

How to Drop or Select Columns in PySpark DataFrames

Authored by
stats writer

February 7, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Drop or Select Columns in PySpark DataFrames*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=129606>

To exclude columns in PySpark, the user can use the "drop" function with the specified columns as parameters. This function will remove the selected columns from the dataframe and return a new dataframe. Alternatively, the "select" function can be used to select all columns except the ones specified. For example, if we have a dataframe with columns "A", "B", "C", and "D", and we want to exclude "C" and "D", we can use the code `df.drop("C", "D")` or `df.select("A", "B")`. This will exclude the specified columns and return a new dataframe with only columns "A" and "B". These functions provide a convenient way to exclude unwanted columns in PySpark.

Exclude Columns in PySpark (With Examples)

You can use the following methods to exclude specific columns in a PySpark DataFrame:

Method 1: Exclude One Column

```
#select all columns except 'points' column  
df_new = df.drop('points')
```

Method 2: Exclude Multiple Columns

```
#select all columns except 'conference' and 'points'  
columns  
df_new = df.drop('conference', 'points')
```

The following examples show how to use each method 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|conference|points|assists|
```

```
+----+-----+-----+-----+
```

```
| A| East| 11| 4|
```

```
| A| East| 8| 9|
```

```
| A| East| 10| 3|
```

```
| B| West| 6| 12|
```

```
| B| West| 6| 4|
```

```
| C| East| 5| 2|
```

```
+----+-----+-----+-----+
```

Example 1: Exclude One Column in PySpark

We can use the following syntax to select all columns in the DataFrame, excluding the points column:

```
#select all columns except 'points' column
```

```
df_new = df.drop('points')
```

```
#view new DataFrame
```

```
df_new.show()
```

```
+----+-----+-----+
```

```
|team|conference|assists|
```

```
+----+-----+-----+
```

```
| A| East| 4|
```

```
| A| East| 9|
```

```
| A| East| 3|
```

```
| B| West| 12|
```

```
| B| West| 4|
```

```
| C| East| 2|
```

```
+----+-----+-----+
```

Notice that all columns in the DataFrame are selected except for the points column.

Example 2: Exclude Multiple Columns in PySpark

We can use the following syntax to select all columns in the DataFrame, excluding the conference and points column:

```
#select all columns except 'conference' and 'points' columns
```

```
df_new = df.drop('conference', 'points')
```

```
#view new DataFrame
```

```
df_new.show()
```

```
+----+-----+-----+
```

```
|team|assists|
```

```
+----+-----+-----+
```

```
| A| 4|
```

```
| A| 9|
```

```
| A| 3|
```

```
| B| 12|
```

```
| B| 4|
```

| C| 2|

+----+-----++

Notice that all columns in the DataFrame are selected except for the conference and points columns.

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

ARABPSYCHOLOGY.COM