

# Spark Standard Group-by

Java implimentation of standard Group-by for Apache Spark.

## Prerequisites

To run the algorithms you will need Java version 8, Spark version 2.3.2, and the latest version of Eclipse.

Spark 2.3.2 is incompatible with newer versions of Java, i.e. 9 >, so the use of Java 8 is a strict requirement.

## Local Mode Walkthrough

1. Download both the java files and the dimension 200, SF1 through SF5 files.
2. Create a new project in Eclipse and place the java files in the `src` folder.
3. Add the Spark jars to your project.
4. To run the algorithm in Eclipses (local mode), replace line 27 with

```
SparkConf conf = new SparkConf().setAppName("SparkGroupBy").setMaster("local[*]");
```

5. In your run configuration, copy and paste the following parameter values.

Parameters:

```
80
200
1
path/to/the/input
path/to/the/output
```

Refer to lines 37-42 for a description of the input parameters.

Since Spark is running locally, simply specify the input path as the location where you choose to store the files on your system. The output path can be whatever you specify given that the directory does not already exist.

6. Hit run in Eclipse.

## Cluster Mode Walkthrough

1. Download both the java files and the dimension 200, SF1 through SF5 files.
2. Create a new project in Eclipse and place the java files in the `src` folder.
3. Add the Spark jars to your project.
4. Export the project the project to a jar and add place the jar in your cluster.
5. As part of the submission, copy and paste the following parameter values.

Parameters:

80

200

1

path/to/the/input

path/to/the/output

Refer to lines 37-42 for a description of the input parameters.

Since Spark is running in cluster mode, place the data files into your cluster and specify the location as part of your input values. The output path can be whatever you specify given that the directory does not already exist.

6. Submit the job.