How to Calculate Percentage of Total by Group Using PySpark's groupBy Function

Authored by

stats writer

February 6, 2026

The groupBy function in PySpark allows for the grouping of data by a specific column or set of columns in a dataset. This function can be used to calculate the percentage of total for each group in the dataset. By specifying the groupBy column, the data can be grouped together, and then by using the sum or count function, the total for each group can be calculated. Finally, by dividing the total for each group by the overall total, the percentage of total for each group can be obtained. This allows for a better understanding and analysis of the dataset, as it provides insights into the relative contribution of each group to the overall dataset.

PySpark: Calculate Percentage of Total with groupBy

You can use the following syntax to calculate the percentage of total rows that each group represents in a PySpark DataFrame:

```
#calculate total rows in DataFrame
n = df.count()

#calculate percent of total rows for each team
df.groupBy('team').count().withColumn('team_percent',
(F.col('count')/n)*100).show()
```

This particular example counts the number of occurrences for each unique value in the team column and then calculates the percentage of total rows that each unique value represents.

The following example shows how to use this syntax in practice.

Example: Calculate Percentage of Total with groupBy in PySpark

Suppose we have the following PySpark DataFrame that contains information about the points scored by various basketball players:

```
from pyspark.sql import SparkSession
spark = SparkSession.builder.getOrCreate()

#define data
data = ,
,
,
,
,
,
,
]

#define column names
columns =

#create dataframe using data and column names
df = spark.createDataFrame(data, columns)

#view dataframe
```

```
df.show()

+----+--------+------+
|team|position|points|
+----+--------+------+
| A| Guard| 11|
| A| Guard| 8|
| A| Forward| 22|
| A| Forward| 22|
| B| Guard| 14|
| B| Guard| 14|
| B| Forward| 13|
| C| Forward| 7|
+----+--------+------+
```

We can use the following syntax to count the number of occurrences if each unique value in the team column and then calculate the percentage of total rows that each unique team value represents:

```
#calculate total rows in DataFrame
n = df.count()

#calculate percent of total rows for each team
df.groupBy('team').count().withColumn('team_percent',
```

```
(F.col('count')/n)*100).show()

+----+-----+------------+
|team|count|team_percent|
+----+-----+------------+
| A| 4| 50.0|
| B| 3| 37.5|
| C| 1| 12.5|
+----+-----+------------+
```

The team_percent column shows the percentage of total rows represented by each unique team.

For example, there are 8 total rows in the DataFrame.

From the team_percent column, we can see:

There are 4 occurrences of team A, which represents 4/8 = 50% of the total rows.There are 3 occurrences of team B, which represents 3/8 = 37.5% of the total rows.There is 1 occurrence of team C, which represents 1/8 = 12.5% of the total rows.

Note: You can find the complete documentation for the PySpark groupBy function .

The following tutorials explain how to perform other common tasks in PySpark: