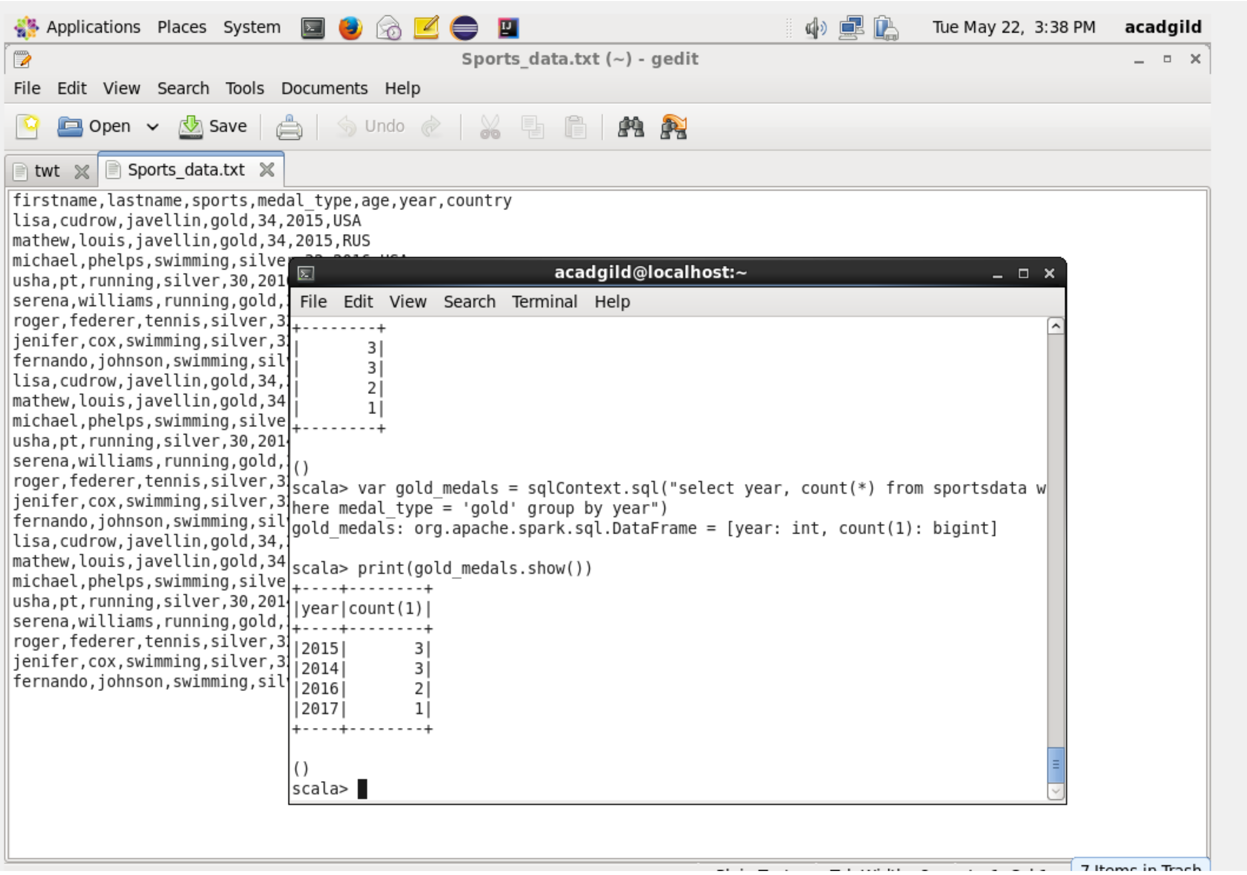
TASK1

1) Spark-shell –packages com.databricks:spark-csv_2.11:1.3.0
2) Val sqlContext = new org.apache.spark.sql.SQLContext(sc)
3) Val sports_data = sqlContext.read.format(“com.databricks.spark.csv”).option(“header”, “true”).option(“inferSchema”, “true”).load(“sports_data.txt”)
4) Sports_data.registerTempTable(“sportsdata”)
5) Sports_data.printSchema()
6) Var gold_medals = sqlContext.sql(“select year, count(*) from sportsdata where medal_type = ‘gold’ group by year”)



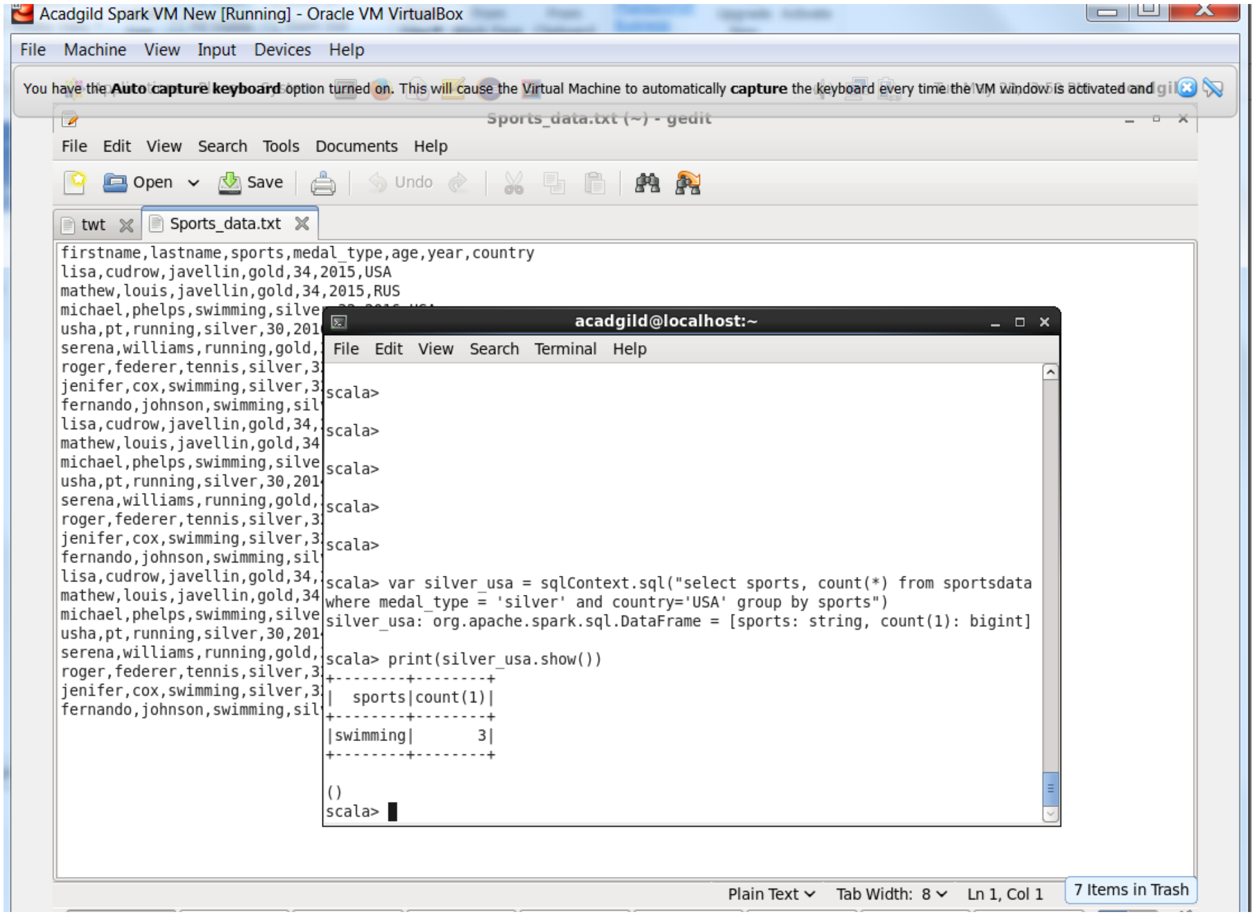
```
firstname,lastname,sports,medal_type,age,year,country
lisa,cudrow,javellin,gold,34,2015,USA
mathew,louis,javellin,gold,34,2015,RUS
michael,phelps,swimming,silve
usha,pt,running,silver,30,201
serena,williams,running,gold,
roger,federer,tennis,silver,3
jenifer,cox,swimming,silver,3
fernando,johnson,swimming,sil
lisa,cudrow,javellin,gold,34,
mathew,louis,javellin,gold,34
michael,phelps,swimming,silve
usha,pt,running,silver,30,201
serena,williams,running,gold,
roger,federer,tennis,silver,3
jenifer,cox,swimming,silver,3
fernando,johnson,swimming,sil
lisa,cudrow,javellin,gold,34,
mathew,louis,javellin,gold,34
michael,phelps,swimming,silve
usha,pt,running,silver,30,201
serena,williams,running,gold,
roger,federer,tennis,silver,3
jenifer,cox,swimming,silver,3
fernando,johnson,swimming,sil
```

```
+--------+
|       3|
|       3|
|       2|
|       1|
+--------+


()
scala> var gold_medals = sqlContext.sql("select year, count(*) from sportsdata w
here medal_type = 'gold' group by year")
gold_medals: org.apache.spark.sql.DataFrame = [year: int, count(1): bigint]

scala> print(gold_medals.show())
+----+--------+
|year|count(1)|
+----+--------+
|2015|       3|
|2014|       3|
|2016|       2|
|2017|       1|
+----+--------+


()
scala> 
```

How many silver medals have been won by the USA in each sport?

Var silver_usa = sqlContext.sql(“select sports, count(*) from sportsdata where medal_type = ‘silver’ and country = ‘USA’ group by sports)

Print(silver_usa.show())

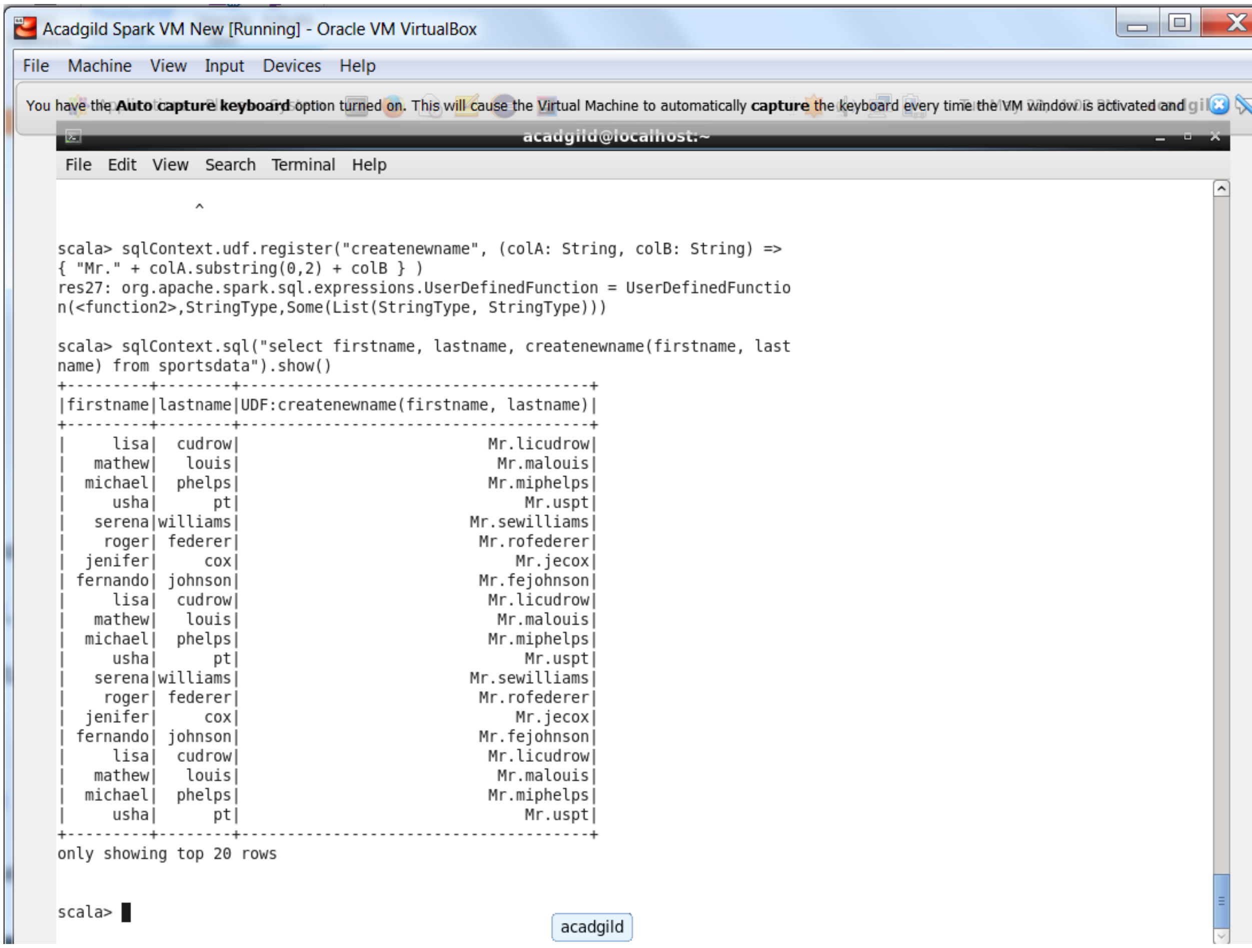

Acadgild Spark VM New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
You have the Auto capture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and
Sports_data.txt (~) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
twt Sports_data.txt
acadgild@localhost:~
File Edit View Search Terminal Help
Plain Text Tab Width: 8 Ln 1, Col 1
7 Items in Trash

```
firstname,lastname,sports,medal_type,age,year,country
lisa,cudrow,javellin,gold,34,2015,USA
mathew,louis,javellin,gold,34,2015,RUS
michael,phelps,swimming,silve
usha,pt,running,silver,30,201
serena,williams,running,gold,
roger,federer,tennis,silver,3
jenifer,cox,swimming,silver,3
fernando,johnson,swimming,sil
lisa,cudrow,javellin,gold,34,
mathew,louis,javellin,gold,34
michael,phelps,swimming,silve
usha,pt,running,silver,30,201
serena,williams,running,gold,
roger,federer,tennis,silver,3
jenifer,cox,swimming,silver,3
fernando,johnson,swimming,sil
lisa,cudrow,javellin,gold,34,
mathew,louis,javellin,gold,34
michael,phelps,swimming,silve
usha,pt,running,silver,30,201
serena,williams,running,gold,
roger,federer,tennis,silver,3
jenifer,cox,swimming,silver,3
fernando,johnson,swimming,sil
```

```
scala>

scala>

scala>

scala>

scala>

scala> var silver_usa = sqlContext.sql("select sports, count(*) from sportsdata
where medal_type = 'silver' and country='USA' group by sports")
silver_usa: org.apache.spark.sql.DataFrame = [sports: string, count(1): bigint]

scala> print(silver_usa.show())
+--------+--------+
|  sports|count(1)|
+--------+--------+
|swimming|       3|
+--------+--------+


()
scala> 
```

TASK 2

1) sqlContext.udf.register(“createnewname”, (colA: String, colB: String) { “Mr.” + colA.substring(0,2) + colB })
2) sqlContext.sql(“select firstname, lastname, createnewname(firstname, lastname) from sportsdata”).show()


Acadgild Spark VM New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
acadgild@localhost:~
File Edit View Search Terminal Help
acadgild

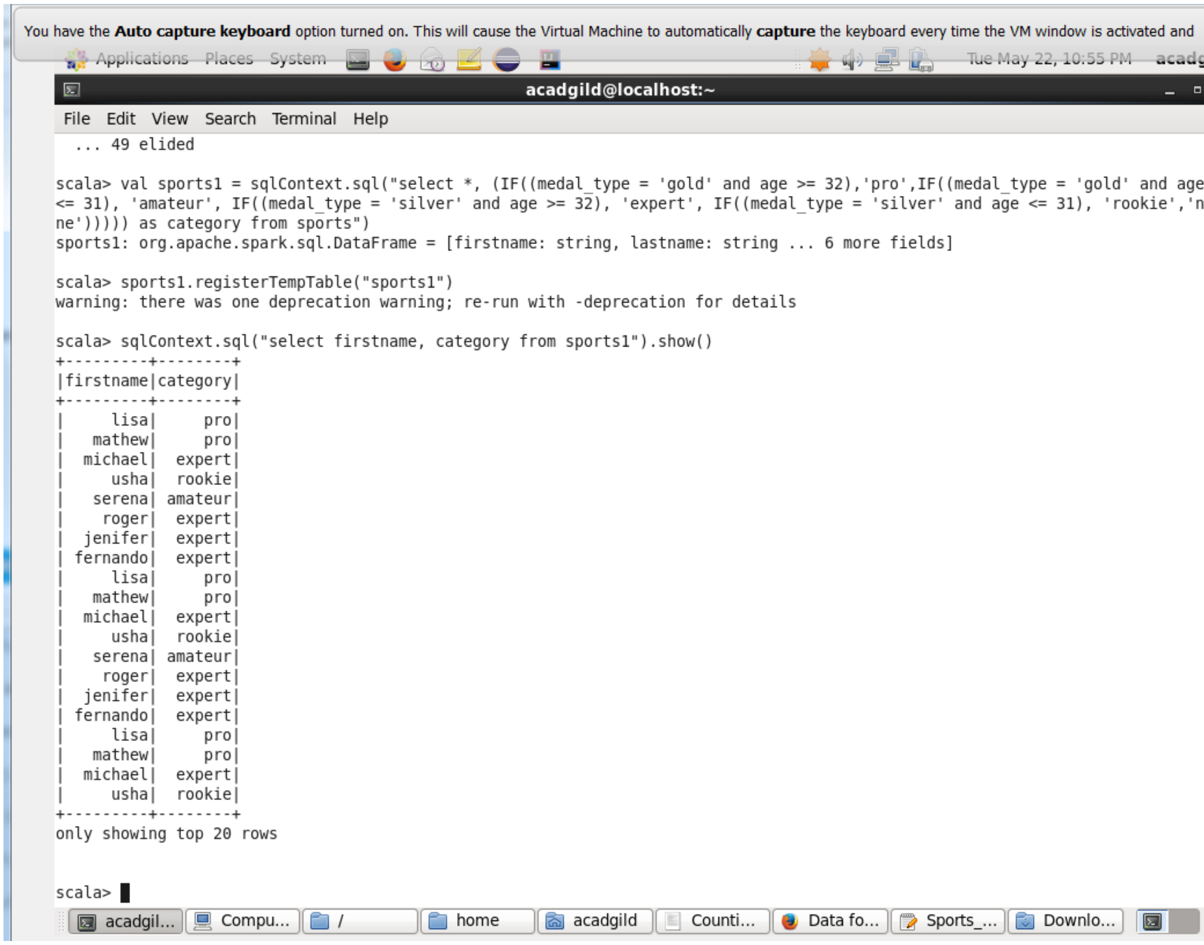
You have the **Auto capture keyboard** option turned on. This will cause the Virtual Machine to automatically **capture** the keyboard every time the VM window is activated and

```
               ^

scala> sqlContext.udf.register("createnewname", (colA: String, colB: String) =>
{ "Mr." + colA.substring(0,2) + colB } )
res27: org.apache.spark.sql.expressions.UserDefinedFunction = UserDefinedFunctio
n(<function2>,StringType,Some(List(StringType, StringType)))

scala> sqlContext.sql("select firstname, lastname, createnewname(firstname, last
name) from sportsdata").show()
+---------+--------+--------------------------------------+
|firstname|lastname|UDF:createnewname(firstname, lastname)|
+---------+--------+--------------------------------------+
|     lisa|  cudrow|                           Mr.licudrow|
|   mathew|   louis|                            Mr.malouis|
|  michael|  phelps|                           Mr.miphelps|
|     usha|      pt|                               Mr.uspt|
|   serena|williams|                         Mr.sewilliams|
|    roger| federer|                          Mr.rofederer|
|  jenifer|     cox|                              Mr.jecox|
| fernando| johnson|                          Mr.fejohnson|
|     lisa|  cudrow|                           Mr.licudrow|
|   mathew|   louis|                            Mr.malouis|
|  michael|  phelps|                           Mr.miphelps|
|     usha|      pt|                               Mr.uspt|
|   serena|williams|                         Mr.sewilliams|
|    roger| federer|                          Mr.rofederer|
|  jenifer|     cox|                              Mr.jecox|
| fernando| johnson|                          Mr.fejohnson|
|     lisa|  cudrow|                           Mr.licudrow|
|   mathew|   louis|                            Mr.malouis|
|  michael|  phelps|                           Mr.miphelps|
|     usha|      pt|                               Mr.uspt|
+---------+--------+--------------------------------------+
only showing top 20 rows


scala> 
```

Add new column with category names for athletes based on medals earned and age.

1) Val sqlContext = new org.apache.spark.sql.SQLContext(sc)
2) Val data = sc.textFile(“/home/acadgild/Sports_data.txt”)
3) Val header = data.first()
4) Data1 = data.filter(row=>row != header)
5) Case class sports_class(firstname:String, sports:String, medal_type: String, age: Int, year: Int, country: String)
6) Val sports = data1.map(x=>x.split(“,”)).map(x=>sports_class(x(0), x(1), x(2), x(3), x(4).toInt, x(5).toInt, x(6))).toDF
7) Sports.registerTempTable(“Sports”)
8) Val sports1 = sqlContext.sql(“select *, IF((medal_type = ‘gold’ and age >= 32), ‘pro’, IF((medal_type = ‘gold’ and age <= 31), ‘amateur’, IF((medal_type = ‘silver’ and age >= 32), ‘expert’, IF(medal_type = ‘silver’ and age <= 31), ‘rookie’, ‘none’))))) as category from sports”)
9) Sports1.registerTempTable(“sports1”)

10) sqlContext.sql(“select firstname, category from sports1”).show()



You have the **Auto capture keyboard** option turned on. This will cause the Virtual Machine to automatically **capture** the keyboard every time the VM window is activated and

```
  ... 49 elided

scala> val sports1 = sqlContext.sql("select *, (IF((medal_type = 'gold' and age >= 32),'pro',IF((medal_type = 'gold' and age
<= 31), 'amateur', IF((medal_type = 'silver' and age >= 32), 'expert', IF((medal_type = 'silver' and age <= 31), 'rookie','n
ne'))))) as category from sports")
sports1: org.apache.spark.sql.DataFrame = [firstname: string, lastname: string ... 6 more fields]

scala> sports1.registerTempTable("sports1")
warning: there was one deprecation warning; re-run with -deprecation for details

scala> sqlContext.sql("select firstname, category from sports1").show()
+---------+--------+
|firstname|category|
+---------+--------+
|     lisa|     pro|
|   mathew|     pro|
|  michael|  expert|
|     usha|  rookie|
|   serena| amateur|
|    roger|  expert|
|  jenifer|  expert|
| fernando|  expert|
|     lisa|     pro|
|   mathew|     pro|
|  michael|  expert|
|     usha|  rookie|
|   serena| amateur|
|    roger|  expert|
|  jenifer|  expert|
| fernando|  expert|
|     lisa|     pro|
|   mathew|     pro|
|  michael|  expert|
|     usha|  rookie|
+---------+--------+
only showing top 20 rows


scala> 
```