# How can I use the PySpark `date_format()` function to convert a date to a string format?

Authored by

**stats writer**

June 24, 2024

The PySpark `date_format()` function allows for the conversion of a date to a string format. This function takes in a date column and a string format pattern as inputs, and outputs a string representation of the date according to the specified format. This can be useful for data analysis and manipulation, as well as for displaying dates in a more readable format. The `date_format()` function is a powerful tool for working with dates in PySpark and can greatly enhance the efficiency of data processing tasks.

The `date_format()` function in PySpark is a powerful tool for transforming, formatting date columns and converting date to string within a DataFrame. This function allows you to convert date and timestamp columns into a specified string format, providing flexibility for various date manipulation tasks.

Leveraging `date_format()`, you can customize the appearance of dates to match different formats required for reporting, visualization, or further data processing. It operates similarly to date formatting functions in SQL and other programming languages, making it a familiar and essential function for data engineers and analysts working with date and time data in PySpark.

## PySpark date_format() Usage

To use `date_format()` in PySpark, first import the function from `pyspark.sql.functions`. This function is primarily used to format Date to String format. This function supports all Java Date formats specified in DateTimeFormatter.

**Following are the Syntax and Example of date_format() Function**:

```
# Syntax:
date_format(column,format)
```

Example:

```
# Import
from pyspark.sql.functions import *
```

```
# Create DataFrame
df=spark.createDataFrame([],)

# Using date_format()
df.select(current_date().alias("current_date"),
date_format(current_timestamp(),"yyyy MM dd").alias("yyyy MM dd"),
date_format(current_timestamp(),"MM/dd/yyyy hh:mm").alias("MM/dd/yyyy"),
```

date_format(current_timestamp(),"yyyy MMM dd").alias("yyyy MMMM dd"),
date_format(current_timestamp(),"yyyy MMMM dd E").alias("yyyy MMMM dd E")
).show()

**Explanation:**

```
# Output:
+------------+----------+---------------+------------+---------------+
|current_date|yyyy MM dd| MM/dd/yyyy|yyyy MMMM dd| yyyy MMMM dd E|
+------------+----------+---------------+------------+---------------+
| 2024-05-28|2023 04 28|05/28/2024 02:18| 2024 May 28|2024 May 28 Tue|
+------------+----------+---------------+------------+---------------+
```

## Using date_format() with SQL Query

Alternatively, using the same functions, you can convert Data to String with SQL.

```
# SQL
spark.sql("select current_date() as current_date, "+
"date_format(current_timestamp(),'yyyy MM dd') as yyyy_MM_dd, "+
"date_format(current_timestamp(),'MM/dd/yyyy hh:mm') as MM_dd_yyyy, "+
"date_format(current_timestamp(),'yyyy MMM dd') as yyyy_MMMM_dd, "+
"date_format(current_timestamp(),'yyyy MMMM dd E') as yyyy_MMMM_dd_E").show()
```

## Complete Example of Convert Date to String

```
from pyspark.sql import SparkSession
```

# Create SparkSession
spark = SparkSession.builder
.appName('SparkByExamples.com')
.getOrCreate()

from pyspark.sql.functions import *

df=spark.createDataFrame([],)
df.select(current_date().alias("current_date"),
date_format(current_date(),"yyyy MM dd").alias("yyyy MM dd"),

```
date_format(current_timestamp(),"MM/dd/yyyy hh:mm").alias("MM/dd/yyyy"),
date_format(current_timestamp(),"yyyy MMM dd").alias("yyyy MMMM dd"),
date_format(current_timestamp(),"yyyy MMMM dd E").alias("yyyy MMMM dd E")
).show()

#SQL

spark.sql("select current_date() as current_date, "+
"date_format(current_timestamp(),'yyyy MM dd') as yyyy_MM_dd, "+
"date_format(current_timestamp(),'MM/dd/yyyy hh:mm') as MM_dd_yyyy, "+
"date_format(current_timestamp(),'yyyy MMM dd') as yyyy_MMMM_dd, "+
"date_format(current_timestamp(),'yyyy MMMM dd E') as yyyy_MMMM_dd_E").show()
```

## Conclusion:

In this article, you have learned how to convert Date to String format using the Date function `date_format()`.

## Related Articles: