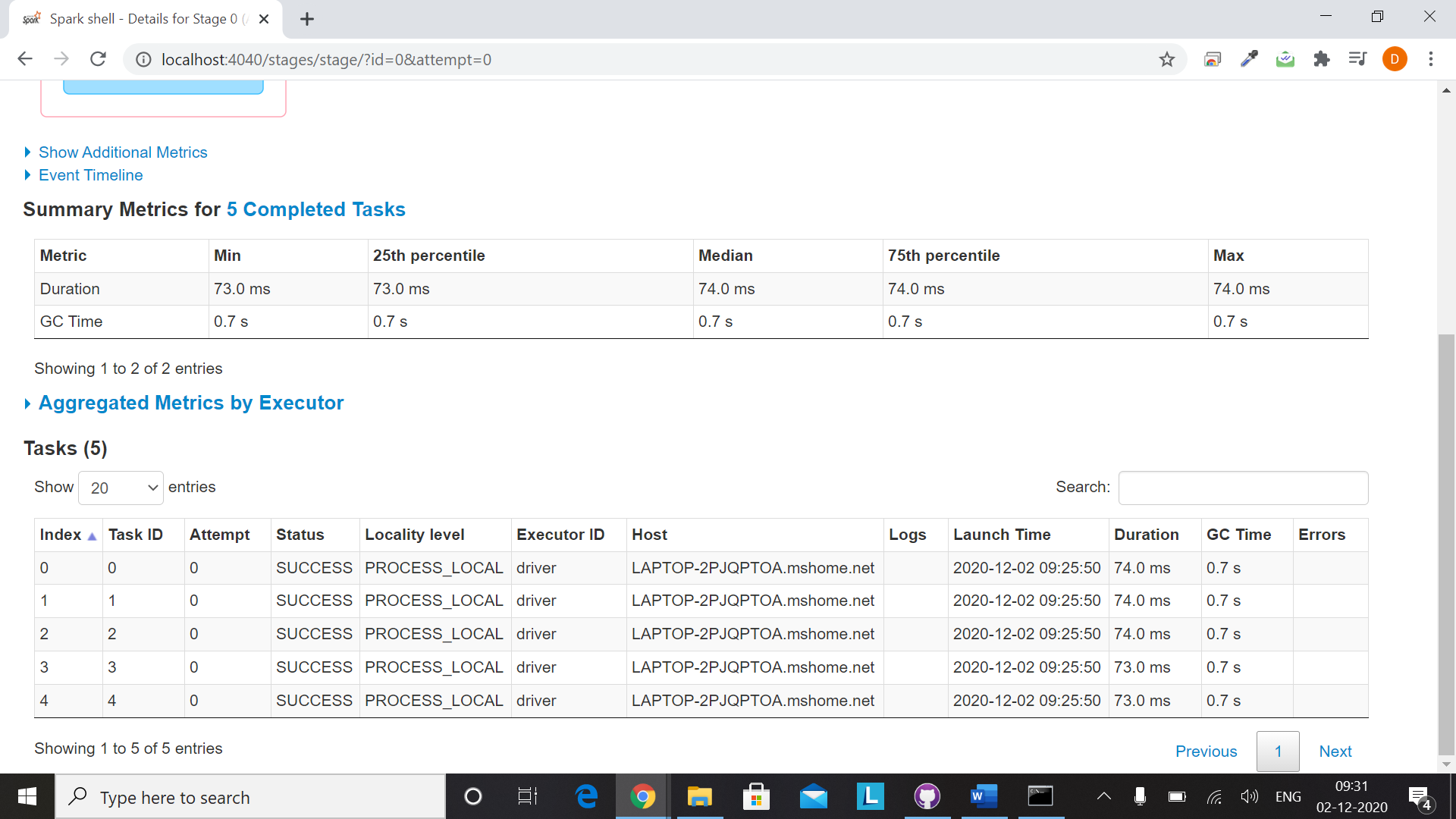
To view the content on web:

Localhost:4040 is the address to be typed on the web



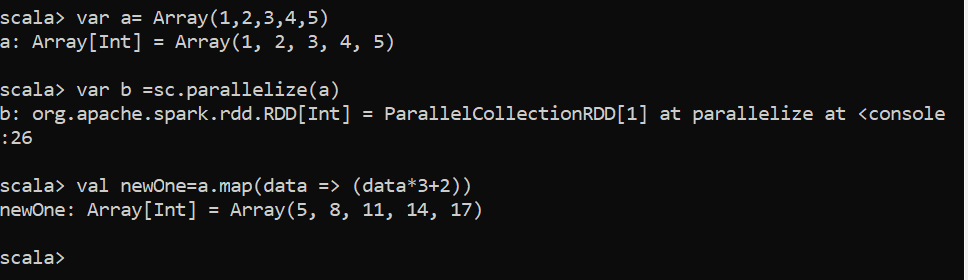
To get more information such as DAG click on the RDD that you’ve created



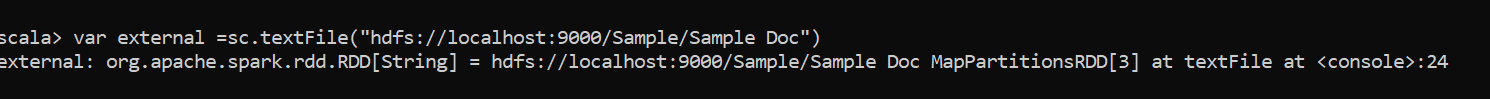


The status SUCCESS in the above screenshot mentions that all 5 tasks are executed successfully

RDD from existing RDD:



RDD from external file sources:

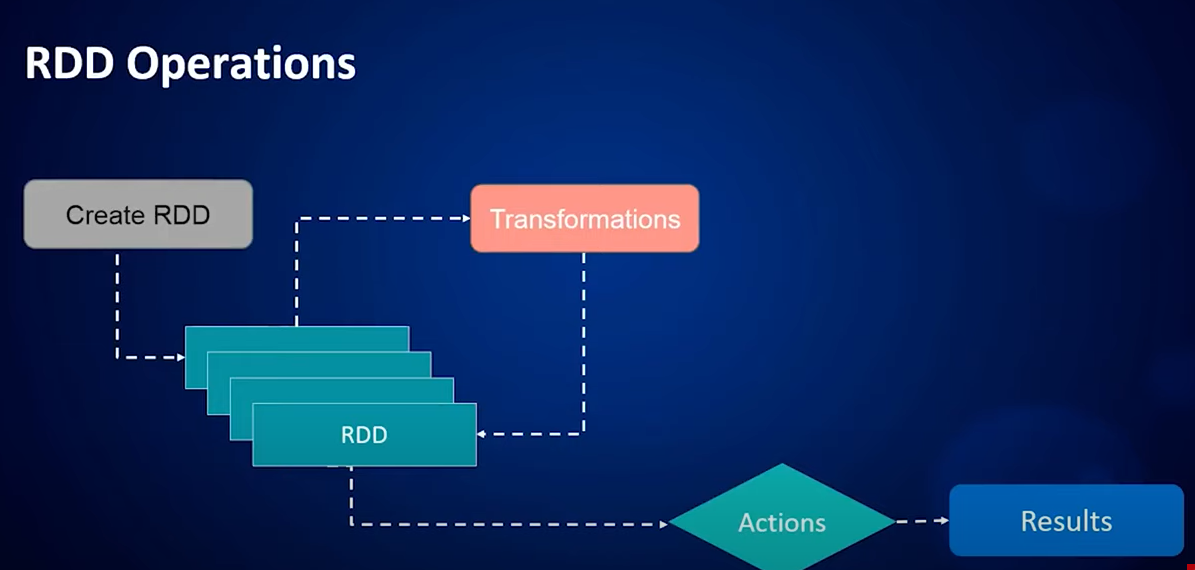


RDD operations:

1)Transformations

2)Actions

As RDD are immutable when we perform transformation that will be executed on old one and returns new one



Spark Architecture:

Spark Consists of RDD and DAG



Spark architecture primarily consists of spark context which is the entry point for any spark application just like how we establish data base connection before writing code in database to perform any operation we need to have spark context such that all functionalities related to our spark code

Spark User code🡪DAG🡪physical execution plan🡪Task🡪sent to cluster

“Install Hadoop before”

Creation of spark application using scala:

Task is to count the number of word in file which is in hdfs directory