MapType: Pyspark MapType is used to represent map key-value pair similar to python Dictionary.

from pyspark.sql import \*  
from pyspark.sql.functions import \*  
from pyspark.sql.types import \*  
  
spark = SparkSession.builder.appName("MapType\_Column").getOrCreate()  
  
mydata = [('Harsha',{'hair':'black','eye':'brown'}),  
 ('Mokshit',{'hair':'black','eye':'blue'})]  
  
myschema = StructType([StructField('name',StringType()),  
 StructField('properties',MapType(StringType(),StringType()))  
 ])  
  
df = spark.createDataFrame(mydata,myschema)  
  
df.show(truncate=False)  
df.printSchema()

**Output:**

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

|name |properties |

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

|Harsha |{eye -> brown, hair -> black}|

|Mokshit|{eye -> blue, hair -> black} |

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

root

|-- name: string (nullable = true)

|-- properties: map (nullable = true)

| |-- key: string

| |-- value: string (valueContainsNull = true)

Access MapType Elements

from pyspark.sql import \*  
from pyspark.sql.functions import \*  
from pyspark.sql.types import \*  
  
spark = SparkSession.builder.appName("MapType\_Column").getOrCreate()  
  
mydata = [('Harsha',{'hair':'black','eye':'brown'}),  
 ('Mokshit',{'hair':'black','eye':'blue'})]  
  
myschema = StructType([StructField('name',StringType()),  
 StructField('properties',MapType(StringType(),StringType()))])  
  
df = spark.createDataFrame(mydata,myschema)  
  
df1 = df.withColumn('hair',df.properties['hair'])  
  
df2 = df1.withColumn('eye',df.properties['eye'])  
  
df2.show(truncate=False)

**Output:**

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

|name |properties |hair |eye |

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

|Harsha |{eye -> brown, hair -> black}|black|brown|

|Mokshit|{eye -> blue, hair -> black} |black|blue |

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