

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

- PySpark Tutorial**
 - [PySpark Tutorial For Beginners \(https://sparkbyexamples.com/pyspark-tutorial/\)](https://sparkbyexamples.com/pyspark-tutorial/)
 - [PySpark – Features \(https://sparkbyexamples.com/pyspark-tutorial/#features\)](https://sparkbyexamples.com/pyspark-tutorial/#features)
 - [PySpark – Advantages \(https://sparkbyexamples.com/pyspark-tutorial/#advantages\)](https://sparkbyexamples.com/pyspark-tutorial/#advantages)
 - [PySpark – Modules & Packages \(https://sparkbyexamples.com/pyspark-tutorial/#modules-packages\)](https://sparkbyexamples.com/pyspark-tutorial/#modules-packages)
 - [PySpark – Cluster Managers \(https://sparkbyexamples.com/pyspark-tutorial/#cluster-manager\)](https://sparkbyexamples.com/pyspark-tutorial/#cluster-manager)
 - [PySpark – Install on Windows \(https://sparkbyexamples.com/pyspark-tutorial/#pyspark-installation\)](https://sparkbyexamples.com/pyspark-tutorial/#pyspark-installation)
 - [PySpark – Web/Application UI \(https://sparkbyexamples.com/spark/spark-web-ui-understanding/\)](https://sparkbyexamples.com/spark/spark-web-ui-understanding/)
 - [PySpark – SparkSession \(https://sparkbyexamples.com/pyspark/pyspark-what-is-sparksession/\)](https://sparkbyexamples.com/pyspark/pyspark-what-is-sparksession/)
 - [PySpark – RDD \(https://sparkbyexamples.com/pyspark-rdd\)](https://sparkbyexamples.com/pyspark-rdd)
 - [PySpark – Parallelize \(https://sparkbyexamples.com/pyspark/pyspark-parallelize-create-rdd/\)](https://sparkbyexamples.com/pyspark/pyspark-parallelize-create-rdd/)
 - [PySpark – repartition\(\) vs coalesce\(\) \(https://sparkbyexamples.com/pyspark/pyspark-repartition-vs-coalesce/\)](https://sparkbyexamples.com/pyspark/pyspark-repartition-vs-coalesce/)
 - [PySpark – Broadcast Variables \(https://sparkbyexamples.com/pyspark/pyspark-broadcast-variables/\)](https://sparkbyexamples.com/pyspark/pyspark-broadcast-variables/)

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-by-examples-faq/)

More (https://sparkbyexamples.com/)

IABAC Certification - Global
Science Course
Datamites - Data Science Courses

PySpark GroupBy
Explained with
Example

PySpark

groupBy() function is used to collect the identical data into groups on DataFrame and perform aggregate functions on the grouped data. In this article, I will explain several groupBy() examples using PySpark (Spark with Python).

IABAC Certification - Global Data Science Course

Datamites - Data Science Courses in India

6-month/400 learning hours, 120-hour training, capstone & client projects

WEBSITE

Related: [How to group and aggregate data using Spark and Scala \(https://sparkbyexamples.com/spark/using-groupby-on-dataframe/\)](https://sparkbyexamples.com/spark/using-groupby-on-dataframe/)

Syntax:

Laptop Bag Offer by Skechers

Skechers - WHC Road, Dharampeth, Nagpur

Get a Free Laptop Bag! Shop for ₹6000+
Store T&C*



Nagpur

STORE INFO



How to Analyze Other Websites' Traffic



[PySpark – Accumulator
\(https://sparkbyexamples.com/pyspark/pyspark-accumulator-with-example/\)](https://sparkbyexamples.com/pyspark/pyspark-accumulator-with-example/)

PySpark DataFrame

[PySpark – Create a DataFrame
\(https://sparkbyexamples.com/pyspark/different-ways-to-create-dataframe-in-pyspark/\)](https://sparkbyexamples.com/pyspark/different-ways-to-create-dataframe-in-pyspark/)

[PySpark – Create an empty DataFrame
\(https://sparkbyexamples.com/pyspark/pyspark-create-an-empty-dataframe/\)](https://sparkbyexamples.com/pyspark/pyspark-create-an-empty-dataframe/)

[PySpark – Convert RDD to DataFrame
\(https://sparkbyexamples.com/pyspark/convert-pyspark-rdd-to-dataframe/\)](https://sparkbyexamples.com/pyspark/convert-pyspark-rdd-to-dataframe/)

[PySpark – Convert DataFrame to Pandas
\(https://sparkbyexamples.com/pyspark/convert-pyspark-dataframe-to-pandas/\)](https://sparkbyexamples.com/pyspark/convert-pyspark-dataframe-to-pandas/)

[PySpark – show\(\)
\(https://sparkbyexamples.com/pyspark/pyspark-show-display-dataframe-contents-in-table/\)](https://sparkbyexamples.com/pyspark/pyspark-show-display-dataframe-contents-in-table/)

[PySpark – StructType & StructField
\(https://sparkbyexamples.com/pyspark/pyspark-structtype-and-structfield/\)](https://sparkbyexamples.com/pyspark/pyspark-structtype-and-structfield/)

[PySpark – Row Class
\(https://sparkbyexamples.com/pyspark/pyspark-row-using-rdd-dataframe/\)](https://sparkbyexamples.com/pyspark/pyspark-row-using-rdd-dataframe/)

[PySpark – Column Class
\(https://sparkbyexamples.com/pyspark/pyspark-column-functions/\)](https://sparkbyexamples.com/pyspark/pyspark-column-functions/)

[PySpark – select\(\)
\(https://sparkbyexamples.com/pyspark/select-columns-from-pyspark-dataframe/\)](https://sparkbyexamples.com/pyspark/select-columns-from-pyspark-dataframe/)

[PySpark – collect\(\)
\(https://sparkbyexamples.com/pyspark/pyspark-collect/\)](https://sparkbyexamples.com/pyspark/pyspark-collect/)

[PySpark – withColumn\(\)
\(https://sparkbyexamples.com/pyspark/pyspark-withcolumn/\)](https://sparkbyexamples.com/pyspark/pyspark-withcolumn/)

```
groupBy(col1 : scala.Predef.Str  
        org.apache.spark.sql.Rela
```

When we perform `groupBy()` on PySpark DataFrame, it returns `GroupedData` object which contains below aggregate functions.

`count()` - Returns the count of rows for each group.

Our combined feed & effluent heat exchanger will improve your process



`mean()` - Returns the mean of values for each group.

`max()` - Returns the maximum of values for each group.

`min()` - Returns the minimum of values for each group.

`sum()` - Returns the total for values for each group.

`avg()` - Returns the average for values for each group.

Our combined feed & effluent heat exchanger will improve your process



`agg()` - Using [agg\(\)](#) function, we can calculate more than one aggregate at a time.

[PySpark – withColumnRenamed\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-rename-dataframe-column/>).

[PySpark – where\(\) & filter\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-where-filter/>).

[PySpark – drop\(\) & dropDuplicates\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-distinct-to-drop-duplicates/>).

[PySpark – orderBy\(\) and sort\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-orderby-and-sort-explained/>).

[PySpark – groupBy\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-groupby-explained-with-example/>).

[PySpark – join\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-join-explained-with-examples/>).

[PySpark – union\(\) & unionAll\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-union-and-unionall/>).

[PySpark – unionByName\(\).](#)
(<https://sparkbyexamples.com/spark/spark-merge-two-dataframes-with-different-columns/>).

[PySpark – UDF \(User Defined Function\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-udf-user-defined-function/>).

[PySpark – map\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-map-transformation/>).

[PySpark – flatMap\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-flatmap-transformation/>).

[pyspark – foreach\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-loop-iterate-through-rows-in-dataframe/#use-foreach-loop-through-dataframe>).

[PySpark – sample\(\) vs sampleBy\(\).](#)

`pivot()` - This function is used to Pivot the DataFrame which I will not be covered in this article as I already have a dedicated article for [Pivot & Unpivot DataFrame](#)
(<https://sparkbyexamples.com/spark/how-to-pivot-table-and-unpivot-a-spark-dataframe/>).

Preparing Data & creating DataFrame

Before we start, let's [create the DataFrame](#)
(<https://sparkbyexamples.com/pyspark/different-ways-to-create-dataframe-in-pyspark/>) from a sequence of the data to work with. This DataFrame contains columns “employee_name”, “department”, “state”, “salary”, “age” and “bonus” columns.

We will use this PySpark DataFrame to run `groupBy()` on “department” columns and calculate aggregates like minimum, maximum, average, total salary for each group using `min()`, `max()` and `sum()` aggregate functions respectively. and finally, we will also see how to do group and aggregate on multiple columns.

```
simpleData = [("James","Sales",
              ("Michael","Sales","NY",8600),
              ("Robert","Sales","CA",81000),
              ("Maria","Finance","CA",9000),
              ("Raman","Finance","CA",9900),
              ("Scott","Finance","NY",8300),
              ("Jen","Finance","NY",79000),
              ("Jeff","Marketing","CA",8000),
              ("Kumar","Marketing","NY",9000)
              ]

schema = ["employee_name","depart
df = spark.createDataFrame(data=
df.printSchema()
df.show(truncate=False)
```

Yields below output.

[\(https://sparkbyexamples.com/pyspark/pyspark-sampling-example/\)](https://sparkbyexamples.com/pyspark/pyspark-sampling-example/).

[PySpark – fillna\(\) & fill\(\). \(https://sparkbyexamples.com/pyspark/pyspark-fillna-fill-replace-null-values/\)](https://sparkbyexamples.com/pyspark/pyspark-fillna-fill-replace-null-values/).

[PySpark – pivot\(\)_\(Row to Column\). \(https://sparkbyexamples.com/pyspark/pyspark-pivot-and-unpivot-dataframe/\)](https://sparkbyexamples.com/pyspark/pyspark-pivot-and-unpivot-dataframe/).

[PySpark – partitionBy\(\). \(https://sparkbyexamples.com/pyspark/pyspark-partitionby-example/\)](https://sparkbyexamples.com/pyspark/pyspark-partitionby-example/).

[PySpark – ArrayType Column \(Array\). \(https://sparkbyexamples.com/pyspark/pyspark-arraytype-column-with-examples/\)](https://sparkbyexamples.com/pyspark/pyspark-arraytype-column-with-examples/).

[PySpark – MapType \(Map/Dict\). \(https://sparkbyexamples.com/pyspark/pyspark-maptype-dict-examples/\)](https://sparkbyexamples.com/pyspark/pyspark-maptype-dict-examples/).

PySpark SQL Functions

[PySpark – Aggregate Functions \(https://sparkbyexamples.com/pyspark/pyspark-aggregate-functions/\)](https://sparkbyexamples.com/pyspark/pyspark-aggregate-functions/).

[PySpark – Window Functions \(https://sparkbyexamples.com/pyspark/pyspark-window-functions/\)](https://sparkbyexamples.com/pyspark/pyspark-window-functions/).

[PySpark – Date and Timestamp Functions \(https://sparkbyexamples.com/pyspark/pyspark-sql-date-and-timestamp-functions/\)](https://sparkbyexamples.com/pyspark/pyspark-sql-date-and-timestamp-functions/).

[PySpark – JSON Functions \(https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/\)](https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/).

PySpark Datasources

[PySpark – Read & Write CSV File \(https://sparkbyexamples.com/pyspark/pyspark-read-csv-file-into-dataframe/\)](https://sparkbyexamples.com/pyspark/pyspark-read-csv-file-into-dataframe/).

[PySpark – Read & Write Parquet File \(https://sparkbyexamples.com/pyspark/pyspark-read-write-parquet-file/\)](https://sparkbyexamples.com/pyspark/pyspark-read-write-parquet-file/).

```
+-----+-----+-----+
|employee_name|department|state|
+-----+-----+-----+
|      James |      Sales|   NY|
|    Michael |      Sales|   NY|
|    Robert  |      Sales|   CA|
|     Maria  |   Finance|   CA|
|     Raman  |   Finance|   CA|
|     Scott  |   Finance|   NY|
|        Jen |   Finance|   NY|
|      Jeff  |Marketing|   CA|
|     Kumar  |Marketing|   NY|
+-----+-----+-----+
```

PySpark groupBy and aggregate on DataFrame columns

Let's do the `groupBy()` on `department` column of `DataFrame` and then find the sum of salary for each department using `sum()` aggregate function.

```
df.groupBy("department").sum("salary")
+-----+-----+
|department|sum(salary)|
+-----+-----+
|Sales     |257000     |
|Finance   |351000     |
|Marketing |171000     |
+-----+-----+
```

Similarly, we can calculate the number of employee in each department using `count()`

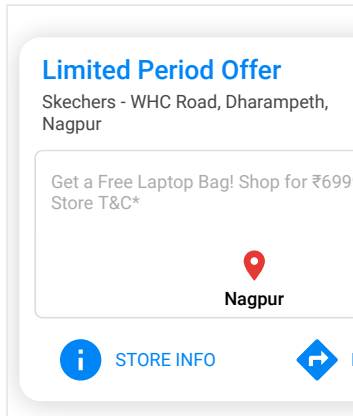
```
df.groupBy("department").count()
+-----+-----+
|department|count|
+-----+-----+
```

Calculate the minimum salary of each department using `min()`

```
df.groupBy("department").min("salary")
+-----+-----+
|department|min(salary)|
+-----+-----+
```

[pyspark/pyspark-read-and-write-parquet-file/](#)

[PySpark – Read & Write JSON file](#)
(<https://sparkbyexamples.com/pyspark/pyspark-read-json-file-into-dataframe/>)



Calculate the maximin salary of each department using `max()`

```
df.groupBy("department").max("s
```

Calculate the average salary of each department using `avg()`

```
f.groupBy("department").avg( "
```

Calculate the mean salary of each department using `mean()`

```
f.groupBy("department").mean(
```

Spark groupBy and aggregate on multiple columns

Similarly, we can also run `groupBy` and `aggregate` on two or more DataFrame columns, below example does group by department, state and does `sum()` salary and bonus columns.

```
/*GroupBy on multiple columns
f.groupBy("department","state")
    .sum("salary","bonus") \
    .show(false)
```

; yields the below output.

PySpark Built-In Functions

[PySpark – when\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-when-otherwise/>)

[PySpark – expr\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-sql-expr-expression-function/>)

[PySpark – lit\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-lit-add-literal-constant/>).

[PySpark – split\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-convert-string-to-array-column/>).

[PySpark – concat_ws\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-convert-array-column-to-string-column/>).

[Pyspark – substring\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-substring-from-a-column/>).

[PySpark – translate\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-replace-column-values/#translate-replace-character-by-character>).

[PySpark – regexp_replace\(\).](#)
(https://sparkbyexamples.com/pyspark/pyspark-replace-column-values/#regexp_replace-replace-string-columns).

[PySpark – overlay\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-replace-column-values/#overlay-function>).

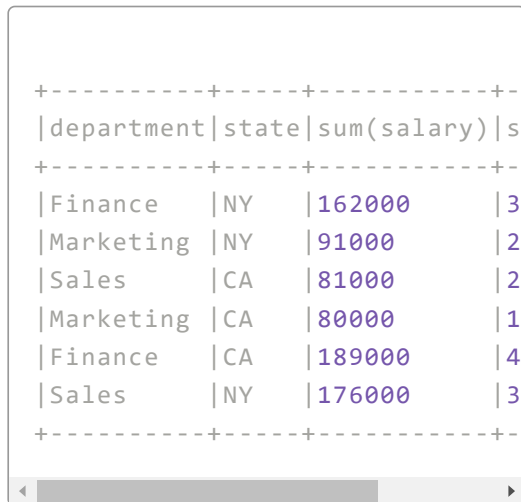
[PySpark – to_timestamp\(\).](#)
(https://sparkbyexamples.com/spark/pyspark-to_timestamp-convert-string-to-timestamp-type/).

[PySpark – to_date\(\).](#)
(https://sparkbyexamples.com/pyspark/pyspark-to_date-convert-timestamp-to-date/).

[PySpark – date_format\(\).](#)
(https://sparkbyexamples.com/pyspark/pyspark-date_format-convert-date-to-string-format/).

[PySpark – datediff\(\).](#)
(<https://sparkbyexamples.com/pyspark/pyspark-difference-between-two-dates-days-months-years/#datediff>).

[PySpark – months_between\(\).](#)
([https://sparkbyexamples.com/pyspark/pyspark-difference-between-two-dates-days-months-years/#months_between\(\)](https://sparkbyexamples.com/pyspark/pyspark-difference-between-two-dates-days-months-years/#months_between())).

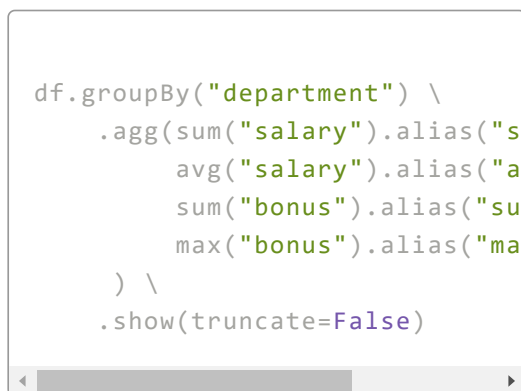


department	state	sum(salary)	sum(bonus)
Finance	NY	162000	30000
Marketing	NY	91000	20000
Sales	CA	81000	20000
Marketing	CA	80000	10000
Finance	CA	189000	40000
Sales	NY	176000	30000

similarly, we can run group by and aggregate on tow or more columns for other aggregate functions, please refer below source code for example.

Running more aggregates at a time

Using `agg()` aggregate function we can calculate many aggregations at a time on a single statement using PySpark SQL aggregate functions `sum()`, `avg()`, `min()`, `max()` `mean()` e.t.c. In order to use these, we should import "from pyspark.sql.functions import sum, avg, max, min, mean, count"



```
df.groupBy("department") \
    .agg(sum("salary").alias("sum_salary"),
         avg("salary").alias("avg_salary"),
         sum("bonus").alias("sum_bonus"),
         max("bonus").alias("max_bonus")) \
    .show(truncate=False)
```

This example does group on department column and calculates `sum()` and `avg()` of salary for each department and calculates `sum()` and `max()` of bonus for each department.

[PySpark – explode\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-explode-nested-array-into-rows/>).

[PySpark – array_contains\(\)](#)
(https://sparkbyexamples.com/pyspark/pyspark-arraytype-column-with-examples/#array_contains).

[PySpark – array\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-arraytype-column-with-examples/#array>).

[PySpark – collect_list\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-aggregate-functions/#collect-list>).

[PySpark – collect_set\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-aggregate-functions/#collect-set>).

[PySpark – create_map\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-convert-dataframe-columns-to-maptype-dict/>).

[PySpark – map_keys\(\)](#)
(https://sparkbyexamples.com/pyspark/pyspark-maptype-dict-examples/#map_keys).

[PySpark – map_values\(\)](#)
(https://sparkbyexamples.com/pyspark/pyspark-maptype-dict-examples/#map_values).

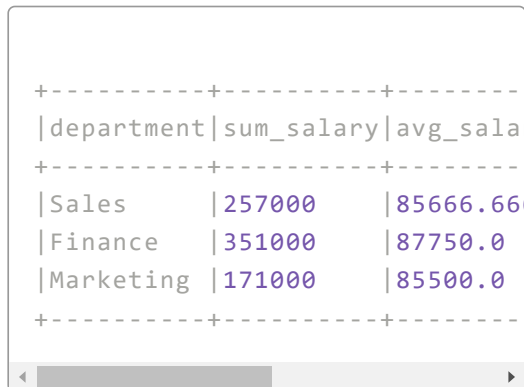
[PySpark – struct\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-structtype-and-structfield/#update-struct-function>).

[PySpark – countDistinct\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-count-distinct-from-dataframe/>).

[PySpark – sum\(\).avg\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-dataframe-groupby-and-sort-by-descending-order/>).

[PySpark – row_number\(\)](#)
(https://sparkbyexamples.com/pyspark/pyspark-window-functions/#row_number).

[PySpark – rank\(\)](#)
(<https://sparkbyexamples.com/pyspark/pyspark-window-functions/#rank>).



department	sum_salary	avg_salary
Sales	257000	85666.66666666667
Finance	351000	87750.0
Marketing	171000	85500.0

Using filter on aggregate data

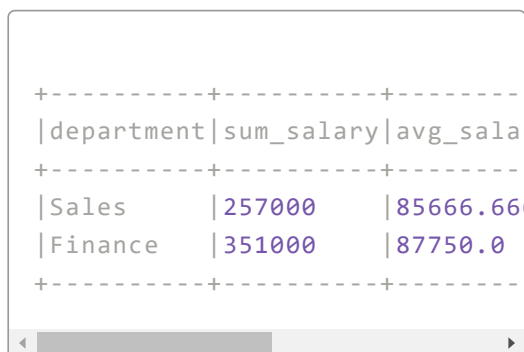
Similar to SQL “HAVING” clause, On PySpark DataFrame we can use either `where()`

(<https://sparkbyexamples.com/pyspark/pyspark-dataframe-filter/>) or `filter()` (<https://sparkbyexamples.com/pyspark/pyspark-dataframe-filter/>) function to filter the rows of aggregated data.



```
df.groupBy("department") \
    .agg(sum("salary").alias("sum_salary"),
         avg("salary").alias("avg_salary"),
         sum("bonus").alias("sum_bonus"),
         max("bonus").alias("max_bonus"))
    .where(col("sum_bonus") >= 50000)
    .show(truncate=False)
```

This removes the sum of a bonus that has less than 50000 and yields below output.



department	sum_salary	avg_salary
Sales	257000	85666.66666666667
Finance	351000	87750.0

PySpark groupBy Example Source code

[PySpark – dense_rank\(\).
\(https://sparkbyexamples.com/pyspark/pyspark-window-functions/#dense_rank\)](https://sparkbyexamples.com/pyspark/pyspark-window-functions/#dense_rank)

[PySpark – percent_rank\(\).
\(https://sparkbyexamples.com/pyspark/pyspark-window-functions/#percent_rank\)](https://sparkbyexamples.com/pyspark/pyspark-window-functions/#percent_rank)

[PySpark – typedLit\(\).
\(https://sparkbyexamples.com/pyspark/pyspark-lit-add-literal-constant/#typedlit\)](https://sparkbyexamples.com/pyspark/pyspark-lit-add-literal-constant/#typedlit)

[PySpark – from_json\(\).
\(https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#from_json\)](https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#from_json)

[PySpark – to_json\(\).
\(https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#to_json\)](https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#to_json)

[PySpark – json_tuple\(\).
\(https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#json_tuple\)](https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#json_tuple)

[PySpark – get_json_object\(\).
\(https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#get_json_object\)](https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#get_json_object)

[PySpark – schema_of_json\(\).
\(https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#schema_of_json\)](https://sparkbyexamples.com/pyspark/pyspark-json-functions-with-examples/#schema_of_json)

```
import pyspark
from pyspark.sql import SparkSession
from pyspark.sql.functions import col, sum, avg, max

spark = SparkSession.builder.appName('PySpark GroupBy').getOrCreate()

simpleData = [("James", "Sales", "NY", 86000, 15000),
              ("Michael", "Sales", "NY", 86000, 15000),
              ("Robert", "Sales", "CA", 81000, 12000),
              ("Maria", "Finance", "CA", 90000, 20000),
              ("Raman", "Finance", "CA", 99000, 22000),
              ("Scott", "Finance", "NY", 83000, 17000),
              ("Jen", "Finance", "NY", 79000, 16000),
              ("Jeff", "Marketing", "CA", 80000, 14000),
              ("Kumar", "Marketing", "NY", 91000, 18000)]

schema = ["employee_name", "department", "state"]
df = spark.createDataFrame(data=simpleData, schema=schema)
df.printSchema()
df.show(truncate=False)

df.groupBy("department").sum("salary").show(truncate=False)

df.groupBy("department").count().show(truncate=False)

df.groupBy("department", "state") \
    .sum("salary", "bonus") \
    .show(truncate=False)

df.groupBy("department") \
    .agg(sum("salary").alias("sum_salary"),
         avg("salary").alias("avg_salary"),
         sum("bonus").alias("sum_bonus"),
         max("bonus").alias("max_bonus")) \
    .show(truncate=False)

df.groupBy("department") \
    .agg(sum("salary").alias("sum_salary"),
         avg("salary").alias("avg_salary"),
         sum("bonus").alias("sum_bonus"),
         max("bonus").alias("max_bonus")) \
    .where(col("sum_bonus") >= 20000) \
    .show(truncate=False)
```

This example is also available at [GitHub PySpark Examples](https://github.com/spark-examples/pyspark-examples/blob/master/pyspark-groupby.py) [project for reference.](https://github.com/spark-examples/pyspark-examples/blob/master/pyspark-groupby.py)

Conclusion

In this tutorial, you have learned how to use `groupBy()` and `aggregate` functions on PySpark DataFrame and also learned how to run these on multiple columns and finally filtering data on the aggregated columns.

Thanks for reading. If you like it, please do share the article by following the below social links and any comments or suggestions are welcome in the comments sections!

Happy Learning !!

Share this:



(<https://sparkbyexamples.com/pyspark/pyspark-groupby-explained-with-example/?share=facebook&nb=1>)



(<https://sparkbyexamples.com/pyspark/pyspark-groupby-explained-with-example/?share=reddit&nb=1>)



(<https://sparkbyexamples.com/pyspark/pyspark-groupby-explained-with-example/?share=pinterest&nb=1>)

1



(<https://sparkbyexamples.com/pyspark/pyspark-groupby-explained-with-example/?share=tumblr&nb=1>)



(<https://sparkbyexamples.com/pyspark/pyspark-groupby-explained-with-example/?share=pocket&nb=1>)



(<https://sparkbyexamples.com/pyspark/pyspark-groupby-explained-with-example/?share=linkedin&nb=1>)



(<https://sparkbyexamples.com/pyspark/pyspark-groupby-explained-with-example/?share=twitter&nb=1>)

TAGS: [AGG](#)

([HTTPS://SPARKBYEXAMPLES.COM/TAG/AGG/](https://sparkbyexamples.com/tag/agg/)),

[GROUPBY](#)

([HTTPS://SPARKBYEXAMPLES.COM/TAG/GROUPBY/](https://sparkbyexamples.com/tag/groupby/)).



NNK

([Https://Sparkbyexamples.Com/Author/Admin/](https://sparkbyexamples.com/author/admin/))

(<https://sparkbyexamples.com>)

SparkByExamples.com is a Big Data and Spark examples community page, all examples are simple and

[mples.co](#)
[m/author/](#)
[admin/](#)

easy to understand and well tested in our development
environment [Read more ..](#)
(<https://sparkbyexamples.com/about-sparkbyexamples/>)

➤ THIS POST HAS 2 COMMENTS



Sneha 31 OCT 2020 [REPLY](#)

This is really great. I am learning pyspark in databricks and though there were a few syntax changes, the tutorial made me understand the concept properly.



NNK 31 OCT 2020 [REPLY](#)

Thanks, Sneha for your comments, and glad you like the articles. If you find any syntax changes in Databricks please do comment, others might get benefit from your findings.

Leave a Reply

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/>)

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 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/>)

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/>)

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/>)

(https: (https:

//www. //www.

(https: facebo linkedi (https:

//twitter ok.co n.com/ //github

[r.com/](#) [m/spar](#) [in/n-](#) [b.com/](#)

sparkb kbyex nk- spark-

yexam ample b860a exam

ples) s/) 8193/) les/)

