Home (/) About (https://sparkbyexamples.com/about-sparkbyexamples/) | { One stop for all Spark Examples }

Spark by {Examples} (https://sparkbyexamples.com/)

Spark (https://sparkbyexamples.com/)

PySpark (https://sparkbyexamples.com/pyspark-tutorial/)

Hive (https://sparkbyexamples.com/apache-hive-tutorial/)

HBase (https://sparkbyexamples.com/apache-hbase-tutorial/)

Kafka (https://sparkbyexamples.com/apache-kafka-tutorials-with-examples/)

FAQ's (https://sparkbyexamples.com/spark-questions/)

More (https://sparkbyexamples.com/)

Spark Tutorial

Spark – Installation on Windows (https://sparkbyexamples.com/spark/apache-spark-installation-on-windows/)

Spark – Installation on Linux | Ubuntu (https://sparkbyexamples.com/spark/spark-installation-on-linux-ubuntu/)

Spark – Cluster Setup with Hadoop Yarn (https://sparkbyexamples.com/spark/spark-setup-on-hadoop-yarn/)

Spark – Web/Application UI (https://sparkbyexamples.com/spark/spark-web-ui-understanding/)

Spark – Setup with Scala and IntelliJ (https://sparkbyexamples.com/spark/spark-setup-run-with-scala-intellij/)

Spark – How to Run Examples From this Site on IntelliJ IDEA (https://sparkbyexamples.com/spark/how-to-run-spark-examples-from-intellij/)

Spark – SparkSession (https://sparkbyexamples.com/spark/sparksession-explained-with-examples/)

Spark – SparkContext (https://sparkbyexamples.com/spark/spark-sparkcontext/)

Spark RDD Tutorial

Spark RDD – Parallelize (https://sparkbyexamples.com/apache-spark-rdd/how-to-create-an-rdd-using-parallelize/)

Spark RDD – Read text file (https://sparkbyexamples.com/apache-spark-rdd/spark-read-multiple-text-files-into-a-single-rdd/)

Spark map() vs mapPartitions() with Examples

NNK (https://sparkbyexamples.com/author/admin/) -

Apache Spark (https://sparkbyexamples.com/category/spark/)

Spark `map()` and `mapPartitions()` transformations apply the function on each element/record/row of the DataFrame/Dataset and returns the new DataFrame/Dataset, In this article, I will explain the difference between `map()` vs `mapPartitions()` transformations, their syntax, and usages with Scala examples.

- map() – Spark `map()` transformation applies a function to each row in a DataFrame/Dataset and returns the new transformed `Dataset`.

[Spark RDD – Read CSV (https://sparkbyexamples.com/apache-spark-rdd/spark-load-csv-file-into-rdd/)](https://sparkbyexamples.com/apache-spark-rdd/spark-load-csv-file-into-rdd/)

[Spark RDD – Create RDD (https://sparkbyexamples.com/apache-spark-rdd/different-ways-to-create-spark-rdd/)](https://sparkbyexamples.com/apache-spark-rdd/different-ways-to-create-spark-rdd/)

[Spark RDD – Create Empty RDD (https://sparkbyexamples.com/apache-spark-rdd/spark-how-to-create-an-empty-rdd/)](https://sparkbyexamples.com/apache-spark-rdd/spark-how-to-create-an-empty-rdd/)

[Spark RDD – Transformations (https://sparkbyexamples.com/apache-spark-rdd/spark-rdd-transformations/)](https://sparkbyexamples.com/apache-spark-rdd/spark-rdd-transformations/)

[Spark RDD – Actions (https://sparkbyexamples.com/apache-spark-rdd/spark-rdd-actions/)](https://sparkbyexamples.com/apache-spark-rdd/spark-rdd-actions/)

[Spark RDD – Pair Functions (https://sparkbyexamples.com/apache-spark-rdd/spark-pair-rdd-functions/)](https://sparkbyexamples.com/apache-spark-rdd/spark-pair-rdd-functions/)

[Spark RDD – Repartition and Coalesce (https://sparkbyexamples.com/spark/spark-repartition-vs-coalesce/)](https://sparkbyexamples.com/spark/spark-repartition-vs-coalesce/)

[Spark RDD – Shuffle Partitions (https://sparkbyexamples.com/spark/spark-shuffle-partitions/)](https://sparkbyexamples.com/spark/spark-shuffle-partitions/)

[Spark RDD – Cache vs Persist (https://sparkbyexamples.com/spark/spark-difference-between-cache-and-persist/)](https://sparkbyexamples.com/spark/spark-difference-between-cache-and-persist/)

[Spark RDD – Persistance Storage Levels (https://sparkbyexamples.com/spark/spark-persistence-storage-levels/)](https://sparkbyexamples.com/spark/spark-persistence-storage-levels/)

[Spark RDD – Broadcast Variables (https://sparkbyexamples.com/spark/spark-broadcast-variables/)](https://sparkbyexamples.com/spark/spark-broadcast-variables/)

[Spark RDD – Accumulator Variables (https://sparkbyexamples.com/spark/spark-accumulators/)](https://sparkbyexamples.com/spark/spark-accumulators/)

[Spark RDD – Convert RDD to DataFrame (https://sparkbyexamples.com/](https://sparkbyexamples.com/)

- mapPartitions() – This is exactly the same as map(); the difference being, Spark `mapPartitions()` provides a facility to do heavy initializations (for example Database connection) once for each partition instead of doing it on every DataFrame row. This helps the performance of the job when you dealing with heavy-weighted initialization on larger datasets.

Key Points:

- One key point to remember, these both transformations returns the `Dataset[U]` but not the `DataFrame` (In Spark 2.0, `DataFrame = Dataset[Row]`) .
- After applying the transformation function on each row of the input DataFrame/Dataset, these return the same number of rows as input but the schema or number of the columns of the result could be different.
- If you know `flatMap()` transformation, this is the key difference between map and flatMap where map returns only one row/element for every input, while flatMap() can return a list of rows/elements.

Spark `map()` vs `mapPartitions()` Example

Let's see the differences with example. First let's [create a Spark DataFrame (https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/)](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/)

apache-spark-rdd/convert-spark-rdd-to-dataframe-dataset/)

Spark SQL Tutorial

Spark SQL – Create DataFrame (https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/)

Spark SQL – Select Columns (https://sparkbyexamples.com/spark/spark-select-columns-from-dataframe/)

Spark SQL – Add and Update Column (withColumn) (https://sparkbyexamples.com/spark/spark-dataframe-withcolumn/)

Spark SQL – Rename Nested Column (https://sparkbyexamples.com/spark/rename-a-column-on-spark-dataframes/)

Spark SQL – Drop column (https://sparkbyexamples.com/spark/spark-drop-column-from-dataframe-dataset/)

Spark SQL – Where | Filter (https://sparkbyexamples.com/spark/spark-dataframe-where-filter/)

Spark SQL – When Otherwise (https://sparkbyexamples.com/spark/spark-case-when-otherwise-example/)

Spark SQL – Collect data to Driver (https://sparkbyexamples.com/spark/spark-dataframe-collect/)

Spark SQL – Distinct (https://sparkbyexamples.com/spark/spark-remove-duplicate-rows/)

Spark SQL- Pivot Table DataFrame (https://sparkbyexamples.com/spark/how-to-pivot-table-and-unpivot-a-spark-dataframe/)

Spark SQL – Data Types (https://sparkbyexamples.com/spark/spark-sql-dataframe-data-types/)

```
val structureData = Seq(
  Row("James","","Smith","366
  Row("Michael","Rose","","40
  Row("Robert","","Williams",
  Row("Maria","Anne","Jones",
  Row("Jen","Mary","Brown","3
)

val structureSchema = new Str
  .add("firstname",StringType
  .add("middlename",StringTyp
  .add("lastname",StringType)
  .add("id",StringType)
  .add("location",StringType)
  .add("salary",IntegerType)

val df2 = spark.createDataFra
  spark.sparkContext.parallel
df2.printSchema()
df2.show(false)
```

Yields below output

```
root
 |-- firstname: string (nullabl
 |-- middlename: string (nullab
 |-- lastname: string (nullable
 |-- id: string (nullable = tru
 |-- location: string (nullable
 |-- salary: integer (nullable 

+---------+----------+--------+
|firstname|middlename|lastname|
+---------+----------+--------+
|James    |          |Smith   |
|Michael  |Rose      |        |
|Robert   |          |Williams|
|Maria    |Anne      |Jones   |
|Jen      |Mary      |Brown   |
+---------+----------+--------+
```

In order to explain `map()` and `mapPartitions()` with an example, let's also create a "`Util`" class with a method `combine()`, this is a simple method that takes three string arguments and combines them with a comma delimiter. In realtime, this could be a third-party class that does complex transformation.

Spark SQL – StructType | StructField (https://sparkbyexamples.com/spark/spark-sql-structtype-on-dataframe/)

Spark SQL – Schema (https://sparkbyexamples.com/spark/spark-schema-explained-with-examples/)

Spark SQL – Groupby (https://sparkbyexamples.com/spark/using-groupby-on-dataframe/)

Spark SQL – Sort DataFrame (https://sparkbyexamples.com/spark/spark-how-to-sort-dataframe-column-explained/)

Spark SQL – Join Types (https://sparkbyexamples.com/spark/spark-sql-dataframe-join/)

Spark SQL – Union and UnionAll (https://sparkbyexamples.com/spark/spark-dataframe-union-and-union-all/)

Spark SQL – map() vs mapPartitions() (https://sparkbyexamples.com/spark/spark-map-vs-mappartitions-transformation/)

Spark SQL – foreach() vs foreachPartition() (https://sparkbyexamples.com/spark/spark-foreachpartition-vs-foreach-explained/)

Spark SQL – map() vs flatMap() (https://sparkbyexamples.com/spark/spark-map-vs-flatmap-with-examples/)

Spark SQL – Persist and Cache (https://sparkbyexamples.com/spark/spark-dataframe-cache-and-persist-explained/)

Spark SQL – UDF (User Defined Functions) (https://sparkbyexamples.com/spark/spark-sql-udf/)

Spark SQL – Array (ArrayType) Column (https://sparkbyexamples.com/spark/spark-array-arraytype-dataframe-column/)

```
class Util extends Serializable
  def combine(fname:String,mnam
    fname+","+mname+","+lname
  }
}
```

We will create an object for this class by initializing and call the `combine()` method for each row in a DataFrame.

Spark map() transformation

Spark `map()` transformation applies a function to each row in a DataFrame/Dataset and returns the new transformed `Dataset`. As mentioned earlier, map() returns one row for every row in a input DataFrame, in other words, input and the result exactly contains the same number of rows.

For example, if you have 100 rows in a DataFrame, after applying the function map() return with exactly 100 rows. However, the structure or schema of the result could be different.

Syntax:

```
1) map[U](func : scala.Function
        : org.apache.spark.sql.
2) map[U](func : org.apache.spa
        : org.apache.spark.sql.
```

Spark provides 2 map transformation signatures one takes `scala.function1` as argument and

Spark SQL – Map (MapType) column (https://sparkbyexamples.com/spark/spark-dataframe-map-maptype-column/)

Spark SQL – Flatten Nested Struct Column (https://sparkbyexamples.com/spark/spark-flatten-nested-struct-column/)

Spark SQL – Flatten Nested Array Column (https://sparkbyexamples.com/spark/spark-flatten-nested-array-column-to-single-column/)

Spark SQL – Explode Array & Map Columns (https://sparkbyexamples.com/spark/explode-spark-array-and-map-dataframe-column/)

Spark SQL – Sampling (https://sparkbyexamples.com/spark/spark-sampling-with-examples/)

Spark SQL – Partitioning (https://sparkbyexamples.com/spark/spark-partitioning-understanding/)

Spark SQL Functions

Spark SQL String Functions (https://sparkbyexamples.com/spark/usage-of-spark-sql-string-functions/)

Spark SQL Date and Timestamp Functions (https://sparkbyexamples.com/spark/spark-sql-date-and-time-functions/)

Spark SQL Array Functions (https://sparkbyexamples.com/spark/spark-sql-array-functions/)

Spark SQL Map Functions (https://sparkbyexamples.com/spark/spark-sql-map-functions/)

Spark SQL Sort Functions (https://sparkbyexamples.com/spark/spark-sql-sort-functions/)

Spark SQL Aggregate Functions (https://sparkbyexamples.com/

the other takes `MapFunction` and if you notice both these functions return Dataset[U] but not DataFrame (which is Dataset[Row]). If you want a DataFrame as output then you need to convert the Dataset to DataFrame using `toDF()` function.

Usage:

```
import spark.implicits._
val df3 = df2.map(row=>{
  // This initialization happen
  // If it is heavy initilizati
  // It degrates the performanc
  val util = new Util()
  val fullName = util.combine(r
              (fullName, row.get
})
val df3Map =  df3.toDF("fullNam

df3Map.printSchema()
df3Map.show(false)
```

Since map transformations execute on worker nodes, we have initialized and create an object of the `Util` class inside the map() function and the initialization happens for every row in a DataFrame. This causes performance issues when you have heavily weighted initializations.

Note: When you running it on Standalone mode, initializing the class outside of the map() still works as both executors and driver run on the same JVM but running this on cluster fails with exception.

Above example yields below output.

spark/spark-sql-aggregate-functions/)

Spark SQL Window Functions (https://sparkbyexamples.com/spark/spark-sql-window-functions/)

Spark SQL JSON Functions (https://sparkbyexamples.com/spark/spark-most-used-json-functions-with-examples/)

Spark Data Source API

Spark – Read & Write CSV file (https://sparkbyexamples.com/spark/spark-read-csv-file-into-dataframe/)

Spark – Read and Write JSON file (https://sparkbyexamples.com/spark/spark-read-and-write-json-file/)

Spark – Read & Write Parquet file (https://sparkbyexamples.com/spark/spark-read-write-dataframe-parquet-example/)

Spark – Read & Write XML file (https://sparkbyexamples.com/spark/spark-read-write-xml/)

Spark – Read & Write Avro files (https://sparkbyexamples.com/spark/read-write-avro-file-spark-dataframe/)

Spark – Read & Write Avro files (Spark version 2.3.x or earlier) (https://sparkbyexamples.com/spark/using-avro-data-files-from-spark-sql-2-3-x/)

Spark – Read & Write HBase using “hbase-spark” Connector (https://sparkbyexamples.com/spark/spark-read-write-using-hbase-spark-connector/)

Spark – Read & Write from HBase using Hortonworks (https://sparkbyexamples.com/spark/create-spark-dataframe-from-hbase-using-hortonworks/)

Spark – Read & Write ORC file (https://sparkbyexamples.com/spark/spark-read-orc-file-into-dataframe/)

```
root
 |-- fullName: string (nullable
 |-- id: string (nullable = tru
 |-- salary: integer (nullable

+----------------+-----+------+
|fullName        |id   |salary|
+----------------+-----+------+
|James,,Smith    |36636|3100  |
|Michael,Rose,   |40288|4300  |
|Robert,,Williams|42114|1400  |
|Maria,Anne,Jones|39192|5500  |
|Jen,Mary,Brown  |34561|3000  |
+----------------+-----+------+
```

As you notice the above output, the input of the DataFrame has 5 rows so the result of the map also has 5 but the column counts are different.

Spark mapPartitions() transformation

Spark `mapPartitions()` provides a facility to do heavy initializations (for example Database connection) once for each partition instead of doing it on every DataFrame row. This helps the performance of the job when you dealing with heavy-weighted initialization on larger datasets.

Syntax:

```
1) mapPartitions[U](func : scal
          : org.apache.spark.sql.
2) mapPartitions[U](f : org.apa
          : org.apache.spark.sql.
```

map partitions also have 2 signatures, one take `scala.Function1` and other takes spark `MapPartitionsFunction` arguments.

mapPartitions() keeps the result of the partition in-memory until it finishes executing all rows in a partition.

Usage:

[Spark – Read Binary File (https://sparkbyexamples.com/spark/spark-read-binary-file-into-dataframe/)](https://sparkbyexamples.com/spark/spark-read-binary-file-into-dataframe/)

Spark Streaming & Kafka

[Spark Streaming – OutputModes (https://sparkbyexamples.com/spark/spark-streaming-outputmode/)](https://sparkbyexamples.com/spark/spark-streaming-outputmode/)

[Spark Streaming – Reading Files From Directory (https://sparkbyexamples.com/spark/spark-streaming-read-json-files-from-directory/)](https://sparkbyexamples.com/spark/spark-streaming-read-json-files-from-directory/)

[Spark Streaming – Reading Data From TCP Socket (https://sparkbyexamples.com/spark/spark-streaming-from-tcp-socket/)](https://sparkbyexamples.com/spark/spark-streaming-from-tcp-socket/)

[Spark Streaming – Processing Kafka Messages in JSON Format (https://sparkbyexamples.com/spark/spark-streaming-with-kafka/)](https://sparkbyexamples.com/spark/spark-streaming-with-kafka/)

[Spark Streaming – Processing Kafka messages in AVRO Format (https://sparkbyexamples.com/spark/spark-streaming-consume-and-produce-kafka-messages-in-avro-format/)](https://sparkbyexamples.com/spark/spark-streaming-consume-and-produce-kafka-messages-in-avro-format/)

[Spark SQL Batch – Consume & Produce Kafka Message (https://sparkbyexamples.com/spark/spark-batch-processing-produce-consume-kafka-topic/)](https://sparkbyexamples.com/spark/spark-batch-processing-produce-consume-kafka-topic/)

```
val df4 = df2.mapPartitions(i
  // Do the heavy initializat
  // Like database connection
  val util = new Util()
  val res = iterator.map(row=
    val fullName = util.combi
    (fullName, row.getString(
  })
  res
})
val df4part = df4.toDF("fullN
df4part.printSchema()
df4part.show(false)
```

This yields the same output as above.

Complete example of Spark DataFrame map() & mapPartitions()

Below is complete example of Spark DataFrame map() & mapPartition() example.

```
ackage com.sparkbyexamples.spa

mport org.apache.spark.sql.{Ro
mport org.apache.spark.sql.typ

bject MapTransformation extend

  val spark:SparkSession = Spar
    .master("local[5]")
    .appName("SparkByExamples.c
    .getOrCreate()

  val structureData = Seq(
    Row("James","","Smith","366
    Row("Michael","Rose","","40
    Row("Robert","","Williams",
    Row("Maria","Anne","Jones",
    Row("Jen","Mary","Brown","3
  )

  val structureSchema = new Str
    .add("firstname",StringType
    .add("middlename",StringTyp
    .add("lastname",StringType)
    .add("id",StringType)
    .add("location",StringType)
    .add("salary",IntegerType)

  val df2 = spark.createDataFra
    spark.sparkContext.parallel
  df2.printSchema()
  df2.show(false)

  import spark.implicits._
  val util = new Util()
  val df3 = df2.map(row=>{

    val fullName = util.combine
    (fullName, row.getString(3)
  })
  val df3Map =  df3.toDF("fullN

  df3Map.printSchema()
  df3Map.show(false)

  val df4 = df2.mapPartitions(i
    val util = new Util()
    val res = iterator.map(row=
      val fullName = util.combi
      (fullName, row.getString(
    })
    res
  })
  val df4part = df4.toDF("fullN
  df4part.printSchema()
  df4part.show(false)
}
```

This example is also available at Spark Example github project (https://github.com/spark-examples/spark-scala-examples/blob/master/src/main/scala/com/sparkbyexamples/spark/dataframe/examples/MapTransformation.scala)

Conclusion

In this Spark DataFrame article, you have learned map() and mapPartitions() transformations execute a function on each and every row and returns the same number of records as in input but with the same or different schema or columns. Also learned when you have a complex initialization you should be using mapPratitions() as it has the capability to do initializations once for each partition instead of every DataFrame row..

Thanks for reading. Leave me a comment if you like this article.

Happy Learning !!

Share this:

(https://sparkbyexamples.com/spark/spark-map-vs-mappartitions-transformation/?share=facebook&nb=1)

(https://sparkbyexamples.com/spark/spark-map-vs-mappartitions-transformation/?share=reddit&nb=1)

(https://sparkbyexamples.com/spark/spark-map-vs-mappartitions-transformation/?share=pinterest&nb=1)

(https://sparkbyexamples.com/spark/spark-map-vs-mappartitions-transformation/?share=tumblr&nb=1)

(https://sparkbyexamples.com/spark/spark-map-vs-mappartitions-transformation/?share=pocket&nb=1)

(https://sparkbyexamples.com/spark/spark-map-vs-mappartitions-transformation/?share=linkedin&nb=1)

(https://sparkbyexamples.com/spark/spark-map-vs-mappartitions-transformation/?share=twitter&nb=1)

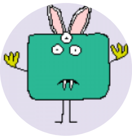
TAGS: MAP() (HTTPS://SPARKBYEXAMPLES.COM/TAG/MAP-2/), MAPPARTITIONS() (HTTPS://SPARKBYEXAMPLES.COM/TAG/MAPPARTITIONS/)

(https://sparkbyexamples.com/author/admin/)

NNK (Https://Sparkbyexamples.Com/Author/Admin/)

SparkByExamples.com is a Big Data and Spark examples community page, all examples are simple and easy to understand and well tested in our development environment Read more .. (https://sparkbyexamples.com/about-sparkbyexamples/)

THIS POST HAS 2 COMMENTS



PANKAJ

13 NOV 2020 REPLY

Good Article. Thanks for sharing

NNK 13 NOV 2020 REPLY

Thanks Pankaj.

Leave a Reply

Enter your comment here...

Categories

Apache Hadoop (https://sparkbyexamples.com/category/hadoop/)

Apache Spark (https://sparkbyexamples.com/category/spark/)

Apache Spark Streaming (https://sparkbyexamples.com/category/spark/apache-spark-streaming/)

Apache Kafka (https://sparkbyexamples.com/category/kafka/)

Recent Posts

Spark regexp_replace() – Replace String Value (https://sparkbyexamples.com/spark/spark-regexp_replace-replace-string-value/)

How to Run a PySpark Script from Python? (https://sparkbyexamples.com/pyspark/run-pyspark-script-from-python-subprocess/)

Spark SQL like() Using Wildcard Example (https://sparkbyexamples.com/spark/spark-sql-like-using-wildcard-example/)

About SparkByExamples.Com

SparkByExamples.com is a Big Data and Spark examples community page, all examples are simple and easy to understand, and well tested in our development environment Read more .. (https://sparkbyexamples.com/about-sparkbyexamples/)

Follow Us

Apache HBase (https://sparkbyexamples.com/category/hbase/)

Apache Cassandra (https://sparkbyexamples.com/category/cassandra/)

Snowflake Database (https://sparkbyexamples.com/category/snowflake/)

H2O Sparkling Water (https://sparkbyexamples.com/category/h2o-sparkling-water/)

PySpark (https://sparkbyexamples.com/category/pyspark/)

Spark isin() & IS NOT IN Operator Example (https://sparkbyexamples.com/spark/spark-isin-is-not-in-operator-example/)

Spark – Get Size/Length of Array & Map Column (https://sparkbyexamples.com/spark/spark-get-size-length-of-array-map-column/)

Spark Using Length/Size Of a DataFrame Column (https://sparkbyexamples.com/spark/spark-using-length-size-of-a-dataframe-column/)

Spark rlike() Working with Regex Matching Examples (https://sparkbyexamples.com/spark/spark-rlike-regex-matching-examples/)

Spark Check String Column Has Numeric Values (https://sparkbyexamples.com/spark/spark-check-string-column-has-numeric-values/)

Spark Check Column Data Type is Integer or String (https://sparkbyexamples.com/spark/spark-check-column-data-type-is-integer-or-string/)

(https://twitter.com/sparkbyexamples)

(https://www.facebook.com/sparkbyexamples/)

(https://www.linkedin.com/in/n-nk-b860a8193/)

(https://github.com/spark-examples/)