How to Select Distinct Rows in PySpark: A Step-by-Step Guide

Authored by

stats writer

February 9, 2026

In PySpark, selecting distinct rows from a dataset allows for efficient data manipulation and analysis. This can be achieved by using the "distinct" function, which eliminates duplicate rows and returns only unique values. This function can be applied to any PySpark dataframe or SQL table. Examples of selecting distinct rows in PySpark include finding unique customer names in a sales dataset, or identifying distinct product categories in an inventory database. By using the "distinct" function, users can easily filter and analyze data without redundant information.

Select Distinct Rows in PySpark (With Examples)

You can use the following methods to select distinct rows in a PySpark DataFrame:

Method 1: Select Distinct Rows in DataFrame

#display distinct rows only
df.distinct().show()

Method 2: Select Distinct Values from Specific Column

#display distinct values from 'team' column only
df.select('team').distinct().show()

Method 3: Count Distinct Rows in DataFrame

#count number of distinct rows
df.distinct().count()

The following examples show how to use each of these

methods in practice with the following PySpark DataFrame:

```
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|
| B| Forward| 7|
+----+--------+------+
```

Example 1: Select Distinct Rows in DataFrame

We can use the following syntax to select the distinct rows in the DataFrame:

```
#display distinct rows only
df.distinct().show()
```

```
+----+--------+------+
|team|position|points|
+----+--------+------+
| A| Guard| 11|
| A| Guard| 8|
```

A	Forward	22
B	Guard	14
B	Forward	13
B	Forward	7
+----+--------+------+

Notice that each row in the resulting DataFrame is distinct.

Example 2: Select Distinct Values from Specific Column in DataFrame

We can use the following syntax to select the distinct values from the team column in the DataFrame:

```
#display distinct values from 'team' column only
df.select('team').distinct().show()
```

+----+
|team|
+----+
| A|
| B|
+----+

The output shows the two distinct values from the team column: A and B.

Example 3: Count Distinct Rows in DataFrame

We can use the following syntax to count the number of distinct rows in the DataFrame:

```
#count number of distinct rows
df.distinct().count()
```

```
6
```

The output tells us that there are 6 distinct rows in the entire DataFrame.

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