**Session 7**

**Assignment 7.6**

Student Name: Karthik K

Course: Big Data Hadoop & Spark Training

Problem Statement

Using udfs on dataframe

1. Change firstname, lastname columns into

***Mr.first\_two\_letters\_of\_firstname<space>lastname***

for example - michael, phelps becomes Mr.mi phelps

2. Add a new column called ranking using udfs on dataframe, where:

***gold medalist, with age >= 32 are ranked as pro gold medalists, with age <= 31 are ranked amateur silver medalist, with age >= 32 are ranked as expert silver medalists, with age <= 31 are ranked rookie***

Task – 1 - Change firstname, lastname columns

***Mr.first\_two\_letters\_of\_firstname<space>lastname***

***For example - michael, phelps becomes Mr.mi phelps***

Please see the codes used below,

1. ***val SportsData = sc.textFile("/home/acadgild/hadoop/Sports\_data.txt")***
2. ***val schemaString =***

***"firstname:string,lastname:string,sports:string,medal\_type:string,age:string,year:string,count ry:string"***

1. ***val schema = StructType(schemaString.split(",").map(x => StructField(x.split(":")(0),if(x.split(":")(1).equals("string"))StringType else IntegerType, true)))***
2. ***val rowRDD = SportsData.map(\_.split(",")).map(r => Row(r(0), r(1), r(2), r(3), r(4), r(5), r(6)))***
3. ***val SportsDataDF = spark.createDataFrame(rowRDD, schema)***
4. ***SportsDataDF.createOrReplaceTempView("Sports\_Data")***
5. ***val Name = udf((firstname:String, lastname:String)=>"Mr. ".concat(firstname.substring(0,2)).concat(" ")concat(lastname))***
6. ***spark.udf.register("Full\_Name", Name)***
7. ***val fname = spark.sql("SELECT Full\_Name(firstname, lastname) FROM SportsData").show()***

We will proceed with the tasks,

In order to proceed we need to import some dependencies as shown below,



***import org.apache.spark.sql.Row;***

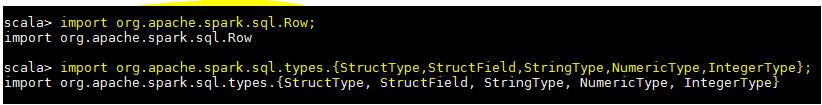
******

***import***

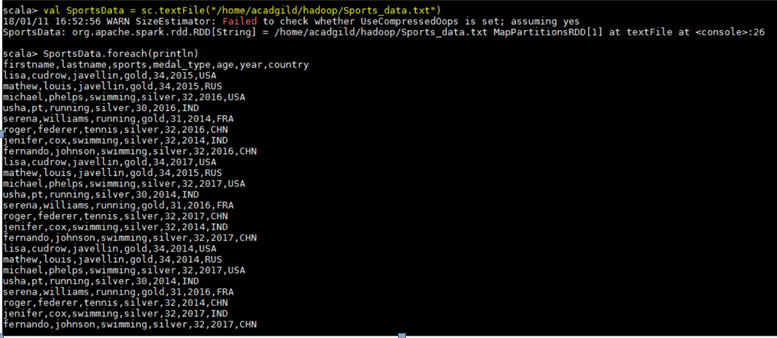
***org.apache.spark.sql.types.{StructType,StructField,StringType,NumericType,IntegerType};***

******

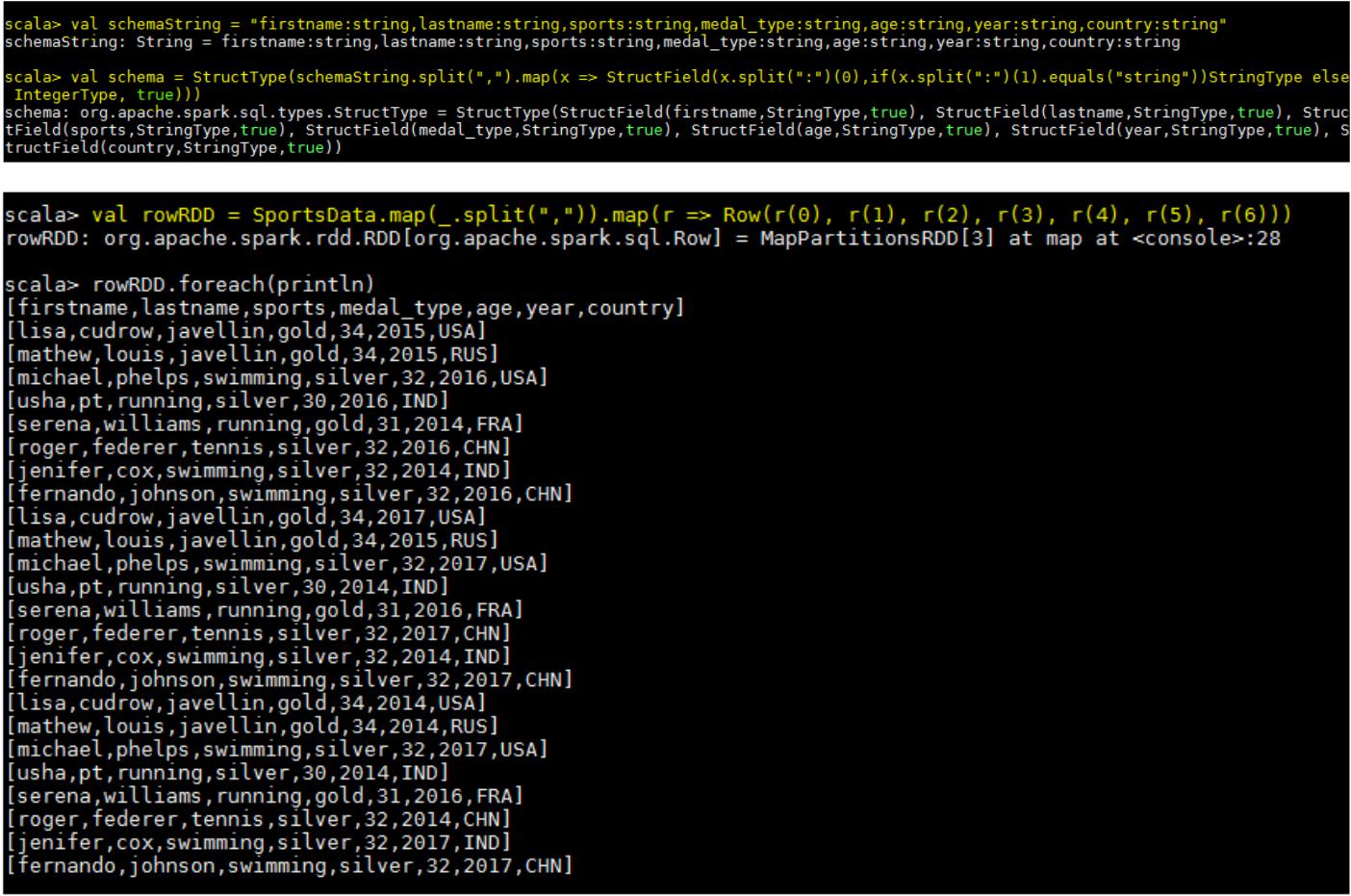
***import org.apache.spark.sql.functions.udf***

******

**Step -1** –we are creating a RDD from Input DataSet,

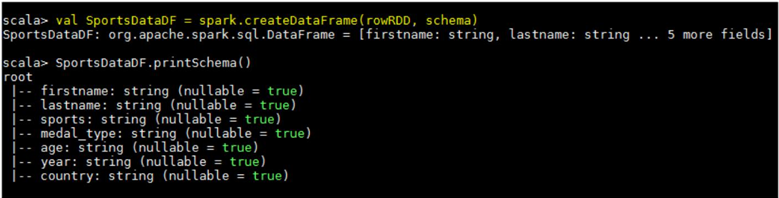


**Step -2** –we are defining a schema since it is a text file and splitting the input file using the delimiters andextracting the rows from it.



We have created the **dataframe** by passing the RDD which reads the file and schema to spark session object-

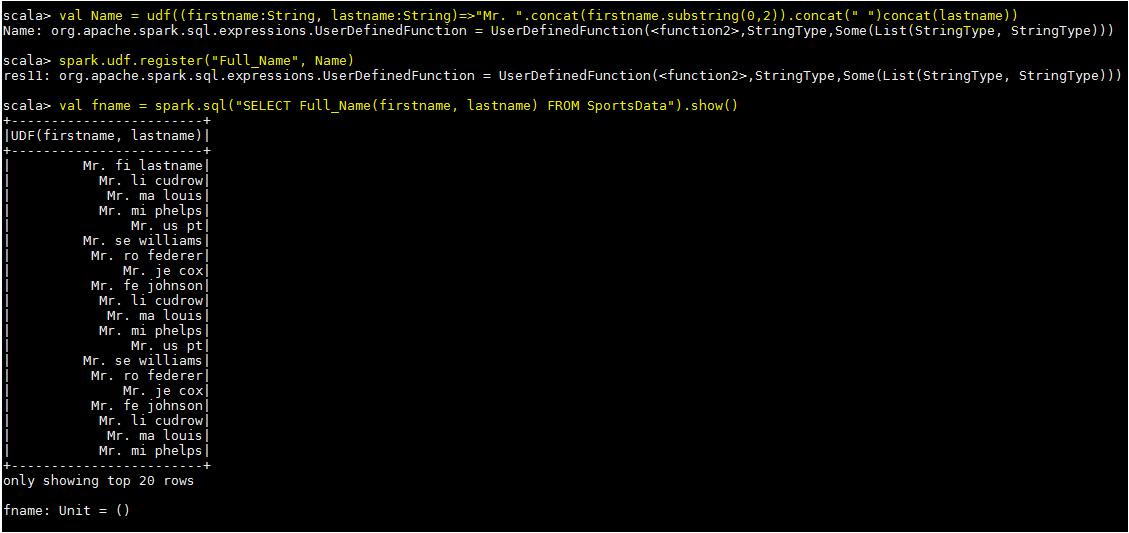
The schema of the created **Dataframe** can be seen below.



**Step – 3** - Here we are defining the UDF which will take 2 strings (columns) as input and will concatenatethem with Mr. appended in it and now we need to register the UDF. Here we doing the same and giving it an alias as **Full\_Name.**

Finally we can apply this UDF on the columns to give the required result-

Expected Output



Task – 2 - Add a new column called ranking using **udfs** on **dataframe**, where:

1. gold medalist, with age >= 32 are ranked as pro
2. gold medalists, with age <= 31 are ranked amateur
3. silver medalist, with age >= 32 are ranked as expert
4. silver medalists, with age <= 31 are ranked rookie

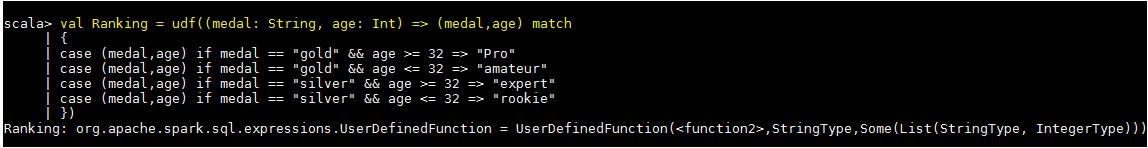
The UDF below, UDF that we have used to define the new column

***val Ranking = udf((medal: String, age: Int) => (medal,age) match***

***{***

***case (medal,age) if medal == "gold" && age >= 32 => "Pro" case (medal,age) if medal == "gold" && age <= 32 => "amateur" case (medal,age) if medal == "silver" && age >= 32 => "expert" case (medal,age) if medal == "silver" && age <= 32 => "rookie" })***

Here we are classifying each player based on age and the medal he has got,



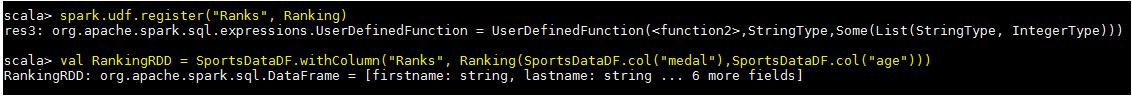
Below code shows the registering of UDF and command to add a new column,



***spark.udf.register("Ranks", Ranking)***

******

***val RankingRDD = SportsDataDF.withColumn("Ranks", Ranking(SportsDataDF.col("medal"),SportsDataDF.col("age")))***

******

And the desired result is shown in the below screen shot,

Expected Output

