

Session 21:

SPARK SQL 2

Assignment 1

Task 1

Using spark-sql, Find:

1. What are the total number of gold medal winners every year
2. How many silver medals have been won by USA in each sport

Solution: // Create a case class globally.

```
//Inferring the Schema Using Reflection.Automatically converting an RDD
containing case classes to a DataFrame.
// The case class defines the schema of the table. The names of the arguments
to the case class are read using reflection
// and become the names of the columns.
```

SOLUTION:

package SQL

//import org.apache.spark.sql.SparkSession

import org.apache.spark.sql.SparkSession

object Sports_Winner {

//Create a case class globally to be used inside the main method

**//Inferring the Schema Using Reflection.Automatically converting an RDD containing case classes
to a DataFrame.**

**// The case class defines the schema of the table. The names of the arguments to the case class
are read using reflection**

// and become the names of the columns.

// Main method - The execution entry point for the program

```
case class
Sports(firstname:String,lastname:String,sports:String,medal_type:String,age:Int,year:Long,country
id:String)
```

```
def main(args: Array[String]): Unit = {

  val spark = SparkSession

    .builder()

    .master("local")

    .appName("Sports_data")

    .config("spark.some.config.option", "some-value")

    .getOrCreate()

  println("spark session object created")

  spark.sparkContext.setLogLevel("WARN")


  //println("spark session object created")

  import spark.implicits._

  val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/Sports_data.txt")

  val header = data.first()

  val header1 = data.filter(row => row != header)

  val sports_data = header1.map(x => x.split(",")).map(x => Sports(x(0), x(1), x(2), x(3), x(4).toInt,
x(5).toLong, x(6))).toDF()

  sports_data.show()

  //Converting the above created schema into an SQL view named sport

  sports_data.createOrReplaceTempView("sport")
```

// 1. What are the total number of gold medal winners every year?

```
// Selecting year & counting the occurrence of each year by filtering medal_type condition as gold.

// grouping by year & ordering the result based upon the year.

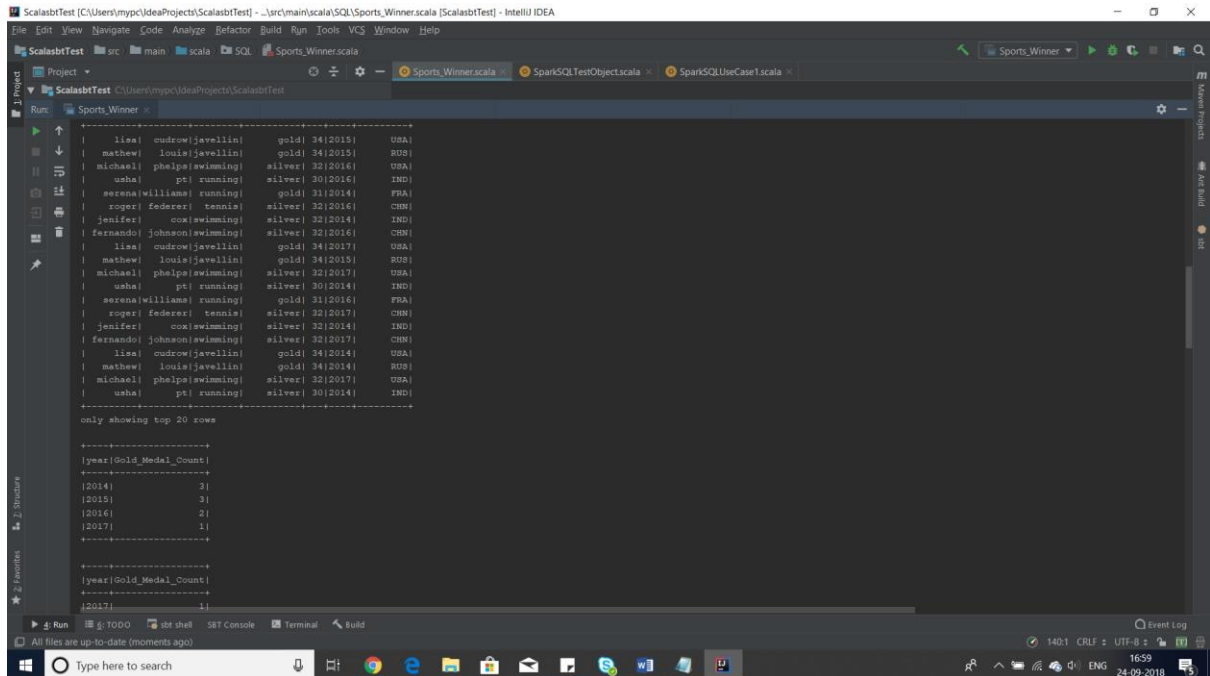
val a1=spark.sql("Select year, count(year)as Gold_Medal_Count from sport where medal_type
='gold' group by year order by year ASC")

val a2=spark.sql("Select year, count(year)as Gold_Medal_Count from sport where medal_type
='gold' group by year order by Gold_Medal_Count ASC")

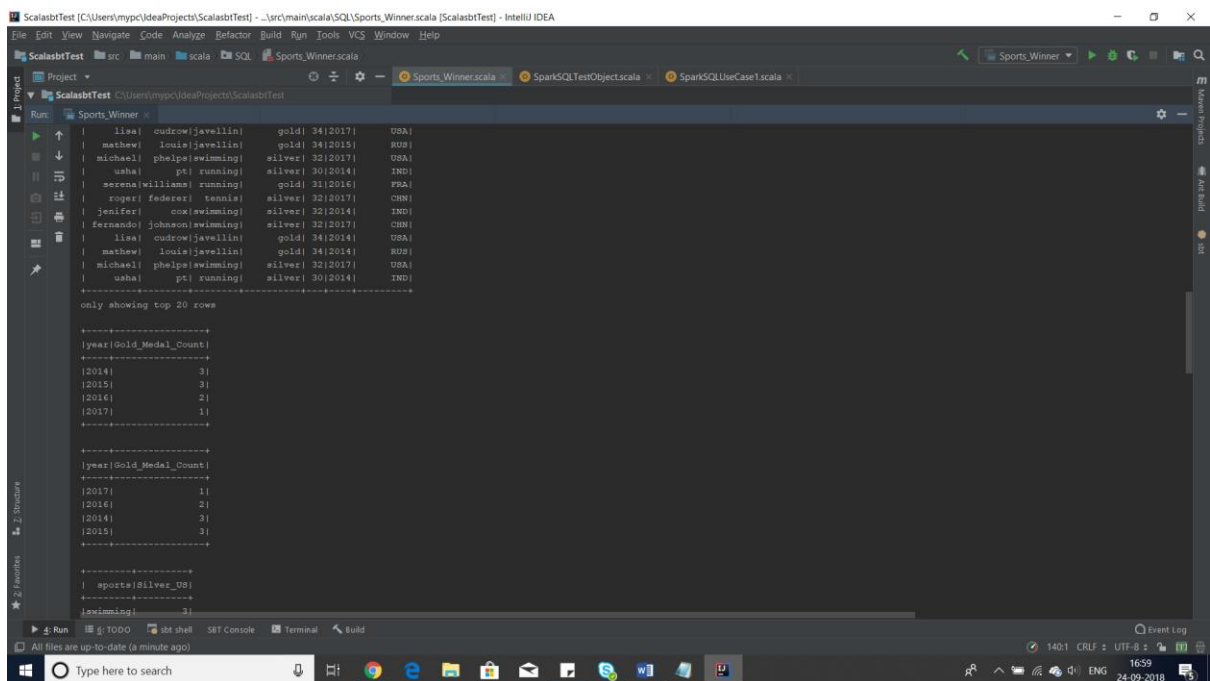
a1.show()
```

a2.show()

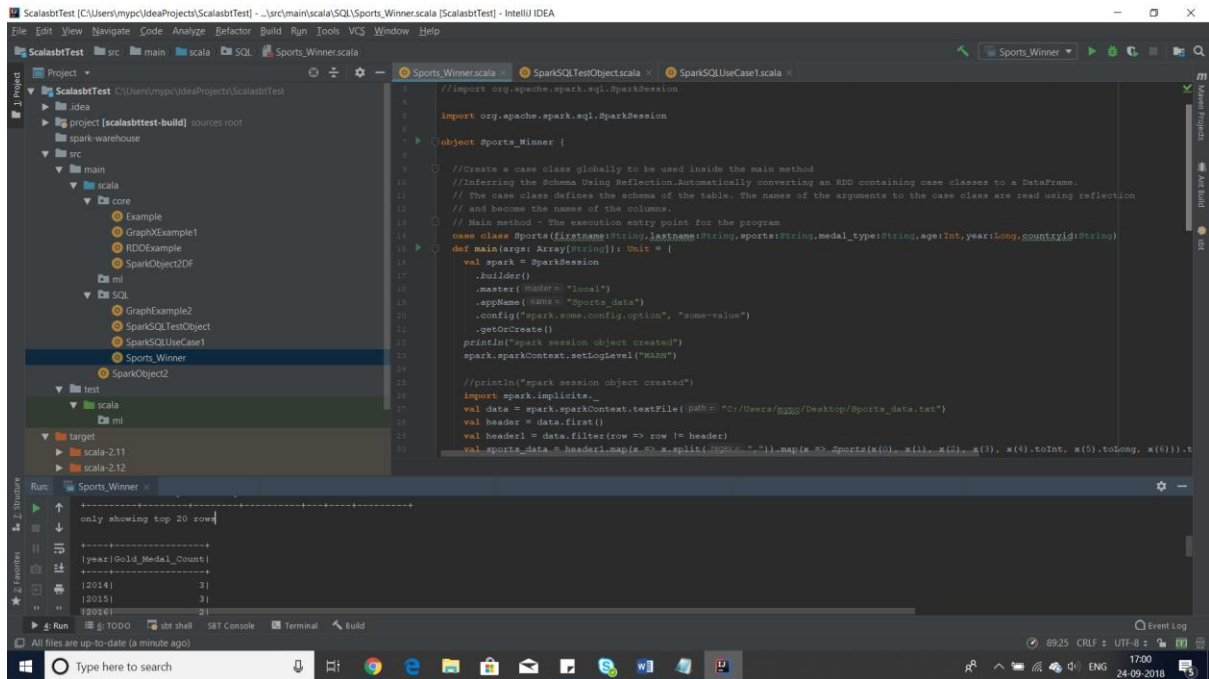
Screenshot:



```
scala> a2.show()
+-----+
| name      | sport      | medal | year | country |
+-----+
| lisa      | cudrow|javelin| gold | 34|2015 | USA    |
| mathew    | louie|javelin| gold | 34|2015 | ROU    |
| michael   | phelps|swimming| silver| 32|2016 | USA    |
| usha      | pt| running| silver| 30|2016 | IND    |
| serena|williams| running| gold  | 31|2016 | FRA    |
| roger     | federer| tennis | silver| 32|2016 | CHN    |
| jenifer   | cox|swimming| silver| 32|2016 | IND    |
| fernando  | johnson|swimming| silver| 32|2016 | CHN    |
| lisa      | cudrow|javelin| gold  | 34|2017 | USA    |
| mathew    | louie|javelin| gold  | 34|2015 | ROU    |
| michael   | phelps|swimming| silver| 32|2017 | USA    |
| usha      | pt| running| silver| 30|2014 | IND    |
| serena|williams| running| gold  | 31|2016 | FRA    |
| roger     | federer| tennis | silver| 32|2017 | CHN    |
| jenifer   | cox|swimming| silver| 32|2014 | IND    |
| fernando  | johnson|swimming| silver| 32|2017 | CHN    |
| lisa      | cudrow|javelin| gold  | 34|2014 | USA    |
| mathew    | louie|javelin| gold  | 34|2014 | ROU    |
| michael   | phelps|swimming| silver| 32|2017 | USA    |
| usha      | pt| running| silver| 30|2014 | IND    |
+-----+
only showing top 20 rows
+-----+
| year|Gold_Medal_Count|
+-----+
|2014| 3|
|2015| 3|
|2016| 2|
|2017| 1|
+-----+
| year|Gold_Medal_Count|
+-----+
|2017| 1|
+-----+
```



```
scala> a2.show()
+-----+
| name      | sport      | medal | year | country |
+-----+
| lisa      | cudrow|javelin| gold | 34|2017 | USA    |
| mathew    | louie|javelin| gold | 34|2015 | ROU    |
| michael   | phelps|swimming| silver| 32|2017 | USA    |
| usha      | pt| running| silver| 30|2014 | IND    |
| serena|williams| running| gold  | 31|2016 | FRA    |
| roger     | federer| tennis | silver| 32|2017 | CHN    |
| jenifer   | cox|swimming| silver| 32|2014 | IND    |
| fernando  | johnson|swimming| silver| 32|2017 | CHN    |
| lisa      | cudrow|javelin| gold  | 34|2014 | USA    |
| mathew    | louie|javelin| gold  | 34|2014 | ROU    |
| michael   | phelps|swimming| silver| 32|2017 | USA    |
| usha      | pt| running| silver| 30|2014 | IND    |
+-----+
only showing top 20 rows
+-----+
| year|Gold_Medal_Count|
+-----+
|2014| 3|
|2015| 3|
|2016| 2|
|2017| 1|
+-----+
| year|Gold_Medal_Count|
+-----+
|2017| 1|
|2016| 2|
|2014| 3|
|2015| 3|
+-----+
| sport|Silver_US|
+-----+
|swimming| 3|
+-----+
```



//2. How many silver medals have been won by USA in each sport?

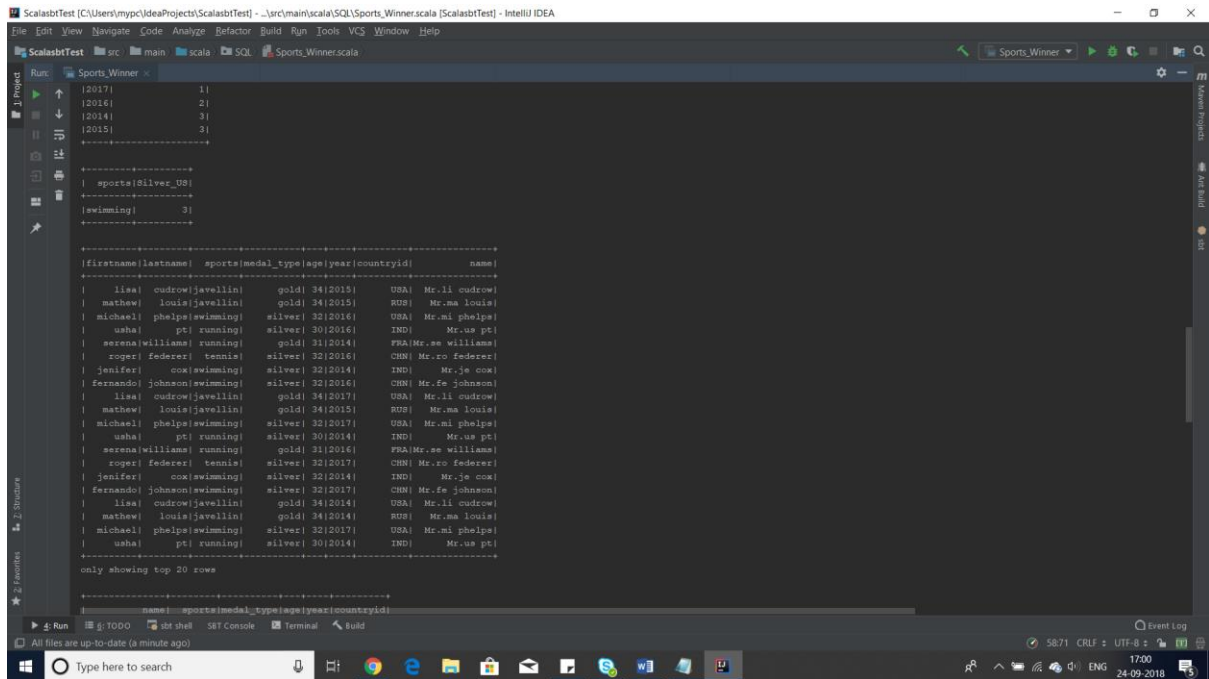
// Selecting sports & count of sports as Silver_US from sports view. Provided the filter as medal_type = 'silver' &

// coutry as USA, grouping by count & ordering the result based upon count of medals won.

val a3 = spark.sql("select sports, count(sports) as Silver_US from sport where (medal_type = 'silver' AND countryid = 'USA') group by sports order by Silver_US ASC")

a3.show()

}}



Task 2

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

Solution:

/*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*/

```

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

def udf_change_columns = udf((firstname:String,lastname:String)=>{
    val twochars_Firstname=firstname.substring(0,2)
    val name ="Mr."+twochars_Firstname+" "+lastname
    name})

val df
=sports_data.withColumn("name",udf_change_columns($"firstname", $"last
name"))

df.show()

val df1=df.select("name","sports","medal_type","age","year","countryid")
// val df2 = df.drop("firstname","lastname")
df1.show()


val udf_add_columns = udf((medal_type:String,age:Int)=>{
    val ranking= if (medal_type.equals("gold") && age >= 32 ) "pro"
    else if(medal_type.equals("gold") && age <= 31 ) "amateur"
    else if(medal_type.equals("silver") && age >= 32 ) "expert"
    else if(medal_type.equals("silver") && age <= 31 ) "rookie"
    else "NA"
    ranking
})

val added_column
=df1.withColumn("ranking",udf_add_columns($"medal_type", $"age"))
added_column.show()

```

```

val added_column1=
sports_data.withColumn("ranking",udf_add_columns($"medal_type",$"age"
))

added_column1.show()

}}

```

Screenshot:

The screenshot shows the IntelliJ IDEA interface with a Scala test file named `Sports_Winner.scala`. The code defines a SQL query that filters for silver medalists in swimming and displays the top 20 rows. The results are shown in a table format within the IDE's output window.

```

Run: Sports_Winner
| sports|Silver_US|
|swimming| 3|
=====
|firstname|lastname| sports|medal_type|age|year|countryid| name|
|-----|-----|-----|-----|-----|-----|-----|-----|
| lisa| cudrow|javelin| gold| 34|2015| USA| Mr.l1 cudrow|
| mathew| louis|javelin| gold| 34|2015| RUB| Mr.ma louis|
| michael| phelps|swimming| silver| 32|2016| USA| Mr.mi phelps|
| usha| pt| running| silver| 30|2016| IND| Mr.us pt|
| serena|williams| running| gold| 31|2014| FRA|Mr.se williams|
| roger| federer| tennis| silver| 32|2016| CHN| Mr.ro federer|
| jennifer| cox|swimming| silver| 32|2014| IND| Mr.je cox|
| fernando| johnson|swimming| silver| 32|2016| CHN| Mr.fe johnson|
| lisa| cudrow|javelin| gold| 34|2017| USA| Mr.l1 cudrow|
| mathew| louis|javelin| gold| 34|2015| RUB| Mr.ma louis|
| michael| phelps|swimming| silver| 32|2017| USA| Mr.mi phelps|
| usha| pt| running| silver| 30|2014| IND| Mr.us pt|
| serena|williams| running| gold| 31|2016| FRA|Mr.se williams|
| roger| federer| tennis| silver| 32|2017| CHN| Mr.ro federer|
| jennifer| cox|swimming| silver| 32|2014| IND| Mr.je cox|
| fernando| johnson|swimming| silver| 32|2017| CHN| Mr.fe johnson|
| lisa| cudrow|javelin| gold| 34|2014| USA| Mr.l1 cudrow|
| mathew| louis|javelin| gold| 34|2014| RUB| Mr.ma louis|
| michael| phelps|swimming| silver| 32|2017| USA| Mr.mi phelps|
| usha| pt| running| silver| 30|2014| IND| Mr.us pt|
=====
only showing top 20 rows
=====
| name| sports|medal_type|age|year|countryid|
|-----|-----|-----|-----|-----|-----|
| Mr.l1 cudrow|javelin| gold| 34|2015| USA|
| Mr.ma louis|javelin| gold| 34|2015| RUB|
| Mr.mi phelps|swimming| silver| 32|2016| USA|
| Mr.us pt| running| silver| 30|2016| IND|
| Mr.se williams| running| gold| 31|2014| FRA|

```

```
ScalabTest [C:\Users\myc\IdeaProjects\ScalabTest] - ...src\main\scala\SQL\Sports_Winner.scala [ScalabTest] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
ScalabTest src main scala SQL Sports_Winner.scala
Run: Sports_Winner
| matthew| louis|javelin| gold| 34|2015| ROU| Mr.ma louis|
| michael| phelps|swimming| silver| 32|2017| USA| Mr.mi phelps|
| usha| pt| running| silver| 30|2014| IND| Mr.us pt|
| serena|williams| running| gold| 31|2016| FRA|Mr.se williams|
| roger| federer| tennis| silver| 32|2017| CHN| Mr.ro federer|
| jennifer| cox|swimming| silver| 32|2014| IND| Mr.je cox|
| fernando| johnson|swimming| silver| 32|2017| CHN| Mr.fe johnson|
| lisa| cudrow|javelin| gold| 34|2014| USA| Mr.li cudrow|
| matthew| louis|javelin| gold| 34|2014| ROU| Mr.ma louis|
| michael| phelps|swimming| silver| 32|2017| USA| Mr.mi phelps|
| usha| pt| running| silver| 30|2014| IND| Mr.us pt|
only showing top 20 rows
+-----+
| name| sports| medal_type| age| year| country| id|
+-----+
| Mr.li cudrow|javelin| gold| 34|2015| USA|
| Mr.ma louis|javelin| gold| 34|2015| ROU|
| Mr.mi phelps|swimming| silver| 32|2016| USA|
| Mr.us pt| running| silver| 30|2016| IND|
| Mr.se williams| running| gold| 31|2014| FRA|
| Mr.ro federer| tennis| silver| 32|2016| CHN|
| Mr.je cox|swimming| silver| 32|2014| IND|
| Mr.fe johnson|swimming| silver| 32|2016| CHN|
| Mr.li cudrow|javelin| gold| 34|2017| USA|
| Mr.ma louis|javelin| gold| 34|2015| ROU|
| Mr.mi phelps|swimming| silver| 32|2017| USA|
| Mr.us pt| running| silver| 30|2014| IND|
| Mr.se williams| running| gold| 31|2016| FRA|
| Mr.ro federer| tennis| silver| 32|2017| CHN|
| Mr.je cox|swimming| silver| 32|2014| IND|
| Mr.fe johnson|swimming| silver| 32|2017| CHN|
| Mr.li cudrow|javelin| gold| 34|2014| USA|
| Mr.ma louis|javelin| gold| 34|2014| ROU|
| Mr.mi phelps|swimming| silver| 32|2017| USA|
| Mr.us pt| running| silver| 30|2014| IND|
only showing top 20 rows
Type here to search
```

```
ScalabTest [C:\Users\myc\IdeaProjects\ScalabTest] - ...src\main\scala\SQL\Sports_Winner.scala [ScalabTest] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
ScalabTest src main scala SQL Sports_Winner.scala
Run: Sports_Winner
| Mr.us pt| running| silver| 30|2014| IND|
only showing top 20 rows
+-----+
| name| sports| medal_type| age| year| country| id| ranking|
+-----+
| Mr.li cudrow|javelin| gold| 34|2015| USA| pro|
| Mr.ma louis|javelin| gold| 34|2015| ROU| pro|
| Mr.mi phelps|swimming| silver| 32|2016| USA| expert|
| Mr.us pt| running| silver| 30|2016| IND| rookie|
| Mr.se williams| running| gold| 31|2014| FRA| amateur|
| Mr.ro federer| tennis| silver| 32|2016| CHN| expert|
| Mr.je cox|swimming| silver| 32|2014| IND| expert|
| Mr.fe johnson|swimming| silver| 32|2016| CHN| expert|
| Mr.li cudrow|javelin| gold| 34|2017| USA| pro|
| Mr.ma louis|javelin| gold| 34|2015| ROU| pro|
| Mr.mi phelps|swimming| silver| 32|2017| USA| expert|
| Mr.us pt| running| silver| 30|2014| IND| rookie|
| Mr.se williams| running| gold| 31|2016| FRA| amateur|
| Mr.ro federer| tennis| silver| 32|2017| CHN| expert|
| Mr.je cox|swimming| silver| 32|2014| IND| expert|
| Mr.fe johnson|swimming| silver| 32|2017| CHN| expert|
| Mr.li cudrow|javelin| gold| 34|2014| USA| pro|
| Mr.ma louis|javelin| gold| 34|2014| ROU| pro|
| Mr.mi phelps|swimming| silver| 32|2017| USA| expert|
| Mr.us pt| running| silver| 30|2014| IND| rookie|
only showing top 20 rows
+-----+
| firstname| lastname| sports| medal_type| age| year| country| id| ranking|
+-----+
| lisa| cudrow|javelin| gold| 34|2015| USA| pro|
| matthew| louis|javelin| gold| 34|2015| ROU| pro|
| michael| phelps|swimming| silver| 32|2016| USA| expert|
| usha| pt| running| silver| 30|2016| IND| rookie|
| serena|williams| running| gold| 31|2014| FRA| amateur|
| roger| federer| tennis| silver| 32|2016| CHN| expert|
| jennifer| cox|swimming| silver| 32|2014| IND| expert|
Type here to search
```



```
ScalabTest [C:\Users\myrc\IdeaProjects\ScalabTest] - ...src\main\scala\SQL\Sports_Winner.scala [ScalabTest] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
ScalabTest src main scala SQL Sports_Winner.scala
Run: Sports_Winner
| Mr.ro federer| tennis| silver| 32|2017| CHN| expert|
| Mr.je cox|swimming| silver| 32|2014| IND| expert|
| Mr.fe johnson|swimming| silver| 32|2017| CHN| expert|
| Mr.li cudrow|javelin| gold| 34|2014| USA| pro|
| Mr.ma louis|javelin| gold| 34|2014| RUS| pro|
| Mr.mi phelps|swimming| silver| 32|2017| USA| expert|
| Mr.us pt| running| silver| 30|2014| IND| rookie|
only showing top 20 rows
+-----+
|firstname|lastname| sports|medal_type|age|year|countryid|ranking|
+-----+
| lisa| cudrow|javelin| gold| 34|2015| USA| pro|
| matthew| louis|javelin| gold| 34|2015| RUS| pro|
| michael| phelps|swimming| silver| 32|2016| USA| expert|
| usha| pt| running| silver| 30|2016| IND| rookie|
| serena|williams| running| gold| 31|2014| FRA|amateur|
| roger| federer| tennis| silver| 32|2016| CHN| expert|
| jenifer| cox|swimming| silver| 32|2014| IND| expert|
| fernando| johnson|swimming| silver| 32|2014| CHN| expert|
| lisa| cudrow|javelin| gold| 34|2017| USA| pro|
| matthew| louis|javelin| gold| 34|2015| RUS| pro|
| michael| phelps|swimming| silver| 32|2017| USA| expert|
| usha| pt| running| silver| 30|2014| IND| rookie|
| serena|williams| running| gold| 31|2016| FRA|amateur|
| roger| federer| tennis| silver| 32|2017| CHN| expert|
| jenifer| cox|swimming| silver| 32|2014| IND| expert|
| fernando| johnson|swimming| silver| 32|2017| CHN| expert|
| lisa| cudrow|javelin| gold| 34|2014| USA| pro|
| matthew| louis|javelin| gold| 34|2014| RUS| pro|
| michael| phelps|swimming| silver| 32|2017| USA| expert|
| usha| pt| running| silver| 30|2014| IND| rookie|
only showing top 20 rows
Process finished with exit code 0
```

```
ScalabTest [C:\Users\myrc\IdeaProjects\ScalabTest] - ...src\main\scala\SQL\Sports_Winner.scala [ScalabTest] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
ScalabTest src main scala SQL Sports_Winner.scala
Project: Sports_Winner.scala, SparkSQLTestObject.scala, SparkSQLUseCase1.scala
ScalabTest CHUsers\myrc\IdeaProjects\ScalabTest
src
  main
    scala
      core
        Example
        GraphExample1
        RDDExample
        SparkObject2DF
      SQL
        GraphExample2
        SparkSQLTestObject
        SparkSQLUseCase1
        Sports_Winner
        SparkObject2
  test
    scala
    ml
target
  scala-2.11
  scala-2.12
  streams
  history
build.sbt
External Libraries
Scratches and Consoles
Run: Sports_Winner
| Mr.li cudrow|javelin| gold| 34|2014| USA| pro|
| Mr.ma louis|javelin| gold| 34|2014| RUS| pro|
only showing top 20 rows
// Group by USA, grouping by count & ordering the result based upon count of medals won.
val a3 = spark.sql("select sports, count(sports) as silver_US from sport where (medal_type = 'silver' AND countryid = 'US')")
a3.show()
// Using udfs on dataframe
1. Change first_name, last_name columns into
Mr.first_two_letters_of_firstname-spaces-lastname
for example - michael, phelps becomes Mr.mi phelps
import org.apache.spark.sql.functions.udf
def udf_change_columns (UserDefinedFunction) = udf((firstname:String, lastname:String) => {
  val twochars_firstname=firstname.substring(0,2)
  val name = "Mr."+twochars_firstname+" "+lastname
  name()
})
val df = sports_data.withColumn( @PARAM = "name", udf_change_columns($"firstname", $"lastname"))
df.show()
val df1=df.select( @H = "name", @H = "sports", "medal_type", "age", "year", "countryid")
// val df2 = df.drop("firstname", "lastname")
df1.show()
val udf_add_columns = udf((medal_type:String, age:Int) => {
  val ranking= if (medal_type.equals("gold") && age >= 32 ) "pro"
  else if(medal_type.equals("gold") && age <= 31 ) "amateur"
  else if(medal_type.equals("silver") && age >= 32 ) "expert"
  else if(medal_type.equals("silver") && age <= 31 ) "rookie"
  else "na"
  ranking
})
val added_column =df1.withColumn( @PARAM = "ranking", udf_add_columns($"medal_type", $"age"))
added_column.show()
val added_column1= sports_data.withColumn( @PARAM = "ranking", udf_add_columns($"medal_type", $"age"))
added_column1.show()
})
```

ScalaSBTest [C:\Users\myrc\IdeaProjects\ScalaSBTest] - ...src\main\scala\SQL\Sports_Winner.scala [ScalaSBTest] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

ScalaSBTest src main scala SQL Sports_Winner.scala

Project ScalaSBTest C:\Users\myrc\IdeaProjects\ScalaSBTest

src main scala SQL Sports_Winner.scala

```
1. Change first_name, last_name columns into  
Mr.first_two_letters_of_firstname+space+lastname  
for example "michael, philips" becomes Mr.mi philips//  
import org.apache.spark.sql.functions.udf  
def udf_change_columns (UserDefinedFunction) = udf((firstname:string,lastname:string)=>{  
  val twochars_firstname=firstname.substring(0,2)  
  val name ="Mr."+twochars_firstname+" "+lastname  
  name  
})  
val df =sports_data.withColumn( @Name="name",udf_change_columns($"firstname",$"lastname"))  
df.show()  
val df1=df.select( @Name="name", @Name="sports","medal_type","age","year","countryid")  
// val df2 = df.drop("firstname","lastname")  
df1.show()  
  
val udf_add_columns = udf((medal_type:string,age:int)=>{  
  val ranking= if (medal_type.equals("gold") && age >= 32 ) "pro"  
  else if(medal_type.equals("gold") && age <= 31 ) "amateur"  
  else if(medal_type.equals("silver") && age >= 32 ) "expert"  
  else if(medal_type.equals("silver") && age <= 31 ) "rookie"  
  else "na"  
  ranking  
})  
val added_column =df1.withColumn( @Name="ranking",udf_add_columns($"medal_type", $"age"))  
added_column.show()  
val added_column1= sports_data.withColumn( @Name="ranking",udf_add_columns($"medal_type",$"age"))  
added_column1.show()  
}
```

Run Sports_Winner

first_name	last_name	medal_type	age	year	countryid	ranking
Mr.li	oudrow javeillin	gold	34	2014	USA	pro
Mr.ma	louis javeillin	gold	34	2014	RUS	pro
Mr.mi	philips swimming	silver	32	2017	USA	expert
Mr.us	pt running	silver	30	2014	IND	rookie

only showing top 20 rows

All files are up-to-date (3 minutes ago)

Type here to search

ScalaSBTest [C:\Users\myrc\IdeaProjects\ScalaSBTest] - ...src\main\scala\SQL\Sports_Winner.scala [ScalaSBTest] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

ScalaSBTest src main scala SQL Sports_Winner.scala

Project ScalaSBTest C:\Users\myrc\IdeaProjects\ScalaSBTest

src main scala SQL Sports_Winner.scala

```
def udf_change_columns (UserDefinedFunction) = udf((firstname:string,lastname:string)=>{  
  val twochars_firstname=firstname.substring(0,2)  
  val name ="Mr."+twochars_firstname+" "+lastname  
  name  
})  
val df =sports_data.withColumn( @Name="name",udf_change_columns($"firstname",$"lastname"))  
df.show()  
val df1=df.select( @Name="name", @Name="sports","medal_type","age","year","countryid")  
// val df2 = df.drop("firstname","lastname")  
df1.show()  
  
val udf_add_columns = udf((medal_type:string,age:int)=>{  
  val ranking= if (medal_type.equals("gold") && age >= 32 ) "pro"  
  else if(medal_type.equals("gold") && age <= 31 ) "amateur"  
  else if(medal_type.equals("silver") && age >= 32 ) "expert"  
  else if(medal_type.equals("silver") && age <= 31 ) "rookie"  
  else "na"  
  ranking  
})  
val added_column =df1.withColumn( @Name="ranking",udf_add_columns($"medal_type", $"age"))  
added_column.show()  
val added_column1= sports_data.withColumn( @Name="ranking",udf_add_columns($"medal_type",$"age"))  
added_column1.show()  
}
```

Run Sports_Winner

first_name	last_name	sports	medal_type	age	year	countryid	ranking
liia	oudrow javeillin		gold	34	2015	USA	pro
mathew	louis javeillin		gold	34	2015	RUS	pro
michael	philips swimming		silver	32	2016	USA	expert
saha	pt running		silver	30	2016	IND	rookie
serena	williams	tennis	gold	31	2014	FRA	amateur
serena	williams	tennis	silver	32	2016	CHN	expert

only showing top 20 rows

All files are up-to-date (4 minutes ago)

Type here to search